THE MAGAZINE OF DIGITAL TRANSFORMATION

DIGITALE WELT

SCIENCE MEETS INDUSTRY

Issue 1 ▪ January ▪ February ▪ March ▪ **2020**

ISAAI'19 Proceedings included

Applied AI –
Getting ready for the next generation

Deep Learning
Machine Learning at Large-Scale

Autonomous Agents
Self-Adaptive Services

Quantum Computing
Accelerating AI

PATTERN RECOGNITION
Towards intelligent solutions

Prof. Michael Winikoff
About Autonomous Systems and Agent-Oriented Software Engineering

€ 19,50
€ 21,90

DIGITALE WELT ISSUE 1 | 2020

The next DIGITALE WELT will be released on 06.03.2020

Titel: Rost-9D/istockphoto.com, Robert Cross Fotos: Rost-9D/istockphoto.com, Robert Cross, Privat

Artificial Intelligence – the new Revolutionary Evolution

Recent advances in Artificial Intelligence (AI) have paved the way for exciting applications, which are now playing important roles in everyday life ranging from language translation and image processing to recommender systems and autonomous driving. Most applications are based on machine learning, which achieved great successes due to increasingly available computational resources and data. Applying AI to diverse areas is appealing due to its self-organizing and self-learning nature, which enables generic ways of solving problems with minimal effort on specifications. The current trend of applied AI offers numerous opportunities to contribute within areas of research, theory, technology, and application.

Overview - What makes a system intelligent?

The term intelligence is generally not well defined and understood since it is a hot topic for research in various areas like biology, psychology, and neuro sciences itself. Artificial intelligence exploits the knowledge of these areas to create machines and programs which are able to solve problems in an intelligent and adaptive manner with only little problem specific knowledge than the problem formulation itself. The main components of intelligent systems are a Learning component, a Thinking, and an Acting component:

The Learning component which is usually implemented via Machine Learning methods is either trained with a lot of available data to evaluate states and moves based on Supervised Learning or Unsupervised Learning or with generated data, which is obtained automatically by using trial-error approaches like Reinforcement Learning. Reinforcement Learning has become a popular field in industry and research due to its self-adapting nature for intelligent systems. In Reinforcement Learning an autonomous entity or agent trains itself to achieve a given goal by interacting with the environment similarly to living beings. While there is no explicit guidance to achieve that goal, the agent gets numeric feedback which is known as reward signal. The reward itself is inspired by conditioning from the field of psychology. The agent has to adapt its behavior to maximize the long term expectation of the cumulative reward. In a game the goal would be to win against a strong opponent, or in navigation the goal could be to reach a target destination within given constraints.

The Thinking component involves explicit reasoning about future actions and events. In a game, different future scenarios can be considered to decide on the next move that maximizes the probability of winning, while in a navigation task, careful routing and online planning would be required to safely move an autonomous entity like a car to a desired destination without unnecessary detours, costs, or accidents. Technically, artificial thinking is realized by search or optimization techniques like tree search (e.g., Monte Carlo Tree Search or Alpha-Beta Search), meta-heuristic optimization (e.g., Evolutionary Computation or Simulated Annealing) or routing algorithms (e.g., Dijkstra or A* Search). A model of the environment like a simulator, a map or simply the rules of the games are required for effective search. In situations, where models are not available, Machine Learning could be used to bridge the gap between Learning and Thinking. An example for this is Model-based Reinforcement Learning, which additionally learns a model of the environment, while adapting its strategy to it. Thus, Model-based Reinforcement Learning can be considered as a combination of Learning and Thinking. It closely resembles real intelligent behavior.

The Acting component makes decisions based on information provided by the Learning and the Thinking components. While Learning and Thinking are required to extract useful information from data, simulations, or other kinds of reasoning and optimization, the Acting component bears the “responsibility” of the final decisions made. The Acting component has to consider all the information either learned, thought, or provided by third parties like the engineers, the customers, other intelligent systems, etc. to make a decision which can be considered as “intelligent” by the outside world. Acting involves many aspects of different disciplines ranging from social sciences and psychology, to biology and neuro science. An important aspect of Acting is the consideration of other intelligent beings like living beings (e.g., humans) or other artificially intelligent systems. Such situations are very challenging due to each system trying to achieve its own goal (which might be in conflict to each other) without harming the global world. Acting requires a lot of responsibility, attention, and transparency to be implemented in a meaningful way, to be beneficial for industries and everyday life.

Current Challenges and Trends

Today many of the basic techniques mentioned before are now used in various fields to realize features like Image Processing to sharpen blurry images or to detect objects at different orientations and scales, or Pattern Recognition to identify human behavior and to classify customers in order to improve the service or user experience. The extracted knowledge is often used by Recommender Systems to provide suitable advertisement to the user. Predictive Maintainance should support operative tasks by proactively inform the status to people in charge to avoid outtakes and to optimize processes. Industry 4.0 is an upcoming trend where AI methods are used to create self-organizing systems, which are able to produce individual products with only little handcrafting of manufacturing processes. Autonomous driving is one of the long standing visions of using AI in the real world, by using powerful learning and thinking systems to create self-adaptive vehicles which can safely transport people and objects to their target destinations. Language Processing is an important field to bridge the gap between different human cultures but also between human and machine especially in education, where robots are meant to support teachers and students in their daily tasks. Another upcoming field is AI in Healthcare, where mostly Machine Learning is used to make diagnosis and to evaluate XRay images to make more reliable decisions than humans, which can save lives. A really appealing field is using AI for Social Good. That is to use Learning, Thinking, and Acting approaches to combat common threats of nature or humanity. One example is to catch poachers or to find snares in order to protect wildlife. Recent approaches often use Game Theoretic approaches which can be classified as Acting method.

Although there are many active fields about AI, there are still many challenges to be addressed: while most successes are based on Machine Learning, which itself is enabled by the availability of computing power, scaling up a problem also requires a scalable Machine Learning approach. This could be achieved by designing distributed algorithms or by devising special hardware like the neuro chips of Intel or the TPUs of Google. While there exist many different open-source frameworks for Machine Learning or Deep Learning like TensorFlow or PyTorch, there is currently no open-source framework which integrates all three components (Learning, Thinking, Acting) to provide a uniform base to create powerful intelligent systems. More work needs to be done here to unify the perspectives on AI.

There is a general consensus that AI needs to be explainable and make responsible decisions, especially in cases where safety is critical like in autonomous driving or healthcare. Current AI systems are always based on Deep Learning which are very hard to interpret and only work in a black box manner. In order to make AI widely acceptable, transparency and explainability will be a key property and a great challenge for research and industry, which requires more thoughts and innovation.

How to prepare for the next AI wave?

Right now, AI seems to be everywhere, and everyone who is lacking AI technologies seems to be out of place. But the right thing to do is not to adapt to the existing trend but to prepare for the next AI wave.

As more and more research will be done on AI, a first step to get into the community, other people, and companies is to attend international conferences about AI topics. Since AI is a very wide field, there exist many top venues like NeurIPS, ICML, ICLR, CVPR, AAAI, IJCAI, which are held annually with thousands of visitors all around the world. Visiting these conferences is a good way to start getting into the topics and trends.

Another important step is to create a social network to get in contact with experts and other companies to gain know-how and to collaborate on common projects. This can be initially done at such conferences.

The next crucial step to prepare for the next AI wave is to hire experts from required fields. This can be done after gaining the know-how about the own requirements and after extensive exchange with other experts and companies. Knowing about all of these is important to hire the right people for the right purpose and to maximize the effect within the own company.

For more information on this subject, please feel free to visit our AI-Lab at (http://www.mobile.ifi.lmu.de/ai-lab/). We are looking forward to your visit!

Thomy Phan, Dr. Sebastian Feld, Prof. Dr. Claudia Linnhoff-Popien

Thomy Phan

Thomy Phan is a PhD Student in Computer Science at LMU Munich. His research focuses on artificial intelligence and autonomous systems. He is head of the AI Laboratory of the Mobile and Distributed Systems Group at LMU Munich and member of the Organizing Committee of the First International Symposium on Applied Artificial Intelligence (ISAAI'19).

Dr. Sebastian Feld

Dr. Sebastian Feld is head of the Quantum Applications and Research Lab (QAR-Lab) at the Mobile and Distributed Systems Group of the LMU Munich.
Currently, he pursues the goal of habilitation with a main focus being on optimization problems and the application of quantum technology.
He joined LMU in 2013 and earned his doctorate in 2018 working on planning of alternative routes, time series analysis and geospatial trajectories.

Prof. Dr. Claudia Linnhoff-Popien

Prof. Dr. Claudia Linnhoff-Popien holds the chair „Mobile and Distributed Systems" at the LMU Munich. She is board member of the Institute for Informatics, member of the „Münchner Kreis" and co-founder of the ALOQA GmbH. Further, she is head of the lead project „Innovationszentrum Mobiles Internet" of the Zentrum Digitalisierung.Bayern (ZD.B) funded by the state of Bavaria. She is also scientific advisor of the VIRALITY GmbH and chair of Digitale Stadt München e.V.

Fotos: Privat

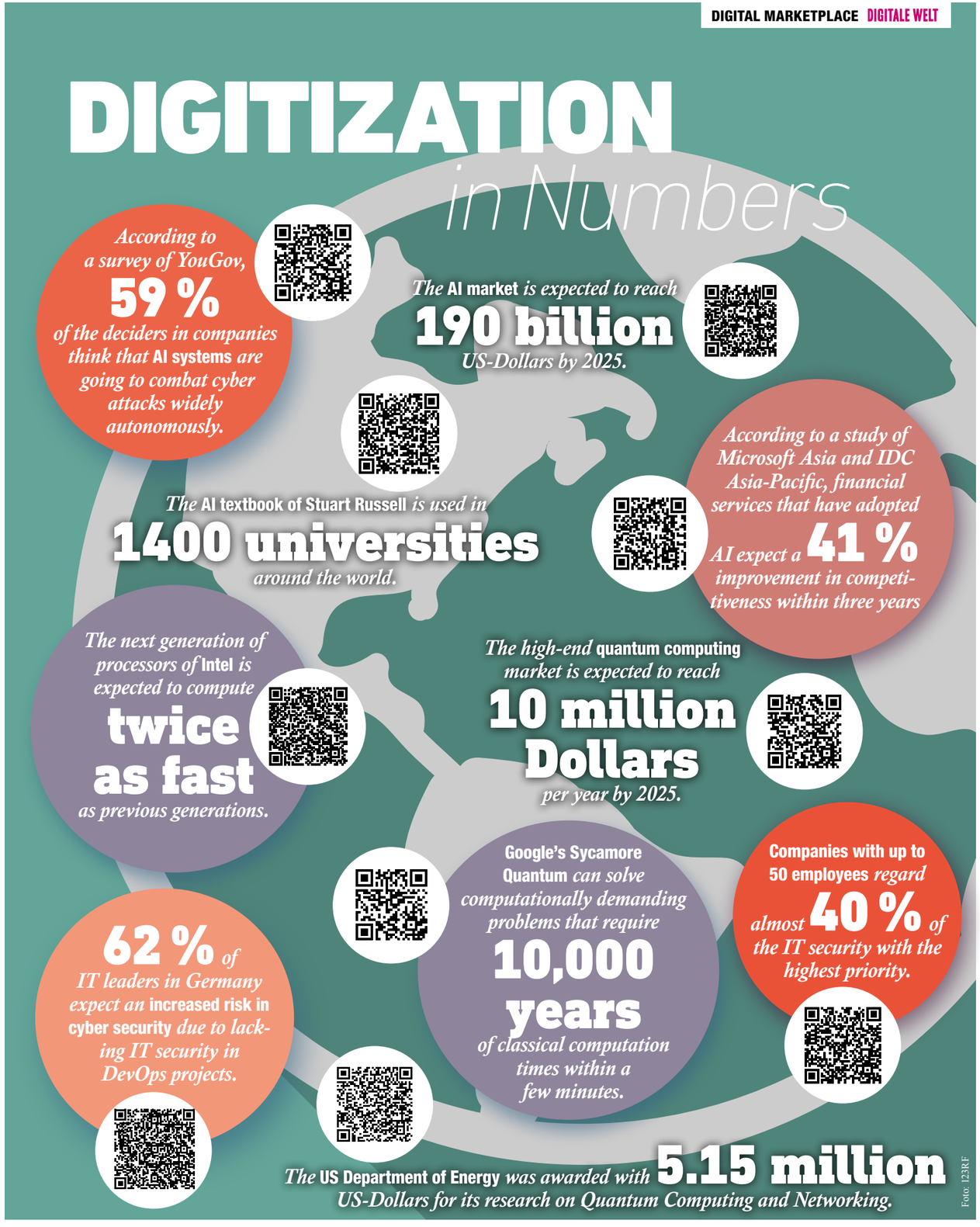


Foto: 123RF

Autonomous Systems: From labs to lives

Prof. Michael Winikoff is an expert in Agent-Oriented Software Engineering. In an interview with the DIGITALE WELT Magazin he allows us a glimpse into his prolific research. He explains the challenges in the field as well as the responsibilities towards the consumers in regard to autonomous systems.

You are not just a scientist, but also a pianist and composer. Did music teach you something about software?

Hmm. That's a good question. I can see a range of things that music and software have in common, but I don't think I've learned anything specific about software from music. More broadly, music, software, and mathematics, all share some common ideas about having rules and structures, and are all concerned with abstract (non-physical) creations. Creating a new piece of music, writing software, or proving a theorem, all have in common working within a formal framework to create something new, and with all three the concept of elegance is relevant. Of course there are also differences: the goal of music is to communicate, in particular emotions, whereas the goal of software is to perform a specified function.

You made research on the factors which enable humans to trust autonomous systems. Please describe them briefly.

Well, the first thing I should say is that the list of factors is not meant to be a definitive list that is always applicable, and complete. Rather, these are some factors that are important, but there are so many different domains in which autonomous systems can be used, that no list is going to be universal. Some domains might have additional factors, and for some domains not all common factors would be relevant.

In any case, the factors that I discuss in my work are: recourse, explanation, verification & validation, and incorporating human values.

Recourse is the idea that there has to be a way to deal with the consequences of an autonomous system doing the wrong thing. If, say, a self-driving car crashes into your fence, how do you get compensated for this? This is obviously primarily a legal and social question, but it is crucially important because no technology is perfectly able to function in all situations.

For a wide range of domains it can be important that autonomous systems have the ability to explain the reasoning that led to a particular action, or course of action, being performed. This is important because sometimes an autonomous system will behave in a way that might be correct, but not obviously correct. A simple example is when a GPS takes you on an unusual (and longer) path. If you don't realise that there has been a traffic accident on the usual route, you might not understand why the GPS is doing something unusual, and your trust in it might reduce. Of course, explanations need to be given in a form that is comprehensible.

Autonomous systems, especially if they operate in safety-critical domains, need to be able to guarantee that they will never do certain things. For example, a robotic medicine dispenser needs to be able to guarantee that it will deliver the correct medicines at the correct time to the correct patient. Traditionally, we assess software by testing it in a range of scenarios. However, when there are many many possible situations that can be encountered, testing becomes infeasible, and there is an important role for formal verification techniques.

Finally, autonomous systems will often function in the context of human society, and to make good decisions and act in an appropriate way, it can sometimes be important for the systems to have representations of human values, and to be able to reason about them and take them into account. For example, in what situations can a personal assistant share a person's location? And with whom? Making this sort of decision requires understanding of privacy, and how it trades off against other important values (such as safety). It also requires awareness of social relationships: sharing a child's location with their parent is different to sharing their location with their friends or with their teachers.

"Autonomous systems, especially if they operate in safety-critical domains, need to be able to guarantee that they will never do certain things."

In your article from "The Conversation", you argue that many decisions which autonomous systems take should be based on human values. However, who decides on those values, since they can differ from culture to culture?

Indeed. And they are also clearly not uniform even within a single culture (however one might define culture!). What technology can aim to provide, is a framework that can be instantiated with different values and priorities. For example, a personal assistant might have the ability to reason about privacy when sharing location information. This could be instantiated with different rules for different contexts. This customisation could be done to some extent at a country or region level (for example, operating with the EU has certain implications for privacy rights), but also by individuals. How to represent these values in a way that allows effective customisation, is still a research challenge.

In your opinion: Does the industry do enough to help clients have trust in autonomous systems?

I don't want to be negative, but I would say that there is more to be done by many stakeholders, not just industry. Industry certainly has a crucial role to play. But there is a tension between rushing to develop and deploy certain technologies, and taking care to ensure that the technologies are fit-for-purpose. This is why there is also a crucial role for regulators and governments: we cannot leave this to industry.

You are best known for your work on design methodologies for agent-based systems, foremost among them, the Pro-

metheus methodology. Prometheus is a detailed and complete (start to end) methodology for developing intelligent agents. Could you give an example of its use? And can you briefly describe the methodology?

Sure. It's been used for a range of multi-agent systems in the literature. Without going to the literature and searching, I'll mention just a couple of smaller examples: a book store realised as a multi-agent system [MAS] (running example in Lin and my 2004 book), and a meeting scheduling system, I can also recall seeing work on a UAV design.

The Prometheus methodology provides concepts, a process, and notations for designing multi-agent systems. It also, crucially, provides detail on how to do various things, for example, if part of the process is to identify the agent types in the system, it's important that a designer (especially one not already experienced in designing MAS) has good guidance on how to identify agent types, and what are the trade-offs involved.

Prometheus consists of three phases (although of course they are not done in a strict linear sequence): specifying the system-to-be in terms of its goals and the environment it interacts with, designing the system by defining the agent types and how they interact (using interaction protocols), and doing a detailed design for each agent type. This last step, detailed design, is where Prometheus assumes, for concreteness, that goal-plan agents (also known as "BDI" agents) are used, but the other parts of the methodology do not assume this.

Which issues are problematic when it comes to testing multi-agent systems? And why is it especially difficult to test Belief-Desire-Intention (BDI) agents?

"There is a tension between rushing to develop and deploy certain technologies, and taking care to ensure that the technologies are fit-for-purpose."

There are a number of issues that combine to make testing multi-agent systems very difficult. Firstly, MAS are by definition parallel distributed systems. Secondly, such systems often have to deal with challenging environments that are complex, non-deterministic, and where things can go wrong, and the systems are expected to be able to recover from such failures. And thirdly, the reasoning mechanism that cognitive architectures use, such as the BDI model, can be powerful, but also can make testing a challenge. Specifically, in some work (with Professor Cranefield) we analysed the BDI model and showed that even relatively small BDI goal-plan trees can give rise to enormous numbers of possible execution traces, which makes assurance through traditional testing infeasible.

How is intention defined in BDI agents?

Well, it depends on your perspective.

Seriously, one of the things that are confusing about the BDI model is that in the literature there are a number of different perspectives. There's the original philosophical work on folk-psychology by Michael Bratman, which is not about software at all. Then there is work on formalising concepts using modal logic, and then there is work about software architectures and languages.

Focusing on the software languages perspective, an intention can be thought of as a partially instantiated plan that the agent is intending to carry out. For example, if an agent has a goal, and it has instantiated a plan to realise the goal, then the parts of the plan instance, that have not yet been done, are the intention.

In practice, when creating BDI agents, the programmer focuses on writing plans, not on intentions. Intentions are not the focus, since they basically are a run-time structure that results from running the plans.

Why is it important that agents generate protocols at runtime?

Well, I would say that it isn't important in general, but that in some situations it becomes important. Having predefined protocols that are created at design-time is simpler. But in some situations, runtime protocols are needed. For example, if one has a system where new agents can join the system (an "open system"), then it may be necessary to allow for protocols to be created at runtime.

What makes debugging cognitive agent programs difficult?

The execution cycle of cognitive agent programs can be difficult to follow. For example, if a particular action was performed, trying to work out why can involve tracing back through various plans, the conditions that were true when those plans were selected, and even failures of earlier actions that resulted in other plans being attempted.

Prof. Michael Winikoff

Prof. Michael Winikoff is a professor at the School of Information Management at Victoria University of Wellington. Michael's research has focussed on software that is conceptualised in terms of "intelligent agents" which are able to exhibit robust and flexible behaviour. Since 1999 he has worked on developing approaches for engineering these sorts of systems. More recently he has been looking at explainable AI, societal consequences of autonomous systems, and the issues that affect trust in these systems. Michael is on the programme committee for the First International Symposium on Applied Artificial Intelligence (ISAAI'19) which will be held in Munich in November. He is also an Associate Editor for the Journal of Autonomous Agents and Multi-Agent Systems, Editor-in-Chief for the International Journal of Agent-Oriented Software Engineering, and he has been both programme and general chair of the International Conference on Autonomous Agents and MultiAgent Systems.

In one of your papers, you argue that it is time to begin the development of a next-generation agent-oriented software engineering (AOSE) methodology, leading ultimately towards a unified AOSE methodology. Why is a unified AOSE methodology of such vital importance?

There's an analogy with object-oriented design: in the early days of OO design there were many methodologies. This diversity is unavoidable in the early days of developing new methodologies, but it poses challenges. Firstly, a practitioner needs to select an appropriate methodology amongst the many available (which is difficult, since they need to have some familiarity with the different options). Secondly, tool support becomes difficult to provide, since with many methodologies, there need to be many tools, what requires much more effort across the community. And, of course, having multiple notations hinders communication and education.

In the OO world, as we know, key people got together, and ultimately the Unified Modeling Language (UML) was created. This provided a single common notation that all designers could learn, and that different tools could support.

In the agent world this has not yet happened: there are still many (dozens of) methodologies, although only a few are well developed and have seen substantial use.

Will traditional AI planning systems and procedural reasoning system (PRS) coexist next to one another in the future? Or will PRS replace traditional AI planning systems?

"In terms of some other applications of autonomous systems that merit attention I would mention drones, a wide range of robots, and, moving away from physically embodied systems, smart grids, smart homes, and digital personal assistants."

I think they will co-exist, and in fact, there has been work on how to better integrate the two. There are two differences between traditional AI planning systems and BDI systems such as PRS. Firstly, traditional AI planning searches to find a complete plan, and it is then executed. On the other hand, situated systems need to interleave planning and acting. Secondly, traditional AI planning involves assembling individual actions into plans based on each action's pre and post conditions, whereas PRS uses human-crafted recipes for combining actions. It is worth noting that HTN planning (Hierarchical Task Network) sits in between: it uses human-crafted recipes, but does lookahead planning.

Are there any alternatives to AgentSpeak as far as agent-oriented programming languages are concerned?

Yes, there are dozens of BDI agent-oriented programming languages. AgentSpeak was proposed by Anand Rao (back in the mid 90s) as an abstraction of previous notations (PRS, dMARS). It was subsequently implemented by a few people, with the most influential, and widely-used, implementation being Jason (subsequently integrated into JaCaMo (Jason+CArtAgO+Moise)). BDI languages other than AgentSpeak include JACK, Jadex, 2APL, Gwendolen, and also one could argue GOAL. But this is far from a complete list!

You are a professor of the School of Information Management at the Victoria University of Wellington. What do students struggle with most, when it comes to understanding autonomous systems?

I have to say that my experience in teaching autonomous systems has been primarily at RMIT University, where Professor Lin Padgham and I had an undergraduate course that taught both agent programming and agent design. At Otago University (where I was until earlier this year), there was less opportunity to teach autonomous systems.

I would have to say that my experience was positive: students (second year undergraduate students) did not really struggle. They were able to understand the concept of autonomous systems, and design and build simple systems in the course of a single semester.

As a professor: Should computer science students start with learning agent-oriented programming or object-oriented programming first? Or should they learn both simultaneously?

I would say that they should learn OO first. Object-oriented programming is more general: not everything is an agent. Of course, there is also a debate about whether objects should be learned first, or procedural programming …

There is a lot of talk about self-driving cars. However, is there a less broadly discussed use of applied autonomous systems which will be of importance in the future?

Self-driving cars are a "flashy" technology, and also one that we can all relate to. However, it is unfortunate that so much attention is focussed on them, because they have some quite particular characteristics, which are not shared by other autonomous systems. For example, being safety critical yet operating in an incredibly challenging environment.

In terms of some other applications of autonomous systems that merit attention I would mention drones, a wide range of robots, and, moving away from physically embodied systems, smart grids, smart homes, and digital personal assistants.

Finally, is there anything else you'd like to mention?

In the interests of keeping this from being too long, I'll just mention one thing.

The development and deployment of AI and of Autonomous Systems obviously poses many challenges for societies. The key questions that we need to grapple with include: what applications are acceptable, and how should we respond to developments?

Let me give a few quick examples. Facial recognition is an example of a technology where acceptability is currently being debated, and some places have banned it. What forms of use of facial recognition should be allowed and accepted (in a given social context)?

Another example are Lethal Autonomous Weapon Systems, where there is a strong argument for having a pre-emptive ban.

An example of a change that requires response is the workforce and the economy. To the extent that automation will result in substantial changes to the workforce, how should society respond to that? What makes such debates difficult is that there are things we just do not know, and cannot predict. We know that automation will transform some jobs (potentially changing the demand), and that it will eliminate some jobs, while creating others. What we do not know are the numbers and patterns.

Interview: Florentina Hofbauer
Text: Michael Winikoff

Fotos: Robert Cross

Marcus Raitner works as an agile transformation agent and agile coach at BMW Group IT. In his blog "Führung erfahren!" he has been writing about leadership, agility, digitalization, and much more since 2010.

The Agile Counterfeiters on Their Way to Cargo Cult Hell

Anyone who imitates Spotify or introduces SAFe or obtains imitated or falsified agile frameworks and disseminates them as best practice will be punished with futile ritual practices of not less than 20 hours per week and employee. The way into the agile cargo cult hell is well paved with best practices, blueprints and frameworks and is bordered by billboards saying: "Don't invent the wheel again!" Agility, however, is less a question of methods than of principles and stance.

A British statistician once said, that all models are wrong, but some are useful. So, let's get this straight: All frameworks and models have their justification and are useful in some way and in the right context (yes, even heavyweight frameworks like SAFe have useful aspects). The problem thus is not the methods, but blind belief in methods. However, all models are wrong if they are introduced and rolled out by top-down directive without the necessary understanding of the principles behind them and without the right attitude.

Yoshihito Wakamatsu reports an interesting anecdote about Taiichi Ohno, who significantly influenced and further developed the Toyota Production System, in his book "The Toyota Mindset, The Ten Commandments of Taiichi Ohno". During a visit to a Toyota plant, Ohno was accompanied by another manager. This manager noticed the apparent flaws in the implementation of the Toyota Production System and asked Ohno why he had not corrected them immediately. His answer:

I am being patient. I cannot use my authority to force them to do what I want them to do. It would not lead to good quality products. What we must do is to persistently seek understanding from the shop floor workers by persuading them of the true virtues of the Toyota System. After all, manufacturing is essentially a human development that depends heavily on how we teach our workers.

Needless to say, Ohno could have demanded compliance with the Toyota Production System by virtue of his authority. But he had understood that the resulting blind obedience would create more problems in the long run than this short-term cure would solve. Instead, he relied on teaching the workers better in the principles behind his model.

For the mere implementation of a framework this is clearly a very tedious and time-consuming procedure. However, Taiichi Ohno aimed for more. For him, the empowerment and enablement of the people affected was an integral part of this change. And that not only for the mere application of a few methods, but also and especially for the continuous further development and adaptation of the Toyota Production System along its underlying principles. This is why Taiichi Ohno demands: "Something is wrong if workers do not look around each day, find things that are tedious or boring, and then rewrite the procedures. Even last month's manual should be out of date."

It is precisely this empowerment that makes the decisive difference between beautifully celebrated cargo cult and real change. So, there's no reason why the agile transformation shouldn't be inspired by good examples like Spotify or based on (seemingly) ready-made frameworks like SAFe, but please not only superficially! As with the often imitated and rarely reached Toyota Production System by Taiichi Ohno, this means promoting the self-organization of those affected on the basis of principles and thereby enabling and empowering them to further develop these methods. This way is stonier, but it is worth it.

Dr. Marcus Raitner

Foto: Privat

The Book "Manifest für menschliche Führung" is available as paperback and e-book at Amazon

ARTIFICIAL INTELLIGENCE ...

»MENTAL INTELLIGENCE«

How to influence our brain for better results

Karen was sitting at her desk, eyes closed, wearing headphones obviously listening to music.

Her manager standing next to her finally tapped on Karen's shoulder. »This is not the time for deep relaxation; we have to finish that project planning in one hour!« Karen opened her eyes. »I'm just manipulating my brainwaves for a better result, you want to try it yourself?«

What exactly was she doing? She simply listened to music with alternating acoustic signals. We know that billions of nerve cells in our brain communicate via bioelectrical impulses. When a stimulus arrives, this information is processed. Frequency fluctuations then take place. Certain frequencies are assigned to certain states of the person.

Some frequencies are particularly suitable for creative processes or for storing memories.

Here is a list of the different frequencies and their characteristics:

Gamma 100 – 38 Hz:	Peak performance, assembling specific perceptions. Measurable while performing highly concentrated work.
Beta 12 – 38 Hz:	Normal frequency band, alert, attentive, focused. Measurable while performing concentrated work.
Alpha 8 – 12 Hz	Alert but relaxed, very creative. Measurable after waking up, before falling asleep, during relaxation exercises or light meditation.
Theta 3 – 8 Hz	Dream state, subconscious is active. Measurable in REM phases, during deep relaxation or meditation.
Delta 0.2 – 3 Hz	Deep, dreamless sleep, consciousness is switched off.

According to recent studies, some frequencies can be easily changed in listening to sounds accompanied by pleasant music, alternating between the left and the right ear. These tones provide an impulse for slight eye movements to the left and right. These eye movements are found in the REM (Rapid Eye Movement) sleep phase, in which the emotional experiences of the day are processed in the brain. They also create inner balance and improve concentration and creativity. Therefore, EMDR or wingwave® music is used in some companies, for example in development laboratories to increase creativity or in call centers to reduce strain and tension. Especially for pupils diagnosed having an attention deficit syndrome, there are trainings using neurofeedback. Participants train to focus their concentration on a special task, for example to move a figure on a screen simply with the power of thoughts. The processes and results of these trainings are then mentally transferred in daily life at school.

Frequencies can also be changed in order to give the brain a necessary frontal brain pause. Mindfulness exercises and meditations for example have an effect on the alpha and theta states and help to relax the brain and body. This, in turn, ensures that we can get a fresh start with more concentration and creativity.

In addition to that, there are many international studies on how meditation works. Most importantly, they prove that the technique builds lasting emotional stability. At the same time, they have proved that sleep disorders, anxiety disorders, and depression can be reduced and disappear completely through regular meditation without the use of psychotropic drugs. In many cases, a positive influence of regular meditation – without medication – in lowering blood pressure has also been confirmed.

Considering the power of thoughts influencing the metabolism in any direction, we must recognize that our brain is a treasure chamber. We easily can find mental pictures that via dopamine influence the outcome of our activities - either positive or negative. For example, if we have a clear picture showing the closing of a negotiation, our brain system will focus attention and energy on this result due to a cocktail of dopamine, endorphin and oxytocin. And our rewarding system will finally help to relax and to refresh to start again.

There are a lot of simple techniques – it's definitely worth analyzing and perhaps adding some of these to your toolbox.

I would like to invite you on this trip discovering how to direct your thoughts for more well-being, health, and success in your life.

Enjoy!

Dr. Petra Bernatzeder,
psychologist, consultant, coach, director of upgrade human resources
www.upgrade-hr.com

Foto: Privat

Help me, Machine!

Is this the next step in humans' evolution: after hand axe, wheel, and electricity humans create self-thinking machines as their new helping hand. Is this kind of super-invention the solution for such a high amount of problems? We spoke with Jiayu Zhou not only about promises and benefits of our new assistants, but also about their limits and risks.

Jiayu Zhou has been working on machine learning theories and methodologies, including multi-task learning, feature selection, sparse learning, matrix completion, numerical optimization and their applications on healthcare analytics. He currently works as an Assistant Professor in the Department of Computer Science and Engineering at Michigan State University.

In which sectors of a human being can machines work better than humans?

When defining "better", there are two dimensions. The first dimension is "faster", and the other dimension is "more accurate". We know that in many automation domains such as production, service domains, and warehouse management, a machine can do much faster and thus better than humans. On the other hand, many procedures require complicated decision making that is much more complex than automation, such as most medical diagnosis or operating vehicles. And in these cases human is still much better than machines are.

"Indeed, we can profile the genetics and predict many risk factors, but many many other factors depend on the environment, which is too complicated to be modeled even for a short period amount of time of a small cell, not to say the entire lifespan of a fetus."

In 1748 Julien Offray de La Mettrie published his scandalous book L'homme machine (Translation: „The man as a machine"). What do you think about this trans-humanistic concept of a man being a machine? Where are the boundaries between human and machine with your definition of machine?

I cannot agree with this statement, with my definition of "machine" as a human-made apparatus, such as computers. I think two differences between human and machine are that 1) knowledge-driven intelligence and 2) emotion. The former differentiates human and machines in terms of intelligence (natural or artificial), whereas the latter defines the basis of natural intelligence.

Why do machines need humans?

The question will be a much meaningful question if asked the other way round. Machines are created by humans and for the human to accomplish specific tasks, automation, entertainment, assistance, etc. And the purpose of creating machines is to improve our civilization and improve life-quality. Machines, on the other hand, do not have a purpose and cannot do anything beyond what human told them to do.

Can you imagine a world with machines and without humans? What do you think about ethical arguments about creating machines considering possible danger for human civilization?

If there are no humans, no one cares what machines would do. There are lots of educational science fictions warning us against the danger of machines, and however and also unfortunately, the technology we have currently mastered can only make machines to help with simple tasks. After all, there have no machines passed a Turning test yet.

How do you teach a machine to learn?

In the early years, humans taught a machine to learn by creating low-level instructions, in the form of programs. Later on, humans designed programs for higher-level instructions (ask the robot to turn left, find a box, and move to the right) which were then fulfilled by machines. More recently, due to the advances of machine learning, especially deep reinforcement learning, the machine can infer rules by experiencing the environment in a trial-and-error manner.

Is there a chance to create a machine which is able to teach other machines?

Yes, there is, through a human-made language in a human-designed way. Right now, developers have built micro-services which are talking to each other through machine-understandable formats such as JSON. By handing over experiences with each other, we can say the machines are teaching each other. However, I have to emphasize that human invents all the mechanisms, and therefore, they are not created by machines with any purpose.

In 2018 you participated at the 1st Int'l Workshop on Big Traffic Data Analytics at San Diego. Can intelligent machines help to solve our traffic issues? How?

Yes, intelligent machines can indeed help to solve our traffic issues. Machine learning scientists are now converting the traffic optimization problems into optimization problems (traffic prediction, traffic control, route planning, etc) and solving them in a data-driven way using the massive data collected in different sources. Over the last few years, we see high-accuracy traffic estimation status using trajectory data, efficient fleet management status using share ride dispatch data, and data-driven traffic light control systems that dramatically reduce the traffic in intersections, and many more to come.

"I think two differences between human and machine are that 1) knowledge-driven intelligence and 2) emotion."

Your researches take part of biomedical questions. Are there particular needs for using machines in medical contexts? Which are they?

Most of the current machine-assisted medical research takes advantage of medical data available from sources like mobile sensors, electronic medical records, cohort study data from clinical trials. Machine learning scientists like me use the data to build computational models by solving a data-guided large-scale optimization problem (e.g., classification, regression, or clustering). We heavily rely on machines' superior computing power to solve these (again human-designed) optimization problems. The procedure above shares the same spirit as those recent deep learning advances in other domains.

A vision: The tomorrow's gynecologist uses his computer to analyse the foetus. Artificial intelligence can recognize future prospects. Maybe the gynecologist can show us the whole foetus' life in a virtual preview. With the use of DNA-"correction" we can intervene in human's genesis. What do you think about it?

This vision was based on the assumption that a virtual preview is possible, but I don't personally think this is possible at the current computing resource and modeling technique we have. Common knowledge instructs that the development of a fetus is dependent on two aspects 1) genetics and 2) environmental interactions. Indeed we can profile the genetics and predict many risk factors, but many many other factors depend on the environment, which is too complicated to be modeled even for a short period amount of time of a small cell, not to say the entire lifespan of a fetus. Nevertheless, if someday we know for sure that some genes will definitely be giving an infant a liver cancer in her 70s, why not removing them? But given that the human development is such a complicated system, most likely we are going to play the trade-off game: do we want to get rid of genes and meanwhile take the risk of getting another disease in another period of her life.

Let's talk about science. What's the role of academic science in a world with artificial intelligence? Do you describe artificial intelligence in the meaning of understanding or do you produce artificial intelligence?

As the mainstream of artificial intelligence nowadays is, in fact, advanced machine learning that combines superior computational power and the vast amount of data we have (or can acquire), industries played an important role that is sometimes considered to be more important than academic science. However, our understanding is advanced machine learning supporting AI nowadays is somewhat limited and is much less than that of earlier machine learning algorithms. Academic science is carrying a critical mission in AI to advance such fundamentals and understandings. I would say this understanding is one of the most crucial steps toward producing artificial intelligence.

Is there a possibility to integrate machine's thinking in everyday's life? Can our smartphone be our artificial intelligent solution for everything?

I am not aware of machine's thinking. I do see human plots that are infiltrating every aspect of our lives through smartphones. We are interacting with Apps and websites every day, whose owners end up collecting all the behavior data from us and turn them into "intelligence" that ultimately becomes the source of their profit. How do big companies like Facebook and Google make money otherwise? Of course, I would like to thank them for the excellent free services they provide.

Machines are complex-organized containers full of data. In a world full of data – is there any chance of „private" and „secure" data?

Yes. If your data is not exposed to the internet, then it can be mostly thought of as private and secure. But it is really challenging to be as we are living in an ever-increasingly connected world.

Interview: Hannes Mittermaier

Prof. Jiayu Zhou

Prof. Jiayu Zhou is an Assistant Professor in the Department of Computer Science and Engineering at Michigan State University. He received his Ph.D. degree in computer science from Arizona State University in 2014. He has a broad research interest in large-scale machine learning and data mining, and biomedical informatics. He served as technical program committee member of premier conferences such as NIPS, ICML, and SIGKDD. Jiayu's research is supported by the National Science Foundation and Office of Naval Research. He is a recipient of National Science Foundation CAREER Award (2018). His papers received the Best Student Paper Award in 2014 IEEE International Conference on Data Mining (ICDM), the Best Student Paper Award at 2016 International Symposium on Biomedical Imaging (ISBI), and Best Paper Award at 2016 IEEE International Conference on Big Data.

Foto: Privat

FIRST INTERNATIONAL SYMPOSIUM ON APPLIED ARTIFICIAL INTELLIGENCE (ISAAI'19)

The First International Symposium on Applied Artificial Intelligence (ISAAI'19) promotes recent results of applications or practical methods of artificial intelligence from science and industry. ISAAI'19 publishes 17 high-quality international articles, with each of them presenting valuable results about applied artificial intelligence. As the first scientific event of the DIGICON, All accepted submissions of ISAAI'19 show the enormous interest of science and industry in applying artificial intelligence to real-world problems to improve our world and everyday life. ISAAI'19 is organized in four sessions Deep Learning & Natural Language Processing, Pattern Recognition & Medical Applications, Autonomous Agents, and Quantum Computing – Accelerating AI, which cover important aspects of Applied Artificial Intelligence by proposing novel and exciting techniques, analysis and applications for the next big AI wave.

ISAAI'19 Overview

The First International Symposium on Applied Artificial Intelligence (ISAAI'19) is an event to promote artificial intelligence research for real-world applications and use cases and was organized by Claudia Linnhoff-Popien, Thomy Phan and Sebastian Feld of LMU Munich. ISAAI'19 received 35 submissions from 14 countries coming from Africa, America, Asia, and Europe.

The submissions covered a wide range of applications and methods of artificial intelligence mostly related to the wide field of machine learning for pattern recognition and reinforcement learning for autonomous systems, as well as theoretic analysis about practice relevant aspects and deployment strategies for real-world use cases ranging from distributed architectures and special hardware.

Out of the reviewed submissions, 14 submissions were accepted as full papers (leading to a full paper acceptance rate of 40%) and 5 additional submissions were accepted as extended abstract (leading to an overall acceptance rate of 54,3 %).

All accepted full papers and extended abstracts are organized in four sessions Deep Learning & Natural Language Processing, Pattern Recognition & Medical Applications, Autonomous Agents, and Quantum Computing – Accelerating AI which are held during the DIGICON 2019 event, where all submissions are either presented orally or as poster. The poster session is held on Wednesday, November 20th and the oral presentation of publications is held on Thursday, November 21th parallel to the talks held at DIGICON 2019.

What to expect from ISAAI'19?

Right now, AI seems to be everywhere and everyone who is lacking AI technologies seems to be out of place. But the right thing to do is not to adapt to the existing trend as the trend always indicates a lagging but instead to proactively invest and to prepare for the next AI wave.

The DIGICON 2019 is an international business conference where science and industry meet to present and to learn about the most recent trends and technologies about digitization. This year the main topic of DIGICON is artificial intelligence which was inspired by the preceding events which focused on machine learning, data mining, and business intelligence. At DIGICON about 350 participants from all over the world are expected to attend this exciting event with top-talks of innovators, decision makers, and creative directors from science, business, and politics.

ISAAI'19 is the first scientific event of DIGICON and already received a considerable resonance from international experts, showing the enormous interest of science and industry in applied artificial intelligence as well as to connect to gain know-how about different areas in order to enable the application of artificial intelligence at an extremely large-scale.

The accepted publications along with the information about the authors of each paper or extended abstract are all available in this journal.

CONTENTS

Extracting Keywords from Publication Abstracts for an **Automated Researcher Recommendation System**

Marco Kretschmann, Andreas Fischer, Benedikt Elser

This paper presents an automated keyword assignment system for scientific abstracts. That system is applied to paper abstracts collected in a local publication database and used to drive a researcher recommendation system. Problems like low data volume and missing keywords are discussed. For remediation, training is performed on an extended data set based on large online publication databases. Additionally a closer look at label imbalance in the dataset is taken. Ten multi-label classification algorithms for assigning keywords from a given catalogue to a scientific abstract are compared. The usage of binary relevance as transformation method with LightGBM as classifier yields the best results. Random oversampling before the training phase additionally increases the F1-Score by around 5-6%.

1 Introduction

Modern research environments have become highly complex. Research is conducted more and more in projects with a clear set of goals. Project teams are increasingly composed in a multi-disciplinary manner. This can easily lead to duplication of effort on research questions in separate project teams. Even researchers at a single institution may be working on similar problems without knowing about each other. Recommendation systems are a field of artificial intelligence that provides methods which can help to overcome this problem.

At Deggendorf Institute of Technology (DIT) an app for researcher matching is currently in development. In this application users can create a profile with information about the membership at faculties, research groups, institutes and research projects. They can also add keywords to describe relevant research topics.

It is, however, complicated and time consuming for the user to find matching keywords in a given catalogue. Automating this process is important to simplify app usage and therefore raise adoption of the app. A naïve approach is exploiting keywords contained in the user's publications. However, those keywords are often unstructured. Even if they are structured, various publishers may use conflicting catalogs. Finally, many publications in the database lack the respective information.

In this paper, we present a keyword assignment scheme as part of our recommendation system. It learns an initial keyword catalogue based on paper abstracts taken from a local publication database. This catalogue is assumed to remain stable. The system is then used to automatically classify new papers which are added to the database. Paper abstracts are expected to be a highly condensed representation of the paper content. Thus, by learning from abstracts a catalog of high relevance keywords for previously unseen works can be created.

Two challenges have to be overcome for this project: Low volume of data for training and significant label imbalance. Regarding the first problem, the local publication database is missing either the abstracts or useful keywords in a lot of cases. In order to improve classification performance, information from external databases is used during the learning phase. Secondly, the data is heavily imbalanced: Some keywords are much more common than others. That makes it harder for learning algorithms to identify the rare ones during the training stage. To improve the performance despite this problem, the usage of sampling in order to achieve a balanced ratio of labels is evaluated.

The main contributions of this paper are: An analysis of the skewed data structure of our internal publication database; An evaluation of various machine learning classification methods applied to our problem; A discussion of the results and the applicability to an automated keyword extraction system for a local publication database. The classification is implemented as part of a recommendation system available to researchers at DIT.

The rest of this paper is structured as follows: In section 2, background information is given on keyword extraction and existing keyword extraction algorithms. Section 3 presents our methodology for creating the keyword extraction system. Section 4 introduces the measures that were used and discusses the achieved results. Section 5 gives an overview of the related work in the task of assigning keywords to scientific texts. Finally, section 6 concludes the paper and proposes future work.

2 Background

Keyword extraction is a common problem in the fields of text mining, information retrieval and Natural Language Processing (NLP). The goal is to assign a set of keywords, which describe its content, to a text. There exist many different approaches to this problem, which can be roughly divided into four categories [1]:

Statistical approaches: These methods take a look at the statistics of occurring words. Typical example for these approaches are Term Frequency – Inverse Document Frequency (TF-IDF), word cooccurrences or PAT trees.

TF-IDF is a commonly used method, which often finds also application in other approaches. The main idea is, that words, which occur often in a document but rare in the representative text corpus are interesting. It is defined as:

$$tf\text{-}idf = tf(t,d) * idf(t)$$

with $idf(t) = log(\frac{n}{df(t)+1})$

In this equation $tf(t,d)$ denotes the relative frequency of term t in a document d. It is is multiplied with the inverse document frequency $idf(t)$, which relates the number of documents n to the number of documents containing the term $df(t)$.

Linguistic approaches: Lexical, syntactic or semantic analysis is used to gain an insight into the word relations. Their usage enables for example Part of Speech (POS) tagging, which can identify to which part of speech a word belongs to. It can be used to filter out words like articles or prepositions, which don't carry useful information, while keeping the interesting parts of a sentence. They are considered complex NLP problems.

Machine Learning: This approach is often using multi-label classification algorithms for assigning keywords to a text corpus. The label or keyword set is often given in advance for the training and validation process.

Graph-based text representation: The goal is to represent a text as graph, using graphtheoretical algorithms to extract keywords. A simple approach represents the words as vertices and the co-occurrence between them as weighted edges.

Besides that there also exist other approaches, which combine the above mentioned or consider additional features, that are not directly text related, like the document layout.

In our work we combine a machine learning approach for multi-label classification with a statistical one, that uses TF-IDF for the feature vector generation used in the machine learning.

3 Methodology

In this section we discuss our approach to build the initial dataset which was used for the keyword catalogue, training, and validation of the label assignment system. Furthermore we introduce how text data is preprocessed and which algorithms are used for the evaluation.

3.1 Building the initial dataset

The dataset has to represent the research topics studied at DIT, because the extracted keywords will later be used in a matching application tailored to the local research community. Therefore existing datasets, which contain publication abstracts and keywords, will not be used. Instead we create a dataset, starting from the DIT publications as an initial seed. We increase the amount of samples using web scraping. This is a technique where the structure of a web site is analysed and then the necessary information is extracted based on that. It applies to services, where no interfaces are given or the provided ones do not offer the needed data. We are mainly interested in the abstracts as input and the keywords for training and validation, which the author assigned to them. Based on the DIT publications we also want to increase the data volume, because it is too low for further work. Despite the presence of APIs for text mining, which some online libraries such as Elsevier and Institute of Electrical and Electronics Engineers (IEEE) offer, they lack essential functionality for our work.

Table 1: Functionality of online libraries

Portal	API	Keywords	Related
CrossRef	✔	✘	✘
Elsevier	✔	✔	✘
IEEE	✔	✔	✘
SPIE	✘	✘	✘
Springer	✔	✔	✘
Wiley Online	✘	✘	✘

Table 1 gives an overview of the existence of documented APIs and their provided functions on a selected number of online libraries. None of them offers the needed functionality with keywords and related items for our approach.

While some publishers encourage the usage of their APIs for text mining, either directly or via CrossRef, which offers publication metadata cross a whole set of publishers, their focus lays often on full-text retrieval and not on keywords and related items. Because of that, we decided to write a customized web scraper. Based on a number of given document identifiers it is able to crawl through the web pages of IEEE, Elsevier, SPIE and Wiley Online. These databases display all relevant information on their publicly accessible pages and cover most of DIT publications. During that process the scraper can extract relevant data like abstracts, assigned keywords (hereafter: *labels*) and related documents. The related documents are fed back to the scraper for further extraction.

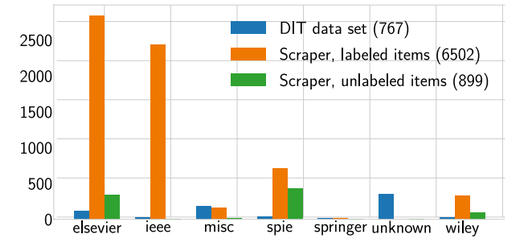


Figure 1: Sample count for DIT dataset and extracted ones by web scraper

Figure 1 shows how the sample volume of 767 items in the DIT publication database was increased by using the web scraper. Based on that we extracted 6502 abstracts with keywords (labeled items) and 899 abstracts without (unlabeled). Discovered abstracts without keywords were later used for validation.

3.2 Preprocessing

To convert a given text to a feature vector for machine learning there are some steps, which have to be done first. First, capitalization is removed by converting the whole text to lowercase.

After that, every non-alphabetic character is filtered out. Next, stop words (i.e., common words such as 'the' or 'and') are deleted from the text. Afterwards, stemming is performed on every word. This process removes suffixes from a word, so that words with the same root can be merged. As an example the words 'building' and 'builder' are converted to their word stem 'build'. Finally to calculate the feature vector TF-IDF is used. This is then used as input for the multi-label classification task.

3.3 Benchmarked algorithms

For comparison we focused on the following three transformation methods which convert a multilabel classification task to a multi-class or binary one. Sorrower [17] and Zhang et al. [22] give an overview about the currently available methods:
Binary Relevance: Each label is classified independently. A separate classifier is used for each one. Label relationships are ignored. The task is converted to a binary classification for each label. **Classifier Chains:** First the labels are ordered and for each one there is an own classifier, which takes as input the input samples related to that label, but also considers the output from the previous classifier in the chain. A problem transformation to a multi-class one is performed.
Label Powerset: The task is transformed into a multi-class problem by creating every label combination present in the training data. This breaks down the task into a multi-class one, which considers every occurring label combination.

Additionally, we consider Multilabel k-Nearest Neighbours (ML-kNN) as an interesting algorithm, which uses a k-nearest neighbour approach to assign labels to unknown samples [21].

Table 2: Algorithms used for evaluation

Method	Classifier	Abbreviation
Binary Relevance	Complement NB Linear SVC LightGBM	br_nb br_svc br_svc
Classifier Chains	Complement NB Linear SVC LightGBM	cc_nb cc_svc cc_lgbm
Label Powerset	Random Forest Passive Aggressive k-NN Vote	lp_rf lp_pa lp_knn
ML-kNN		mlknn

Table 2 gives an overview of the algorithms and the classifiers used in conjunction with the transformation methods. Complement Naive Bayes [15], Linear SVM and LightGBM were used for binary relevance and classifier chains. The latter applies a boosted decision tree (cf. Ke et al. [8]). Together with label powerset we tested random forest, passive aggressive [5] and k-NN Vote. As previously mentioned we also added ML-kNN to the candidates without a transformation method, because it is already specialised in multi-label learning. For the implementation we choose the Python programming language together with the scikitlearn [13] and scikit-multilearn [18] libraries. As toolkit for NLP preprocessing the nltk library [2] is used.

4 Evaluation

This section presents the results and the measures used for performance evaluation of the tested multilabel classification algorithms. Before that we analyze the aquired data set.

4.1 Exploring the dataset

This section explores the dataset by analyzing the label distribution. Charte et al. [3] give an overview of the existing measures for imbalanced datasets, which also are applied in this paper. The total amount of documents in the collections is described by n. L describes the set of all unique labels occurring in the dataset. Y denotes the set of all documents, while Y_i references the set of labels, which are assigned to document i.
Label Cardinality: The cardinality describes how many keywords are assigned to an input sample on average. It is defined as:

$$LCard = \frac{1}{n}\sum_{i=1}^{n}|Y_i|$$

Label Density: Density is the probability that two random chosen samples contain at least one common keyword. It is defined as:

$$LDen = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i|}{|L|}$$

Imbalace ratio per label: This one describes the imbalance of a keyword in relation to the most common one, which has an value of one. It is defined as:

$$IRLbl(y) = \frac{argmax_{y'=Y_1}^{Y_{|Y|}}(\sum_{i=1}^{|D|}h(y',Y_i))}{\sum_{i=1}^{|D|}h(y,Y_i)}$$

$$\text{with } h(y,Y_i) = \begin{cases}1 & y \in Y_i\\ 0 & y \notin Y_i\end{cases}$$

Mean Imbalance Ratio: The mean imbalance ratio is calculated as the arithmetic mean of all imbalance ratios from the whole dataset. It is defined as:

$$\frac{1}{|Y|}\sum_{y=Y_1}^{Y_{|Y|}}(IRLbl(y))$$

Figure 2 displays the Mean Imbalance Ratio, Label Cardinality and Label Density in relation to the label count. With increasing label count the cardinality and mean imbalance ratio increase, while the density decreases. This is caused by the keyword catalogue getting larger. More and more new keywords are added, which do not match the previous samples, thus creating more imbalance and reducing the probability for common keywords.

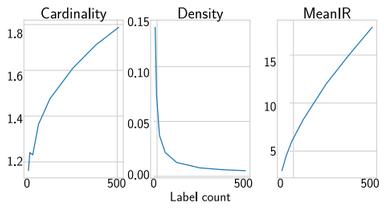


Figure 2: MeanIR, Cardinality and Density compared to label count

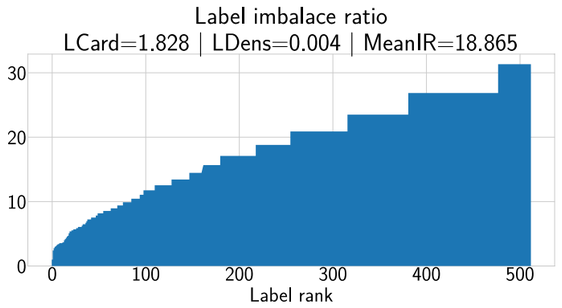


Figure 3: Label imbalance ratio at a catalogue size of 512 keywords

This means, that the keyword imbalance is not uniformly distributed. Figure 3 shows the label imbalance distribution at a catalogue size of 512 keywords. The most common one at label rank zero has a value of one. Following to that the imbalance ratio increases to a peak of around 30. This means that the least common keyword in the catalogue is around 30 times rarer in our dataset than the most common one. While at the beginning imbalance increases rapidly, later it seems to increasing linearly. This complicates a later multi-label classification task, because the higher ranked labels occur so rarely in the dataset. For example transformation methods like label powerset can not work properly because this method depends on the existence of every label combination in the training dataset, which can later occur during prediction.

The keyword imbalance strongly influences their value in classification tasks. Several keywords are so common as to infer almost no information about the paper content. The five most prominent labels in our data set are 'internet of things', 'wireless sensor networks', 'internet', 'support vector machine' and 'x-ray diffraction'.

In contrast, many keywords are related to a single paper only, thus being too specific for classification tasks. Examples for rare keywords in our data set are: 'dielectric breakdown', 'grain boundaries', 'nanomaterials', 'sims' and 'education'. In both cases, the keywords are of little value.

Figure 4: Label Co-Occurrence visualized by OpenOrd algorithm

The problem is depicted in figure 4, which shows the label co-occurrence graph visualized via the OpenOrd algorithm [10]. This was extracted from the dataset by looking at keywords which are assigned to the same abstract. Every vertex in the graph represents one keyword. Edges exist between any two keywords occurring in the same abstract. Vertices with a degree of less than two (i.e., labels which are connected to only one or no other label) have already been filtered out. The dominance of a few very densely connected keywords is quite apparent in this visualization.

4.2 Evaluating the performance of a multi-label classification task

A lot of different measures exist for evaluating the performance of an information retrieval task. In this paper we focus on *Prediction* and *Recall* (cf. Table 3), as well as the derived measures *F1-Score* and the *Jaccard score*, all of which are described in the following. In the definitions, validation labels for each sample are denoted as Y, whereas predicted labels are denoted as Z.

Table 3: Confusion matrix for precision and recall

	Condition positive	Condition negative	
Prediction positive	True positive	False positive	Recall
Prediction negative	False negative	True negative	
	Precision		

Precision ($P = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap Z_i|}{|Z_i|}$) describes whether every label in Z is correct. It is the ratio of true positives to all positive predictions.
Recall ($R = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap Z_i|}{|Y_i|}$) describes whether all correct labels are in Z. It is the ratio of true positives to all positive conditions.
F1-Score ($F_1 = \frac{1}{n}\sum_{i=1}^{n}\frac{2|Y_i \cap Z_i|}{|Y_i|+|Z_i|}$) is the harmonic mean of the precision and recall. It is also known as F-score or F-measure.
Jaccard score ($J = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|}$, also called subset accuracy) is the proportion of correctly predicted labels to the entire validation set.
For score calculation we use micro-averaging, i.e., we calculate the score for each validation sample separately and then take the arithmetic mean of them. All tests were executed for different keyword catalogue sizes, for which each contained samples with the top 8, 16, 32, 64, 128, 256 and 512 keywords of the dataset.

4.3 Results

In this section the results are discussed. For the training and validation, we used a technique named iterative stratification, which tries to maintain the label distribution between the training and validation set [16]. 80% of the labeled input samples are processed during training and 20% are reserved for validation. The unlabeled ones, extracted by the scraper, are used for testing.

Figure 5 shows the precision, recall, F1-Score and Jaccard score compared to label count. Abbreviations for the different transformation and classifier combinations are explained in

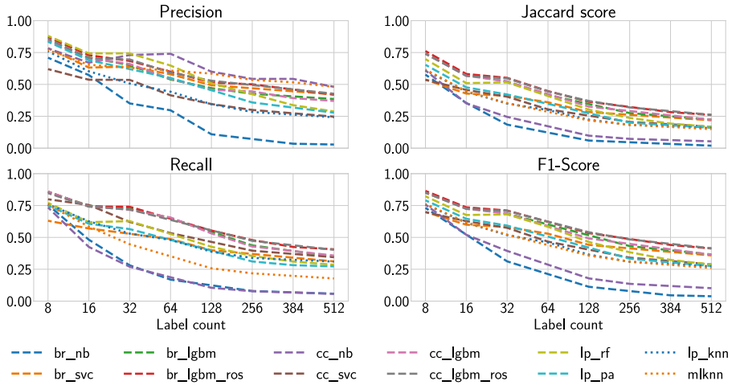


Figure 5: Metrics compared to label count. For abbreviations cf. table 2.

Table 2. If ‘ros’ is appended, Random Oversampling was additionally applied to the input samples. In general we can say, that all measures decrease with increasing label count. Furthermore following observations have been made:

Precision: Classifier chains with naive bayes and ML-kNN achieve the top scores here, while binary relevance with naive bayes falls strongly behind in comparison to the others.

Recall: LightGBM with either binary relevance or classifier chains is leading here. Naive bayes shows the worst performance regardless of the transformation method.

F1-Score and Jaccard score: The results for these measures look similar, but are shifted by around 10%. Here LightGBM is again the best performing one. Binary relevance together with linear support vector has achieved a similar result. Naive bayes falls again strongly behind.

Usage of random oversampling: For LightGBM with binary relevance or classifier chain random oversampling is performed before training. All measures improved by around 5-6%.

Assigning keywords to unseen samples: We tested our model, which uses binary relevance and LightGBM, with our two unlabeled datasets. Both of them were previously never processed during the training or validation stage. At the DIT publication dataset we were able to assign a least one keyword to 78.618% of the samples. For the unlabeled ones, which the web scraper extracted, the coverage was at 58.398%.

The LightGBM classifier in combination with either binary relevance or classifier chains got the highest F1-Score and Jaccard score. Using random oversampling before training further improved scores by around 5-6%. Thus, we attach at the end of our preprocessing stage a random oversampler. We conclude that this combination is best suited for the presented use-case.

5 Related Work

This section presents the related work in the field of keyword extraction. The focus is on supervised algorithms, where training data is required, and collection oriented approaches, where a whole set on documents is used instead of a single text.

Witten et al. [19] built a complete system for keyword extraction named Keyword Extraction Algorithm (KEA). First they identify candidates for keywords using lexical methods and rank them by a scoring scheme, which considers properties like co-occurrence in conjunction with stop words. Then they use binary relevance with naive bayes as machine learning algorithm.

Medelyan et al. [11] extend the KEA approach to Multi-purpose Automatic Topic Indexing (MAUI). Improvements include a modified candidate scoring, taking word positions inside the text into account, and the aid of dictionaries for keyword identification. In contrast, the work presented here makes use of a predefined keyword catalogue which is generated from the training dataset and evaluates multiple classification algorithms on the data.

Krapivin et al. [9] compare machine learning algorithms like support vector machines and random forest for keyphrase extraction from scientific papers. They apply NLP for further improvements. Though their work has similarities, there are some key differences: They use full text papers from Association for Computing Machinery (ACM) instead of abstracts. They do not evaluate the label imbalance. Finally, we consider sampling and gradient boosting as additional refinements.

Gelbukh et al. [6] automate the process of keyword extraction by first trying to identify candidates and then compare learning algorithms for keyphrase assignment. For candidate identification they consider different features like term frequency or word position. A multilayer perceptron and naives bayes are compared as learning algorithms. They use a much smaller dataset and label count.

Wu et al. [20] created a machine learning system specialized for keyword extraction from scientific literature. They construct an initial keyphrase database from their existing documents. For learning a least squares support vector machine is used. While their approach has similarities to ours, they use full-text and not abstracts like in our work.

Also they focus only on support vector machines and don’t consider further algorithms.

Rabby et al. [14] built an unsupervised tree based keyword extraction method for academic literature. They evaluate their algorithm by using papers from ACM. A key difference is, that they focus on full-text papers and not on abstracts.

Pay [12] presents Totally Automated Keyword Extraction (TAKE), an unsupervised algorithm focused on keyword extraction in single documents. POS tagging and filtering heuristics are used for candidate identification. While the problem setting is somewhat different, the approach suggests potential improvements for the preprocessing stage.

Chen and Xiao [4] compare the performance of term frequency, TF-IDF and Term Frequency – Keyword Activity Index (TF-KAI) for keyword extraction in the Chinese language. The TF-KAI measures keyword relevance in a specific domain and can outperform the other methods. Unfortunately, without predefined topics in the dataset their approach is not applicable for our problem.

Hasan Sazzad et al. [7] use a hybrid approach for assigning keywords to scientific documents in the Bengali language. They use an unsupervised algorithm utilizing TF-IDF and naive bayes as an supervised method, which is trained on an already tagged set of documents. The output of both is then merged to the final result. Our approach mainly differs in that we compare multiple algorithms using a larger training dataset with about 6500 items as opposed to 500.

6 Conclusion

This paper presents a keyword assignment system based on the DIT publication database. It handles low volume data and missing keywords by extending the data volume using information from online publication databases, extending the total volume to 6500 items. We compared ten multi-label learning algorithms with focus on transformation methods. The usage of either binary relevance or classifier chains in combination with LightGBM and random oversampling showed the most promising results. Compared to the related work, we were able to achieve a F1-Score of around 41% with a catalogue size of 512 distinct keywords.

The focus on abstracts eliminates the need for full-text papers. A prototype keyword assignment system was built based on these results. It uses random oversampling in preprocessing and LightGBM as classifier with binary relevance as transformation method. At least one keyword could be assigned to around 79% of internal samples and to around 58% of external samples. The prototype is integrated in a matching application in operation at DIT.

Several research directions remain for future work. For further improvements more sampling algorithms can be evaluated. Also the utilization of POS tagging in the preprocessing stage can be evaluated. Additionally, it can be tested, if label space clustering and embedding can bring further result improvements. On top of that, hyperparameter estimation for the algorithms can be performed using techniques like grid search to achieve even better scores. Furthermore it can be evaluated, if deep learning as multi-label classification algorithm can bring any significant improvements.

References **[1]** S. Beliga, A. Meštrović, and S. Martincic-Ipsic. An overview of graph-based keyword extraction methods and approaches. Information and Organizational Sciences, 39, 2015. **[2]** S. Bird, E. Klein, and E. Loper. Natural Language Processing with Python. O’Reilly Media, Inc., 1st edition, 2009. **[3]** F. Charte, A. J. Rivera, M. J. del Jesus, and F. Herrera. Addressing imbalance in multilabel classification: Measures and random resampling algorithms. Neurocomputing, 163:3– 16, 2015. **[4]** G. Chen and L. Xiao. Selecting publication keywords for domain analysis in bibliometrics: A comparison of three methods. Journal of Informetrics, 10:212–223, 2016. **[5]** K. Crammer, O. Dekel, J. Keshet, S. ShalevShwartz, and Y. Singer. Online passiveaggressive algorithms. J. Mach. Learn. Res., 7:551–585, 2006. **[6]** A. Gelbukh, Y. HaCohen-Kerner, Z. Gross, and A. Masa, editors. Automatic Extraction and Learning of Keyphrases from Scientific Articles: Computational Linguistics and Intelligent Text Processing. Springer, 2005. **[7]** K. Hasan Sazzad, M. Mumin, and M. Anwarul Azim. Keyword extraction in bangla scientific documents: A hybrid approach. 11:28–35, 2019. **[8]** G. Ke, Q. Meng, Th. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu. Lightgbm: A highly efficient gradient boosting decision tree. In Advances in Neural Information Processing Systems, pages 3146–3154, 2017. **[9]** M. Krapivin, A. Autayeu, M. Marchese, E. Blanzieri, and N. Segata. Keyphrases extraction from scientific documents: improving machine learning approaches with natural language processing. In Proc. of the role of digital libraries in a time of global change, and 12th int. conf. on Asia-Pacific digital libraries, pages 102–111. Springer, Australia, 2010. **[10]** S. Martin, W.M. Brown, R. Klavans, and K.W. Boyack. OpenOrd: an open-source toolbox for large graph layout. In Proc. Visualization and Data Analysis, volume 7868, 2011. **[11]** O. Medelyan, E. Frank, and I. H. Witten, editors. Human-competitive tagging using au- tomatic keyphrase extraction. Association for Computational Linguistics, 2009. **[12]** T. Pay. Totally automated keyword extraction. In 2016 IEEE International Conference on Big Data (Big Data), pages 3859–3863, Dec 2016. **[13]** F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in python. Journal of Machine Learning Research, 12:2825–2830, 2011. **[14]** G. Rabby, S. Azad, M. Mahmud, K. Z. Zamli, and M. Mostafizur Rahman. A flexible keyphrase extraction technique for academic literature. Procedia computer science, 135:553– 563, 2018. **[15]** J. D. M. Rennie, L. Shih, J. Teevan, and D. R. Karger. Tackling the poor assumptions of naive bayes text classifiers. In Proceedings of the Twentieth International Conference on International Conference on Machine Learning, ICML’03, pages 616–623. AAAI Press, 2003. **[16]** K. Sechidis, G. Tsoumakas, and I. Vlahavas. On the stratification of multi-label data. In Dimitrios Gunopulos, Thomas Hofmann, Donato Malerba, and Michalis Vazirgiannis, editors, Machine Learning and Knowledge Discovery in Databases, pages 145–158, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. **[17]** M. S. Sorower. A literature survey on algorithms for multi-label learning. Oregon State University, Corvallis, 18:1–25, 2010. **[18]** P. Szymański and T. Kajdanowicz. A scikitbased python environment for performing multi-label classification. ArXiv e-prints, 2017. **[19]** I. H. Witten, G. W. Paynter, E. Frank, C. Gutwin, and C. G. Nevill-Manning. Kea: Practical automated keyphrase extraction. In Design and Usability of Digital Libraries: Case Studies in the Asia Pacific, pages 129–152. IGI Global, 2005. **[20]** C. Wu, M. Marchese, J. Jiang, A. Ivanyukovich, and Y. Liang. Machine learning-based keywords extraction for scientific literature. UCS, 13:1471–1483, 2007. **[21]** M.-L. Zhang and Z.-H. Zhou. Ml-knn: A lazy learning approach to multi-label learning. Pattern Recognition, 40(7):2038–2048, 2007. **[22]** M.-L. Zhang and Z.-H. Zhou. A review on multi-label learning algorithms. IEEE transactions on knowledge and data engineering, 26(8):1819–1837, 2014.

Marco Kretschmann

Marco Kretschmann studied for a masters degree in applied computer sciences in 2019 at DIT. He is currently employed as application developer and data scientist at dab: Daten - Analysen & Beratung GmbH in Deggendorf. He is interested in developing user-friendly applications, which include artificial intelligence and allow data visualization and interactive exploration.

Andreas Fischer

Andreas Fischer Is a Professor at DIT. He received his PhD in 2017 at University of Passau. After a post-doc position in Karlstad, Sweden he was appointed as professor for Computer Science at the DIT. He is interested in the development of intelligent and autonomous networks and has conducted extensive research on network resilience, network virtualization and software-defined networks. Lately, he became interested in more general methods and applications of artificial intelligence, teaching courses on the subject at DIT.

Benedikt Elser

Benedikt Elser is a Professor at the DIT. He is also the leader of the “Applied AI” group at the Technology Campus Grafenau, a research and technology centre of the DIT. Prof. Elser received his PhD in 2012 at the Technical University of Munich, worked as a post-doc at the University of Trento, joined a Big Data Startup and finally worked at the Deutsche Bahn, a german train operator on large-scale data problems. His research interests include distributed computing, big data problems and machine learning algorithms.

Input Encodings for Character Level **Convolutional Neural Networks**

Andrei Karpau, Dr. Markus Hofmann

Convolutional neural networks (CNNs) gained much attention during the last decade. They show high efficiency in solving various kinds of problems. Among them are text mining and natural language processing tasks. This paper investigates diverse types of input representations for text CNNs. It explores various encodings of character aware networks and proposes several alternatives to the common one-ofm approach. The paper further investigates their efficiency by applying them to the sentiment analysis problem. It compares acquired metrics and evaluates the effectiveness of the proposed encodings.

1 Introduction

During the last decade, deep learning brought a lot of achievements into the area of machine learning (ML). It includes many powerful algorithms, among them are deep belief networks, recurrent neural networks and convolutional neural networks (CNNs). The latter are traditionally used for solving various kinds of image processing problems. Their architecture was inspired by the research made in the late 50s [5]. The structure of CNN reminds simple and complex cells in the primary visual cortex. First CNNs were described in the 80s when LeCun et al. [14] applied them to the recognition of handwritten zip codes. However, they became really famous during the last decade, after the series of successes in the ImageNet challenge [10, 20, 4].

Many researchers apply deep learning to text mining and natural language processing (NLP) problems. Different deep architectures such as recurrent neural networks or CNNs show high efficiency when dealing with text [13, 9, 23]. Based on the input type, text CNNs can be divided into two classes: word level and character level networks. The first ones consume text encoded by dictionaries of words whereas the second ones use symbols commonly found in alphabets which leads to processing on character level. The alphabet stores a unique vector for each letter, digit or punctuation mark. One-hot vector encoding is the most common approach. In this research we propose several alternatives. We enhance encodings by adding additional information about the types of characters such as digits, punctuation marks, upper or lower case letters. This can increase the awareness of the network and improve the accuracy. Additionally, we suggest using non-sparse encoding based on the ASCII character set. Even though they do not increase the accuracy of the models, ASCII based encodings decrease the network latency, memory consumption and computation time.

2 Related work

Many researches apply CNN to text mining tasks. Among them are sentiment analysis [8, 9, 2, 6, 22, 1, 21], topic categorization [6, 13, 23, 22, 1, 7], classification of question type [8, 9] or cybersecurity problems [18]. Generally, they follow word level, character level or hybrid approaches.

Choice of data representation is a major step in text mining. Popular techniques include bag of words or representation in vector space [12]. CNNs often consume input text encoded by dictionaries. They store vector representations of words or n-grams. These vocabularies are either created from training data or formed separately such as word2vec [6, 9]. Using this approach, the network gets knowledge about complete words, their synonyms and combinations of the two. Kalchbrenner et al. [8] propose a word-level network with several convolutional and pooling layers. This architecture works with n-grams. Due to wide convolution, networks preserve relative positions of phrases and dynamic k-max pooling layer gives the capability to combine several features of words that stay in various positions.

Kim [9] describes a CNN that works with a wordlevel input. It has convolutional, max-over-time pooling and fully connected layers. Convolution filters tend to select several words positioned near one another. The author applies a dropout technique for regularization and tries different initialization methods of which one of them randomly initializes the words and another uses pre-trained vectors from word2vec. This approach usually improves the network efficiency.

Using subword dictionaries is another way of encoding the input text. Sennrich et al. [19] propose to apply a byte pair encoding method (BPE). The technique uses subword dictionaries of predefined size and fills them with the most frequent combinations of characters. Kudo [11] introduces another subword approach that is based on the probabilities of sub-parts occurrences in a word sequence. The result dictionary contains a mixture of characters, subwords and word segmentations. These hybrid approaches help to deal with the rare and unseen words as well as improve the efficiency in multilingual use cases. The size of the dictionaries is fixed and usually contains from 8k to 90k tokens.

Having word dictionaries leads to several drawbacks. They are often very large and not efficient from a memory perspective. Word-level approaches are language specific and hardly deal with infrequent words and different forms. Naturally, a word is a combination of characters. Therefore, many researches consider characterlevel representations. Such approaches do not require knowledge about special syntactic or semantic structures. They do not need huge dictionary of vectors and work with predefined alphabets. They can effectively learn unusual combinations such as misspelled words or emoticons [23]. Additionally, character-level networks tend to be language independent and work with different languages that share the same alphabet. Wehrmann et al. [21] use character CNN for multilingual scenarios and apply it to language agnostic Twitter sentiment analysis. Their model works without machine translation, paired datasets, word-embeddings or other special techniques.

Word level encodings capture syntactic and semantic information while character level representation provide morphological and shape details. Dos Santos and Gatti [2] propose to combine both and to use them as an input for CNNs. Such hybrid network captures more information than a word-level one and thus showed better results on the Twitter dataset. Even though this type of the network considers characters, it has the usual problems for word representations. The dictionary is still huge and makes the network language specific.

Character-level input originally provides all the information about words and sentences and makes the word-level representation redundant as an alphabet can replace a word dictionary. In this case CNN depends only on characters. It can be trained for different languages. Zhang et al. [23] describe a CNN that expects input encoded on character level. They define an alphabet that consists of 70 symbols including lower case characters, digits, punctuation marks and special signs. Each item has a unique vector (1-of-m encoding) assigned to it. The technique encodes each character in the text using the predefined alphabet representing the result as a sequence of vectors. The study applies quantization in backward order which makes it easy for fully connected layers to associate weights with the latest readings [23, 22]. Final tests show that this CNN is capable to learn complex patterns in the input text. However, it depends on the size of the data and tends to work better on larger datasets. Conneau et al. [1] propose a CNN that uses similar alphabet and encoding. It is a deep network that has up to 29 convolutional layers. They are combined into blocks that contain two convolutions followed by temporal Batch-Norm layers and ReLU activations. This network uses small convolutional filters and has fewer parameters in each layer which helps to avoid typical problems that are usual for deep architectures. Tests show that extra levels improve the efficiency of the model which is especially true for very big datasets.

3 Encodings

In this research we investigate 4 types of encodings. We name the basic one ”Sparse”. It is oneof-m vector encoding. Its alphabet consists of 70 characters and is similar to the character sets used in other research such as [23, 1, 22]. It has lowercase letters, digits, punctuation marks, new line character and an empty one. This dictionary does not make any separation between upper-case and lower-case letters. It is encoded into a set of 70 vectors each of size 69 (empty character has all zeroes).

In order to increase the efficiency of the network we suggest to add additional information about the type of the character. We name this encoding ”Sparse Group”. It is still based on the one-of-m vector approach. It has the same set of values as the technique above and 4 additional bits at the end which define the type of the character. Each item is either an upper-case or a lower-case letter, a digit or a special symbol. Four additional bits increase the size of the alphabet from 70 to 96 items. This encoding provides more information about the characters and enhances the awareness of the model.

Both ”Sparse” encodings use a dictionary with most of their elements being zeros. These values do not bring much context. Meanwhile they add additional weights that consume more memory and computational resources. To deal with this problem we propose ”ASCII” encoding. It is a non-sparse encoding based on the 8bit ASCII character set and includes 128 characters that are stored into 7-bit integers

[15]. The last bit is reserved for special cases. The 95 characters are printable and contain digits, lower and upper case Latin letters and punctuation symbols.

Except for ASCII, there are two other popular character sets: Windows-1252 and Unicode. They both are a superset of ASCII and encode more characters. Meanwhile, the first 128 characters are the same. This paper uses a dataset that contains reviews written in English, therefore, ASCII is sufficient to encode most of the characters. Since Unicode and Windows-1252 are ASCII supersets they do not bring any valuable contributions to this project. Nevertheless, these encodings can still bring improvements when using them on multilingual datasets.

The last character set is named ”ASCII Group”. It is ”ASCII” encoding with 4 additional bits for group separation. Groups do not add any new symbols into the ”ASCII Group” alphabet. However, they still increase the awareness of the network about the character types.

4 Research objectives

This research compares the efficiency and performance of character level CNNs that are served by various input encodings. The study proposes 4 different options as outlined in the table 1. Along with the name of each encoding it provides the number of characters in each character set (alphabet size), the length of the encoding vectors and the examples, which include the encoding of lower case letter, upper case letter, punctuation mark and digit.

In this research we empirically compare the proposed encodings using the predefined dataset. We train a separate network for each input type. All CNNs have similar architecture and differ only by the size of the first convolutional layer. This is caused by the length of the vectors provided by each alphabet. For example ”Sparse” encoding has vectors of size 69. So the first convolutional layer contains 128 filters with 69X5 elements. The length of ”ASCII” vectors is 8. Thus, the first layer has 128 filters of size 8X5. This difference makes the networks faster and decreases the memory consumption. However, the number of coefficients effects the final accuracy.

5 Dataset

To compare the encodings, we use a subset of the Amazon Reviews dataset. We train the networks to provide sentiment analysis. Originally, the dataset contains more than 142 million reviews obtained during 18 years [17]. Most of the reviews are not very long and therefore do not require wide input layers. All the items contain some metadata such as reviewer id, name or date. Each review has a corresponding rate from 1 to 5 which can be used for classification. Data are stored in JSON format, are human readable and can be easily parsed during the computation.

Diverse studies [23, 22, 1] use this dataset to train text CNNs. They consider two classification tasks: rating and sentiment polarity prediction. The rate classification usually shows low results due to the task complexity and narrow decision boundaries whereas polarity models tend to show high performance. This paper focuses on both problems.

We are using a 5-core subset from the Amazon review dataset. 5-core means that each of the users and items have at least 5 reviews. This helps to filter out the reviews that are potentially not relevant. The 5-core dataset is smaller than the original one, however, it is still very large and contains approximately 40 million reviews.

We further narrow down the data by filtering certain categories from the 5-core subset. They are: Beauty, Cell Phones and Accessories, Grocery and Gourmet Food, Pet Supplies, Toys and Games. The original data are highly imbalanced as the number of reviews of each rate is different. This can badly influence both rate and sentiment polarity prediction [16]. Therefore, during the preprocessing stage we select a balanced subsample using an undersampling technique. Later, we divide the dataset into training and testing subsets using the stratified sampling method. The training subset has 122,358 items and the test set includes 52,442 reviews. We also carried out some additional text preprocessing such as converting the summary to upper-case, combining it with the review text and separating them by the new line character. Due to its structure, CNN expects an input of fixed size. Thus, the preprocessing script makes all reviews 1,024 characters wide. It either prunes the long reviews or adds empty characters to the short ones.

Table 1: Types of encodings used in the research

Encoding	Size of the Alphabet	Vector Length	Encoding Examples
Sparse	70	69	a: [1,0,0,0,0,...,0,0] A: no upper case ,: [0,0,0,0,...,1,...,0] 1: [0,0,0,0,...,1,...,0]
Sparse Group	96	73	a: [1,0,...,0,0,1,0,0,0] A: [1, 0 ..., 0, 0, 1, 1, 0, 0] ,: [0,...,1,...,0,0,0,1,0] 1: [0,...,1,...,0,0,0,0,1]
ASCII	101	8	a: [0,1,1,0,0,0,0,1] A: [0, 1, 0, 0, 0, 0, 0, 1] ,: [0,0,1,0,1,1,0,0] 1: [0,0,1,1,0,0,0,1]
ASCII Group	101	12	a: [0,1,1,0,0,0,0,1,1,0,0,0] A: [0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0] ,: [0,0,1,0,1,1,0,0,0,0,0,1] 1: [0,0,1,1,0,0,0,1,0,0,1,0]

6 Architecture of Character CNN

Different research projects use character CNNs of diverse size. For example, Zhang and LeCun [22] propose a network with 6 convolutional and 3 fully connected layers. A year later, Conneau et al. [1] build a deep CNN which contained 29 convolutional layers. When trained properly, deeper networks achieve higher results but in parallel, consume a lot of resources.

In this research we build a separate CNN for each input encoding. The structure of the networks differs only by the size of the first layer aiming to allow a fair comparison between them.

The proposed CNN has 3 convolutional blocks and 3 fully connected ones (see Figure 1) which is sufficient to recognize complicated text patterns, while not consuming too many resources. Each convolutional block has a convolutional layer followed by ReLU and Max Pooling. The last layer of the network returns the predicted rate, which is a float value. Later, this value is used to solve both rate and polarity classification tasks, by building the ROC curves and determining the thresholds.

1. Rate classification: the values are mapped to rates from 1 to 5 using the calculated thresholds.
2. Polarity of the sentiment: the values are mapped to the negative and positive categories using the thresholds. Their calculation is based on assumption that rates 1, 2 are negative and 4, 5 are positive. For this task we ignore the reviews with rate 3.

Neural networks often suffer from overfitting. This problem usually occurs for fully connected layers which tend to have major number of weights. Convolutional layers have less coefficients, thus, dropout is not very effective for them [3]. For this reason, the proposed CNN has a dropout unit before each fully connected block.

To train the network, we use Tensorflow framework as it provides main building blocks such as convolutions, rectified linear units, pooling operators or optimizers. Additionally, Tensorflow allows to run the network on GPU devices. We use SAP Leonardo Training Service to run our experiments. This service works in the cloud and automatically configures machines with 1 Tesla K80 GPU, 2 CPUs and 32 GB RAM for each training and evaluation job. It helps to speed up the whole process, by running several jobs in parallel, while keeping the environment consistent.

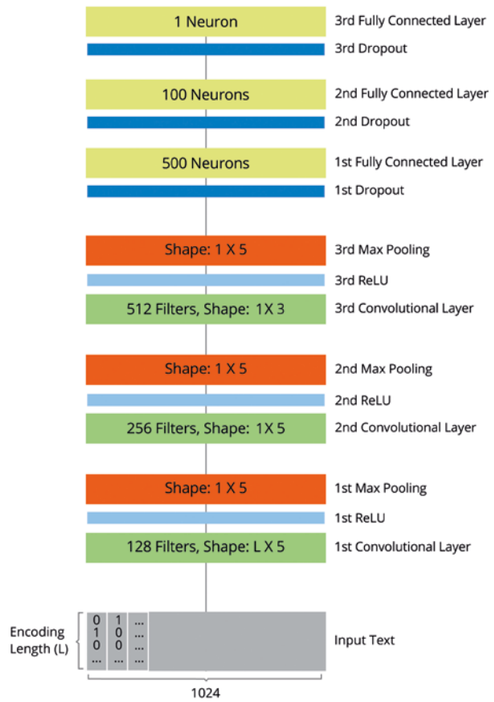


Figure 1: The structure of the character CNN used in the research

7 Evaluating results

7.1 Training

During the training we track the change of the root mean square error (see Figure 2). The models need different number of epochs to converge.

Table 2: Training metrics

Input encoding	Training time	Epochs
Sparse encoding	2 days, 5:03:22	150
Sparse Group	2 days, 16:36:42	175
ASCII encoding	23:27:09	288
ASCII Group	1 day, 6:51:54	345

”Sparse” and ”Sparse Group” encodings require from 150 to 175 epochs. The fluctuation during the training is generally not very high. Although ”Sparse” encodings have some sharp peaks, they quickly decrease back. The training of these two models needs less fine tuning of hyperparameters and fewer epochs. However, it runs slower and takes more time to converge.

The RMSEs of ”ASCII” and ”ASCII Group” are less stable at the beginning. However, the average error still decreases. They require more training epochs and each epoch takes less time compared to the networks with a sparse input layer. Therefore, the training of ”ASCII” and ”ASCII Group” is approximately two times faster. Meanwhile, the training of the ”Sparse” models goes smoother. Their input layers are not very sensitive and cause less fluctuations.

7.2 Validation

The validation of the trained models includes two steps: prediction of rate values and classification. The first step is done in the same cloud environment. It calculates the predicted value for each sample from the test dataset and computes the RMSE. The classification step includes building ROC curves and finding the threshold points for both rate and sentiment polarity cases. Table 3 provides the prediction and classification results. ”Sparse” encodings show lower RMSE and higher accuracy in both multi-class and polarity cases. Meanwhile, ”ASCII” CNNs consume less resources and are generally up to 5 times faster.

7.3 Statistical significance

Group encodings show slightly lower RMSE and higher accuracy. However, the differences between values are very low. A paired samples T-Test helps to verify the statistical significance of the values. We randomly pick samples of size 5,000 from the testing dataset. Then, we calculate the mean squared errors (MSE) of each sample. We run T-Test for pairs ”Sparse” ”Sparse Group” and ”ASCII” ”ASCII Group”. The following list displays its results.

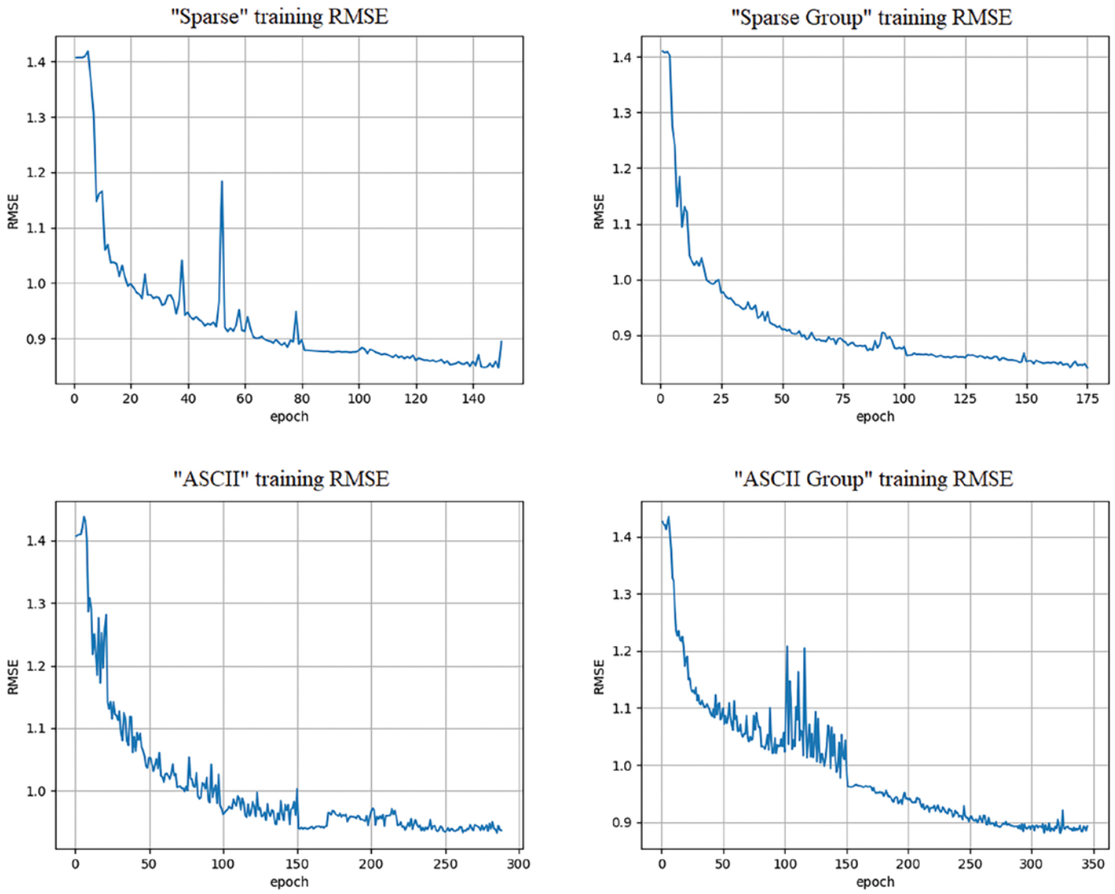


Figure 2: The reduction of the root mean square error during the training

- T-Test results for ”Sparse” and ”Sparse Group” models
  - t = 70.2202
  - p = 2.02824e-24
  - mean MSE (Sparse) = 0.74616
  - mean MSE (Sparse Group) = 0.73119
- T-Test results for ”ASCII” and ”ASCII Group” models
  - t = 84.35381
  - p = 6.28845e-26
  - mean MSE (ASCII) = 0.88632
  - mean MSE (ASCII Group) = 0.83072

P-values of both pairs are very low indicating that the differences between the means of the distributions are statistically significant. The mean values of MSE distributions of the group models are lower than of the non-group ones. Since square root is a monotonic function, this logic can be extended to the RMSE case. The T-Test proves that group models perform better.

Table 3: Validation results

Input encoding	Validation Time	RMSE	Accuracy (Rate)	Accuracy (Polarity)
Sparse encoding	4min	0.865	0.482	0.909
Sparse Group	4min 27sec	0.859	0.488	0.910
ASCII encoding	50 sec	0.945	0.446	0.880
ASCII Group	55 sec	0.916	0.460	0.893

8 Conclusion

This paper reports on analyses of input encodings of character level CNNs. It considers four different options. They are ”Sparse” one-of-m vector, ”Sparse Group”, ”ASCII” and ”ASCII Group” encodings. This study trains a separate prediction model for each input type before computing metrics and comparing these. The analysis of the proposed encodings shows that:

1. Both ”Sparse” models provide lower prediction and classification errors than ”ASCII”.
2. Group models achieve better prediction and classification rates than non-group ones.
   Even though the values do not differ much, statistical tests show their significance.
3. ”ASCII” encodings provide vast improvements in training and testing time. They need less computations, which makes them up to 5 times faster.
4. ”ASCII” models need more training epochs and more fine tuning. Meanwhile, a single epoch needs less computations and therefore, ”ASCII” CNNs require less time to converge.

Generally, ”Sparse” models show better results than ”ASCII” even though the difference is not huge. At the same time, ”ASCII” encodings are more efficient in terms of time and computations. In some cases a vast improvement in speed can be more important than higher prediction rates. It can decrease the business costs both during the training and inference, by consuming less memory and making less CPU/GPU computations. The latency of the ”ASCII” based models is up to 5 times lower. This can be crucial, when running them in the real time applications or when having high workloads. Finally, ASCII character set is one of the most well known in computer science. Such encodings require less programming effort and can be easier adopted by the industry.

References: **[1]** Alexis Conneau, Holger Schwenk, Lo¨ıc Barrault, and Yann Lecun. Very deep convolutional networks for natural language processing. arXiv preprint arXiv:1606.01781, 2016. **[2]** C´icero Nogueira Dos Santos and Maira Gatti. Deep convolutional neural networks for sentiment analysis of short texts. In COLING, pages 69–78, 2014. **[3]** Yarin Gal and Zoubin Ghahramani. Bayesian convolutional neural networks with bernoulli approximate variational inference. preprint arXiv:1506.02158, 2015. arXiv **[4]** Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016. **[5]** David H Hubel and Torsten N Wiesel. Receptive fields of single neurones in the cat’s striate cortex. The Journal of physiology, 148(3):574–591, 1959. **[6]** Rie Johnson and Tong Zhang. Effective use of word order for text categorization with convolutional neural networks. arXiv preprint arXiv:1412.1058, 2014. **[7]** Rie Johnson and Tong Zhang. Deep pyramid convolutional neural networks for text categorization. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), volume 1, pages 562–570, 2017. **[8]** Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. arXiv preprint arXiv:1404.2188, 2014. **[9]** Yoon Kim. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882, 2014. **[10]** Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages 1097–1105, 2012. **[11]** Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. arXiv preprint arXiv:1804.10959, 2018. **[12]** Lokesh Kumar and Parul Kalra Bhatia. Text mining: Concepts, process and applications. Journal of Global Research in Computer Science, 4(3):36–39, 2013. **[13]** Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. Recurrent convolutional neural networks for text classification. In AAAI, volume 333, pages 2267–2273, 2015. **[14]** Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. Neural computation, 1(4):541–551, 1989. **[15]** Charles E Mackenzie. Coded-Character Sets: History and Development. Addison-Wesley Longman Publishing Co., Inc., 1980. **[16]** David Masko and Paulina Hensman. The impact of imbalanced training data for convolutional neural networks, 2015. **[17]** Julian McAuley. Amazon review data. http://jmcauley.ucsd.edu/data/amazon/, 2018. (Accessed on 06/05/2018). **[18]** Joshua Saxe and Konstantin Berlin. expose: A character-level convolutional neural network with embeddings for detecting malicious urls, file paths and registry keys. arXiv preprint arXiv:1702.08568, 2017. **[19]** Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. arXiv preprint arXiv:1508.07909, 2015. **[20]** Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1–9, 2015. **[21]** Joonatas Wehrmann, Willian Becker, Henry EL Cagnini, and Rodrigo C Barros. A character-based convolutional neural network for language-agnostic twitter sentiment analysis. In Neural Networks (IJCNN), 2017 International Joint Conference on, pages 2384–2391. IEEE, 2017. **[22]** Xiang Zhang and Yann LeCun. Text understanding from scratch. arXiv preprint arXiv:1502.01710, 2015. **[23]** Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In Advances in neural information processing systems, pages 649– 657, 2015.

Andrei Karpau
Andrei Karpau is a senior software engineer at Ma- chine Learning Foundation, SAP SE. He actively participates in development of machine learning based projects and helps to adapt them to real customer use cases. Andrei holds a master’s de- gree in data science. His research interests involve applied data science and the performance of ma- chine learning algorithms.

Dr. Markus Hofmann
Dr. Markus Hofmann is a senior lecturer at the Technological University Dublin, Ireland. His main research interests lie in text mining and ap- plied data science. He has published extensively internationally at conference and journal level. Dr. Hofmann is also a key member of a fully online global Masters programme in Applied Analytics and Data Science.

Unsupervised, Non-centralized, Upstream Fusion of Multiple Modalities Using **Deep Directional-Unit Networks**

Denis Garagić, Bradley J. Rhodes

This paper presents an end-to-end spatiotemporal processing pipeline that uses a novel application of dynamic deep generative neural networks for fusing 'raw' and / or feature-level multi-modal and multi-sensor data. This pipeline exploits the learned joint features to perform detection, tracking, and classification of multiple elements of interest (EOI) event signatures. Our deep generative learning framework is composed of Conditional Multimodal Deep Directional-unit Networks that extend deep generative network models to enable a general equivariance learning framework with vector-valued visible and hidden units called directional units (DUs). These DUs explicitly represent sensing state (sensing / not sensing) for each modality and environmental context measurements. Direction within a DU indicates whether a feature (within the feature space) is present and the magnitude measures how strongly that feature is present. In this manner, DUs concisely represent a space of features. Furthermore, we introduce a dynamic temporal component to encoding the visible and hidden layers. This component facilitates spatiotemporal multimodal learning tasks including multimodal fusion, cross-modality learning, and shared representation learning, as well as detection, tracking, and classification of multiple known and unknown EOI classes in an unsupervised and/or semi-supervised way. This approach overcomes the inadequacy of pre-defined features as a means for creating efficient, discriminating, low-dimensional representations from high-dimensional multi-modality sensor data collected under difficult, dynamic sensing conditions. This paper presents results that demonstrate our approach enables accurate, real-time target detection, tracking, and recognition of known and unknown moving or stationary objects or events and their activities evolving over space and time.

1. Introduction

Many challenges confront continuous detection and characterization of known and unknown moving or stationary targets or events involving their activities evolving over space and time from multimodal data collected by a single moving sensor or a team of small unmanned air systems (SUAS). In additions to normal properties of real data, these include limited sensor resolution (often exacerbated as a function of platform standoff distance), clutter in dense target environments, unfavourable environmental conditions, attempts to deceive, mask, or otherwise confuse target or activity existence (e.g., via decoys and / or confusers). Fusing data from multiple sensing modalities helps address such challenges. Often, approaches make detection / characterization decisions for each modality, then try to fuse those decisions. Our prior work has shown that upstream fusion of 'raw' sensor data outperforms downstream (post-decision) fusion [1,2,3] and that fusion is superior to processing of any single modality in isolation. However, fusion alone is insufficient, especially when considering (or requiring) distributed processing.

Compression is important here, but the benefits of upstream fusion suggest that compression should also be upstream (rather than the result of single modality / single platform decisions). Taking full advantage of the structural information embedded within single- and multi-modality data is critical for inferring compact representations with invariant properties to facilitate concise information sharing as well as improving single platform performance. Multiple factors demand that adaptivity be a central feature of a viable approach. These include the interest in unknown or unanticipated targets, the desire to fuse rapidly between different combinations of sensor modalities, the need to transfer information between platforms that host different sensors, and the inadequacy of pre-defined features as a means for creating efficient, discriminating, low-dimensional representations from high-dimensional multi-modality sensor data collected under difficult, dynamic sensing conditions.

Traditional approaches centered on a cascade of signal processing tasks to detect elements of interest (EOIs) within locations/regions of interest (ROIs), followed by temporal tracking and supervised classification of these EOIs over a sequence of observations, is not able to optimally exploit inter-modal characteristics (e.g., spatio-temporal features that co-vary across modalities) of EOI signatures. This paper presents an end-to-end spatiotemporal processing pipeline that uses a novel application of dynamic deep generative neural networks for fusing 'raw' and / or feature-level multi-modal and multi-sensor data. This pipeline exploits the learned compact feature representations fused over multiple sensing modalities, Figure 1, to perform detection, tracking, and classification of multiple EOI event signatures. Our deep generative learning framework is composed of Conditional Multimodal Deep Directional-unit Networks (D3N) that extend deep generative network models [9,10,11] to enable a general equivariance learning framework with vector-valued visible and hidden units called directional units (DUs). These DUs explicitly represent sensing state (sensing / not sensing) for each modality and environmental context measurements. Direction within a DU indicates whether a feature (within the feature space) is present and the magnitude measures how strongly that feature is present. In this manner, DUs concisely represent a space of features. Furthermore, we introduce a dynamic temporal component to encoding the visible and hidden layers. This component facilitates spatiotemporal multimodal learning tasks including multimodal fusion, crossmodality learning, and shared representation learning, as well as detection, tracking, and classification of multiple known and unknown EOI classes in an unsupervised and/or semi-supervised way. This approach overcomes the inadequacy of pre-defined features as a means for creating efficient, discriminating, lowdimensional representations from high-dimensional multi-modality sensor data collected under difficult, dynamic sensing conditions. This paper presents results that demonstrate our approach enables accurate, real-time target detection, tracking, and recognition of known and unknown moving or stationary targets or events and their activities evolving over space and time.

2. Technical Approach

Online unsupervised learning deals with the problem of finding structure in unlabeled streaming data. This approach to learning is flexible when it comes to learning general purpose representations that can enable a wide variety of tasks and adapt quickly to new problems without depending on human supervision. Therefore, there are strong arguments for learning structure in data in an unsupervised way.

The world around us can be perceived and represented in many different ways. For example, the task of localizing, tracking, and identifying multiple moving vehicles with multiple distinct emitting communication devices within an area of interest can be accomplished by using simultaneous inputs from a full motion video of a scene containing these vehicles and observed by an airborne platform, scanned radio frequency (RF) spectrum within the area of interest obtained by multiple passive electromagnetic sensors, observations from acoustic and / or seismic devices, or by observations obtained from a set of active electromagnetic sensors (i.e., Radio Detection and Ranging (RADAR)). These modalities capture complementary as well as shared properties of objects. Together, these multimodal data offers a more complete and rich description of the object compared to any single modality. Real-world systems that rely on sensing their environments are often built with multiple and diverse sensors for redundancy, as well as to provide complementary information. Given the ubiquity and practicality of multimodal

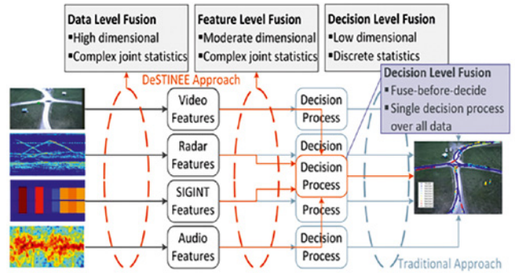


Figure 1: We learn compact upstream feature representation over multiple sensing modalities to discriminate target classes & activities.

data, it is imperative to design unsupervised learning models that can work with and exploit the structure in multimodal data (e.g., how do we exploit the fact that one modality might be somewhat invariant to large changes in another modality).

This provides a rich learning signal for learning representations in terms of latent variables (e.g., type of target classes). In the context of machine learning approaches such as Deep Generative Models [4,5,9], these latent variables (or hidden unit activations) have traditionally been binary or real-valued. While there has been a lot of work done in building different machine learning architectures and optimizing different loss functions, neural activations have typically been thought of as scalar-valued quantities.

In this paper we explore the use of vector-valued distributed hidden units (directional units) within the architecture of dynamic deep generative models [9,12], which we call an autoencoder based Dynamic Deep Directional-unit Networks (D3N). The autoencoder-based **Dynamic Deep Directional-unit Network** (D3N, Figure 2) learns abstract feature representations over multiple hidden layers from multimodal spatiotemporal input data and distributed hidden states efficiently model complex time series enabling this architecture to become an end-to-end spatiotemporal processing pipeline for detection, tracking and classification of multiple moving and stationary objects. Here we apply this end-to-end spatiotemporal processing pipeline for detection, tracking and classification to discriminate multiple moving targets with multiple emitting communication devices from moving targets without emitting communication devices using two sensing modalities (a raw full motion video data and observed communication signal data – raw I/Q data) as a demonstration example.

3. Deep Generative Adaptive Multimodal Fusion

In principle, any general-purpose unsupervised learning technique can simply operate on the concatenation of multiple modalities without bothering to deal with each input channel separately. However, each modality is often characterized by very distinct statistical properties. Having very different statistical properties makes it much harder to discover relationships across modalities than relationships among features in the same modality. There is a lot of structure in the data but it is difficult to discover the highly non-linear relationships that exist between low-level features across different modalities. The ability of deep neural networks to learn successively higher level abstract

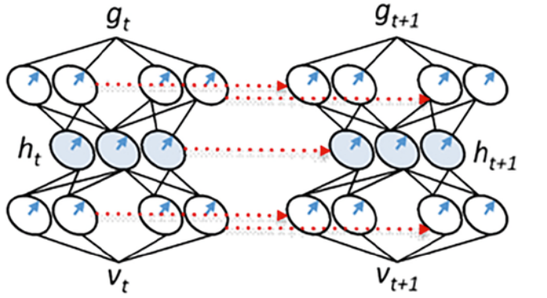


Figure 2: Our Dynamic Deep Directional-unit Network (D4N) derives a compact vector-valued encoding-vector ("hidden" layer h) of the multimodal input signals ("visible" layer v). It maintains accuracy during learning by decoding layer h & reconstructing v at the top ("generative" layer g). In practice, our D3Ns have a sequence of layers at increasing compression ratios leading to layer h. The D3Ns also captures signal state transitions at time t by propagating causal input [10] (dotted red arrows) to its evolution at time t+1.

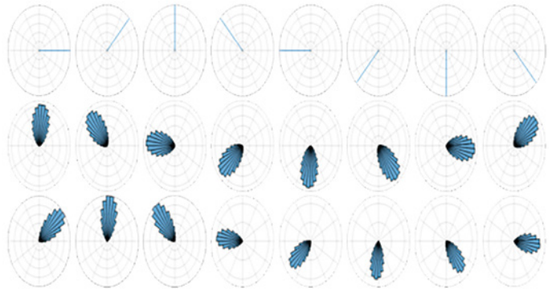


Figure 3: A hidden unit's pose corresponds to the orientation of features in the visible layer. A simple DU-RBM with 2 visible nodes and 1 hidden node; all nodes are 2-D directional units. Rotating the pose of the hidden unit (first row) drives corresponding rotations of the visible unit conditional distributions (second and third rows).

distributed representations makes them a powerful tool in this regard [9].

We employ an autoencoder-based **Dynamic Deep Directional-unit Network** (D3N, Figure 2) to learn compact, abstract feature representations from high-dimensional multi-modal spatiotemporal data composed of simultaneous full motion video data and complex-valued passive RF signals data. This will be a generalization of the Directional-unit Boltzman Machine (DuBM) model proposed by Zemel *et al.* (1992) [10,11]. The visible layer of this network ingests feature values integrated over multiple scales of discrete space/time windows from multiple-sensor platform modalities (e.g., RF, EO, IR, radar). In our future research, we plan to extend this approach to distributed multi-platform environments for learning representations in terms of latent variables (e.g., type of target classes), as well as add additional modalities to the existing mix, including acoustic, seismic and RADAR data.

In a typical neural net, each neuron can be thought of as a detector for a particular feature, and its activation is a measure of how strongly that feature is detected. On the other hand, in the case of directional units, each unit is associated with a space of features. Its activation $\boldsymbol{h} \in \mathbf{R}^N$ is a product of a unit norm direction $\widehat{\boldsymbol{h}}$ and a non-negative scalar magnitude $\|\boldsymbol{h}\|$. The direction indicates whether a feature (within the feature space) is present and the magnitude measures how strongly that feature is present. In other words, each directional unit has a coordinate frame associated with it. This coordinate frame is laid out on the surface of a unit sphere in $\mathbf{R}^N$. In this manner, directional units can concisely represent a space of features. Each point in this coordinate frame corresponds to a "pose" that features in this space can take; this provides a general equivariance learning framework over distributed platforms. For example, we can think of a directional unit that detects edges in a region of the visual field and encodes the precise position of the edge within this region through its direction (Figure 3 provides examples of such units). If the edge moves slightly, the pose rotates accordingly. Therefore, small changes in the position or orientation of the edge lead to an equivariant change in the pose of the feature. However, the direction of the feature is invariant to the precise location of the feature and only indicates its presence. In this manner, directional units can concisely represent a space of features. They can change their pose smoothly in response to change in the pose of their input. A network of directional units can be "locked" in terms of relative poses. Any global change in the input's pose can be simply transferred equivariantly throughout the network. The potential to provide a general equivariance learning framework is a strong motivation for this work.

We also introduce a dynamic temporal component to encoding [12] the visible layer designed for spatiotemporal behaviors and a multi-scale extension to characterize activity and sensor modalities with different temporal dynamics. The i^{th} platform's *local processing D3N* (Figure 2) defines a joint probability distribution, $p_t^i(\boldsymbol{v}_t, \boldsymbol{h}_t | \boldsymbol{v}_{t-K:t-1})$, over an N-dimensional vector of current multimodal measurements (visible units), $\mathbf{v}t$, and a collection of N-dimensional vector of latent variables, $\mathbf{h}t$, conditional on past K observations. The trained i^{th} platform D3N, $p_t^i(\boldsymbol{v}_t, \boldsymbol{h}_t | \boldsymbol{v}_{t-K:t-1})$, represents the *transition density* – the nonlinear prior accounting for the complex target dynamics and feature representations capturing correlations across different sensing modalities. Such powerful priors facilitate efficient inference (filtering and prediction) over target behaviors in the presence of noisy observations, missing observations, and inherent ambiguities.

By defining a joint probability distribution over multimodal spatiotemporal data and binary latent features given a past history of object observations, they fit naturally into the Bayesian inference(filtering and prediction) framework, thus enabling us to perform joint detection, tracking and classification using sequential Monte Carlo inference (aka Particle filter inference). In addition, our deep generative learning framework captures temporal dependencies, which makes it capable of automatically learning multimodal joint feature representations and modelling the temporal dependencies between their activation.

Since we only need the prior to draw samples from the predictive distribution, the unnormalized and online nature of the D3N makes it good fit for, e.g., probabilistic online multi-target detection, tracking (filtering), and classification. In addition, the D3N is capable of reconstructing missing modalities (e.g., EO signatures) given the observed ones (e.g., SAR signatures) as shown in Figure 4.

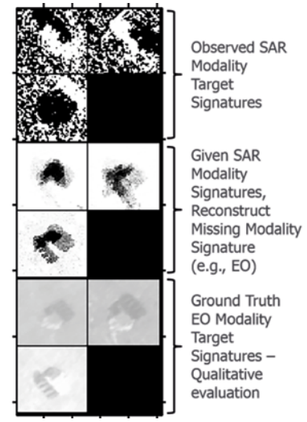


Figure 4: Sensory reconstruction of missing / absent modality driven by input from available modality in testing / transfer setting. This is akin to a mind's eye experience from perspective of a selected sensor.

The generative D3N model decodes the abstract features and fills in gaps in single modalities by projecting primitives to real world features (e.g., location, time, velocity, cardinality) for input to multi-target detection, tracking and classification. Since D3Ns belong to a family of Deep Generative Neural Networks [9] that contain many layers of hidden variables, we can estimate their partition function (the probability of data under the model is known only up to a normalizing constant, known as the partition function), which is helpful for determining optimal model performance and for controlling model complexity. One benefit of this approach is transfer of useful features from previously learned objects to novel ones; another is lower computational complexity relative to 'shallow' methods. D3N benefits from a novel particle filtering inference algorithm for posterior estimation that yields fast scalable model learning. This also mitigates the significant training demands of traditional deep learning approaches. It is a parallelizable 'online' algorithm that processes each detection only once, in sequence.

4. Decentralized Supervised Discrimination

Decentralized Supervised Discrimination uses compact abstract features from the Dynamic Deep Directional-unit Network (D3N, Figure 5) as primitives to efficiently learn and match complex, multi-part, dynamic signatures for targets, behaviors, and clutter phenomena. Decentralized Discrimination operates in supervised and unsupervised modes to learn models for both known (labeled) and unknown (new) targets, behaviors, and phenomena. Decentralized Discrimination applies Bayesian Program Learning (BPL) [6] to capture human-like learning abilities, which are relevant for challenging AFRL data conditions. We adapt the BPL framework from learning of handwritten characters from as few as one exemplar, to learning and matching of targets, composite track behaviors and signature associations for multimodal multi-sensor phenomena as transferable "concepts" called signature programs. Signature programs are built in a probabilistic framework with probabilistic generative models expressed as structured procedures to combine primitives and composites of primitives in an abstract description language [7]. Thus, BPL enables learning distributions of feature primitives over training conditions to understand how signatures vary as a function of the sources of variance (e.g., SUAS pose, sensing modality activation, and environmental conditions), and enables learning programs for a new signature by generalizing already learned probabilistic distributions at each level. Additionally, we make the extension to a Hierarchical BPL (HBPL) framework that learns shared structure within groups of signatures, e.g., for targets of the same class. This breaks classification into subsets, rather than trying to distinguish 1,000s of targets from one another without understanding shared structure. Figure 5 illustrates an HBPL example with 2 dimensional directional unit vectors of inferred multimodal abstract feature *primitives* (phase and magnitude shown as an arrow) from each D3N time step

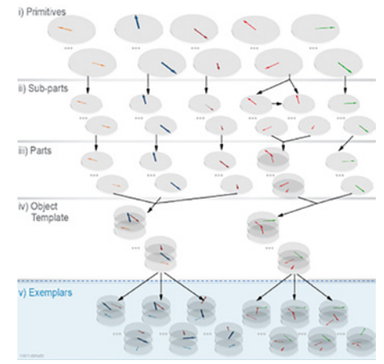


Figure 5: Multi-Modal multiple emitter tracking & localization ingests raw video sensor feeds from 1 sensor (K frames, H x W pixels per frame) and raw passive RF (PRF) data from 3 sensors (X samples per sensor between consecutive video frames).

(denoted by a distinct color) are used as *sub-parts* to constructively build more complete *parts*. These parts can be combined with other parts or themselves with a *relation* (e.g., repeat, attached at start, or co-centered) to form a *signature template*. This signature template is the complex signature for this behavior / phenomena, which is then sampled to understand and model sources of variation (*exemplars*) and generate probabilistic matches to *observations*.

5. Decentralized Supervised Discrimination

This section describes initial processing of multi-modal data using the multimodal D3N-HBPL based feature projection, multi-modal association and tracking, and classification components with the goal to localize and track multiple moving communication devices (i.e., emitters). This description covers the full motion video (FMV) and Passive RF (PRF) data modalities that the multimodal DBM ingests (Motion and Image Features

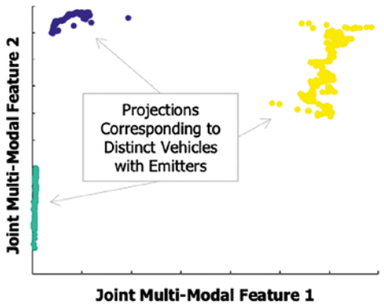


Figure 6: The multimodal D3N efficiently factorizes different variations present in multi-modal data, discriminating between moving vehicles with multiple emitters. Composition of non-linear processing stages transforms multimodal data into higher-level – more abstract – representations at each layer.

Figure 7: Our approach correctly associates the emitter to the moving vehicle carrying it based on the projections shown in Figure 6. Projection (Figure 6) and track (Figure 7) colors map between figures (e.g., yellow to yellow).

and PRF Features respectively) in order to localize and track all emitters of interest and provides intial observations for discriminative tasks such as classifying types of distinct emitters in the scene of interest. We operate at the feature level fusion process depicted in Figure 1. Our ongoing research is developing several configurations based on the multimodal D3N that operates at the data level fusion, e.g., utilize upstream fusion of heterogeneous sensing data (Figure 1).

Figure 6 shows the learned joint probability measure of multimodal data by our multimodal D3N approach on a use-case scenario involving three moving vehicles carrying three distinct emitters. Clearly three distinct classes of vehicles are present in the scene. Tracks are accurately assigned to distinct clusters over time (Figure 7). Supervised training is needed in order to identify type of the emitter.

ACKNOWLEDGEMENTS

The authors thank Air Force Research Laboratory’s Information Directorate for funding this project. AFRL Contract #: FA8750-16-C-0071. DISTRIBUTION A. Approved for public release: distribution unlimited. Case Number: 88ABW-2018-5796.

References: **Garagić, D., Peskoe, J., Liu, F., Cuevas, M., & Rhodes, B.J. (2014).** Long-range dismount activity classification: LODAC. In Proceedings of the SPIE. 9079, Baltimore, USA, May 5-8, 2014. **Garagić , D., et al.,** “Upstream Fusion of Multiple Sensing Modalities using Machine Learning and Topological Analysis,” to be presented at the AeroConf 2018 IEEE Aerospace Conference - Multimodal Fusion Track, March (2018). **Hinton, G.E. & Salakhutdinov, R. (2006).** Reducing the dimensionality of data with neural networks. Science, 313(5786), pp. 504-507. **Salakhutdinov, R. & Hinton, G.E. (2009).** Deep Boltzmann Machines. AISTATS’2009, pp. 448-455. **Lake, B., Salakhutdinov, R., Tenenbaum, J.,** “Human-level concept learning through probabilistic program induction,” SCIENCE, vol. 350 issue 6266, pp 1332-1338, (2015). **Goodman, N. D., Tenenbaum, J. B., & Gerstenberg, T. (2015).** Concepts: New Directions. E. Margolis, S. Laurence (Eds.), MIT Press, Cambridge, MA. **Ghahramani, Z. (2015).** Probabilistic machine learning and artificial intelligence. Nature 521, pp. 452–459. doi:10.1038/nature14541 pmid:26017444. **Zemel, R.S., Williams, C.K.I., & Mozer, M.C. (1995).** Lending direction to neural networks. Neural Networks, 8:503-512. **Zemel, R.S., Williams, C.K.I., & Mozer, M.C. (1992).** Directional-unit Boltzmann machines. In Advances in Neural Information Processing Systems, 5, 172-179. **Häusler, C., Susemihl, A., Nawrot, M. P., & Opper, M. (2013).** Temporal autoencoding improves generative models of time series. arXiv preprint arXiv:1309.3103.

Dr. Denis Garagić

Dr. Denis Garagić is a Chief Scientist in the Sensor Processing and Exploitation (SPX) group under BAE Systems Technology Solutions. He is a key innovator, guiding SPX’s creation of cognitive computing solutions that provide machine intelligence and anticipatory intelligence to solve challenges across any domain for multiple United States Department of Defense customers including DARPA, the services, and the intelligence community. Denis has 20 years of experience in the areas of autonomous cooperative control for unmanned vehicles; game theory for distributed and hierarchical multilevel decision-making, agent-based modeling and simulation; artificial intelligence and machine learning for multi-sensor data fusion, complex scene understanding, motion activity pattern learning and prediction, learning communications signal behavior, speech recognition, and automated text generation. He received his B.S. and M.S. (Mechanical Engineering & Technical Cybernetics; Applied Mathematics) degrees from The Czech Technical University, Prague, Czech Republic, and his Ph.D. in Mechanical Engineering – System Dynamics & Controls from The Ohio State University.

Dr. Brad Rhodes

Dr. Brad Rhodes directs research and technology development of state-of-the-art machine learning capabilities for multi-INT behavior modeling, anomaly detection, behavior prediction, and event and activity pattern learning and recognition across multiple domains for BAE Systems Technology Solutions. Brad previously served for 8 years as Technical Director of Adaptive Reasoning Technologies (ART). ART created machine learning and artificial intelligence computing solutions that provide cognitive, anticipatory intelligence to solve challenges for multiple Department of Defense customers including DARPA, the services, and the intelligence community. He is a key innovator in the successful application of his team’s capabilities to many ISR and EW problems and programs in the areas of computational modeling of neural systems, human sequence learning & production, knowledge hierarchy learning, motion activity pattern learning and prediction, and application of these models and approaches to real-world problems. He received his B.S. and M.S. degrees in Applied Science from RMIT University, Melbourne, Australia, and his Ph.D. in Cognitive & Neural Systems from Boston University.

Fast Online Learning in the Presence of Latent Variables

Durdane Kocacoban, James Cussens

In this paper, we present an improved algorithm based on our previous work for online learning of causal structure in the presence of latent variables. We present a fast online causal structure learning algorithm which can track changes in a causal structure and process data in a dynamic real-time manner, requiring substantially fewer conditional independence tests than in work previously considered. The online causal structure learning algorithm we present here can revise correlation values without reprocessing the entire dataset and use an existing model to avoid relearning the causal links which still fit data in the prior model. The proposed algorithm is demonstrated on synthetic and real-world datasets with continuous features and compared with our previously considered online algorithms. Although the output of the proposed algorithm is slightly less informative in some cases, the algorithm outperformed our previous work in the way of sample and computational complexity.

1 Introduction

We study the problem of online causal structure learning from data with hidden variables. In the literature, there are many kinds of causal structure learning algorithms which have been developed successfully and applied to many different areas [18, 20, 2, 11]. Although all are successful learning algorithms, they share an important feature by assuming that the causal structure does not change during the data collection process. In real-world scenarios, a causal structure often changes [14]. To quickly identify these changes and then learn a new structure are both crucial. Therefore, it is not possible to determine these changes with existing batch-learning approaches; instead, the structure must be learned in an online manner.

We represent an online causal structure algorithm Really Fast Online Fast Causal Inference (RFOFCI) which is a modified version of Fast Online Fast Causal Inference (FOFCI) which is represented in our previous paper [12]. In a nutshell, FOFCI algorithm is an online causal structure learning which minimises the learning cost of the current model by using the causal relationships between the variables of the previous model. We believe that this algorithm made a useful contribution to the online learning area.

FOFCI have to perform a series of conditional independence tests which play an essential role in their complexity. The conditional independence tests performed by the algorithms increase exponentially with the number of variables in the data set so that these algorithms may become computationally infeasible for large graphs.

In the real world, we have always had a vast amount of data such as genetic datasets contain thousands of genes or neuroscience datasets contain tens of thousands of voxels. That means millions of conditional independence tests. This fact indicates that we need to improve these algorithms in a way that will become faster and more flexible. Thus, we decided to take these algorithms one step further in terms of the speed of analysis.

In this purpose, we propose an algorithm to an alternative to FOFCI for one who wants to analyse large data sets in the best possible time. The algorithm we offer here is Really Fast Online Fast Causal Inference (RFOFCI). RFOFCI is a modified version of FOFCI to minimise independence tests by avoiding the conditional independence tests given subsets of Possible-D-SEP sets [21]. As you can see in experimental results part, it may become very large for sparse graphs. Therefore, RFOFCI uses dramatically fewer conditional independence tests than FOFCI. That makes RFOFCI faster than FOFCI for sparse graphs. Conversely, the output of RFOFCI can be less informative in some cases, most notably concerning conditional independence information.

We have organised this paper as follows. We begin with the some real world applications and basic terminologies. Next, we provide a detailed description of the online causal structure learning algorithm. Next, experimental results are given as evidence of successful learning of causal structure in the presence of confounding factors in a dynamic environment. In this study, we assume the data has a multivariate Gaussian distribution. Next, we compared RFOFCI and two causal structure learning algorithms OFCI and FOFCI [12] by concerning average running time, learning performance and the number of independence tests performed by the algorithms when the causal model is just locally changed.

2 Some Real-World Applications

Most real-world problems mean coping with uncertainty and often include several factors or variables to consider when solving them. Probabilistic graphical models provide an ideal framework to solve complex problems with uncertainty such as Bayesian networks have been defined as an appropriate tool for analysing evidence in complex criminal and legal cases [4, 23]. Additionally, graphical models have been applied for analysing gene interactions to detect conditional dependencies [5, 7].

Graphical models have also been applied in much more areas such as Mobile robot localisation, navigation, and planning, Diagnosis for complex industrial equipment such as turbines and power plants. User modelling for adaptive interfaces and intelligent tutors, Speech recognition and natural language processing, image processing, Pollution modelling and prediction, Information retrieval, Energy markets.

Some methods aim to estimate the time-varying causal model. The DOCL algorithm proposed by Kummerfeld and Danks is applied to tracking causal structure [16]. They demonstrated the excellent performance of their algorithm. It is important to note that their algorithm provided inspiration and raw material for our work.

In contrast to our work, the work of Kummerfeld and Danks [16] do not allow for the possibility of hidden and selection variables and relearns a structure only at the points where the structure has changed, and the learned models are not used again. However, the key problem with learning cause and effect from observational (as opposed to interventional) data is the presence of hidden confounders. Different types of probabilistic graphical models are more suitable for different applications.

The critical point for the current structure learning approaches is that they assume the data comes from a single generating structure. Therefore these methods cannot be used directly for learning when a structure changes during the data collection process. Current approaches are not able to keep up with systems in a developing and changing world. Therefore, we need new tools to handle this, which are capable of tracking changes in a structure as online and giving results in a reasonable amount of time.

These methods are not able to cope with realworld data as they suffer from a large number of statistical tests and ignore the existence of confounding factors. We present a new approach which is capable of detecting changes even when multiple times and learning structure in the light of sequentially incoming data in the presence of confounding factors.

3 Basic Terminologies

A graph G is a pair (V, E) where V is a set of variables, here corresponding to random variables, and E is a set of edges. A directed acyclic graph (DAG) is a graph which has only directed edges and contains no directed cycles [6]. A causal directed acyclic graph is a graph whose edges can be interpreted as causal relations. A causal DAG is causally sufficient, if and only if every cause of two variables in the set is also in the set [13].

Three random vertices α, β and γ are called an unshielded triple if α and β are adjacent, β and γ are adjacent, but α and γ are not adjacent. If two random vertices α, β are independent given a set S, then S is called a separation set of α and β. A maximal ancestral graph (MAG) is an ancestral graph where each missing edge corresponds to a conditional independence relationship [3]. If the observed variables of two DAGs encode all the same conditional independence relations, they are called Markov equivalent [22]. A partial ancestral graph (PAG) represents a Markov equivalence class of MAG [17]. The PAGs we will study can have (a subset of) the following edges: → (directed), ↔ (bi-directed), − (undirected), ∘–∘ (nondirected), ∘– (partially undirected) and ∘→ (partially directed).

In 1999, Spirtes et al. [19] proposed the Fast Causal Inference (FCI) algorithm. Therefore, we will not describe how the FCI works and deals with hidden variables [20]. In a nutshell, FCI is modified and extended version of PC algorithm, allowing arbitrarily hidden and selection variables [3]. It accepts the existence of hidden and selection variables and has been designed to show conditional independence and causal information between random variables [3]. The role of a hidden variable in BNs can be seen as a mechanism for gathering information about the interaction between other variables in the network [13].

4 Problem Definition

Given a set of continuous variables V, we assume that we have a true underlying causal model over V at each moment in time. We specify a causal model by a pair hG, Fi, where G denotes a DAG over V, and F is a set of linear equations. These kinds of causal models are also known as recursive causal Structural Equation Models (SEMs) [24]. We assume that the data are independently generated from the true underlying causal model at each moment in time, though we do not assume that this causal model is stationary through time. The Online causal structure learning algorithm proposed here takes a new datapoint as input at each time step and outputs a graphical model (PAG). The algorithms are separated into three functionally different parts.

The Online Covariance Matrix Estimator (OCME) [16] takes each datapoint sequentially as input and performs the online updating of the sufficient statistics (covariances, sample size and means) for causal learning from data [15, 16]. In particular, OCME maintains an estimated covariance matrix over the variables and updates the estimated covariance matrix in return for incoming datapoints. As OCME does not store any of the incoming new datapoints, its memory needs only the estimated covariance matrix, in contrast with batch mode algorithms. Batch mode algorithms require all data-samples and the estimated covariance matrix memory [15, 16]. Thus, online algorithms have a substantial memory advantage compared to them. As we do not assume a stationary causal model, the datapoints should be weighted differently in a way to weight more recent datapoints more heavily after a change occurs. As different datapoints can get different weights, we use the effective sample size [16] rather than the true sample size which is adjusted to the previous effective sample size based on the new datapoints relative weight [16]. If the datapoint weights are constant, then the effective sample size is equal to the true sample size.

The Causal Model Change Detector (CMCD) [16] tracks the fitness in between the current estimated covariance matrix and the input data to detect the changes in the underlying causal model, which require changes in OCME datapoint weights [15, 16]. Specifically, the fit between each incoming datapoint and the current estimated covariance matrix is given by the Mahalanobis distance. A large Mahalanobis distance for any particular datapoint can merely indicate an outlier; consistently large Mahalanobis distances over multiple datapoints state that the current estimated covariance matrix fits poorly to the underlying causal model. Therefore, the new datapoints should be weighted more heavily [15, 16]. The approach is to first calculate the individual p-values for each datapoint. The Mahalanobis distance of a V-dimensional datapoint from a covariance matrix estimated from a sample of size is distributed as Hotellings T2 [8]. So then, a weighted pooling method to aggregate those p-values into a pooled p-value by using Liptaks method [1] is used. Finally, the weight of the next point, given the pooled p-value, is determined.

The Causal Model Learner (CML) [16] learns the causal model from the estimated sufficient statistics (covariance matrix and sample size) provided in OCME. Kummerfeld and Dankss [16] algorithm uses the PC algorithm [20] as a standard constraint-based causal structure learning algorithm. However, the key problem with learning cause and effect from observational (as opposed to interventional) data is the presence of hidden confounders.

In this way, we proposed OFCI and FOFCI in our earlier work [12]. OFCI uses the classic FCI algorithm instead of the PC algorithm in CML part in contrast with the method of Kummerfeld and Dankss work [16]. Therefore, it can be said that OFCI is a slightly modified version of the PC. The main modification is made in FOFCI by adding OFCI a real online feature. FOFCI uses the separations sets as a tool to check edge addition or deletion in the current learned model. In FOFCI, the separation sets of the current model are saved, and modified FCI uses these separation sets as input with covariance matrix and sample size while learning new causal models.

In this modified version unlike the classic FCI, the structure learning part starts with checking of causal links in the separation sets before learning skeleton. If all or some causal links of the prior model still fit incoming data, we do not need to apply the independence tests which are required to find these causal links. Then, the structure learning algorithm finds the initial skeleton by starting with the graph obtained after this analysis and updates the separation sets at the same time. After learning the causal model, the separation sets which are updated according to the new model are stored to use in the next change point.

By comparing to the PC algorithm, FCI also searches Possible-D-SEP in the last step of skeleton search, which is defined as follow.

Definition 1 Possible-D-SEP- Suppose G is a graph, which contains , >, ↔ edge styles and V_x is a vertex in G. Possible-D-SEP (V, G) function computes as follows: V_y vertex is in PossibleD-SEP(V, G), if and only if there is a path p between V_x and V_y in G such that for every subpath < a, b, c > of p, b is a collider on this subpath or it is a triangle in G [9].

The size of the Possible-D-SEP sets plays an essential role in the complexity of the FCI algorithm. As the number of variables in a dataset increases, the number of conditional independence tests performed by the algorithm exponentially grows.

Then, the computational complexity of FOFCI dramatically increases because of both computing all Possible-D-SEP sets and testing conditional independence given all subsets of these sets, which can become very large for sparse graphs. Although the FOFCI algorithm is good to learn the changing causal models, it suffers from exponential runtime. Therefore, FOFCI may not be feasible on data sets with large numbers of variables.

In this purpose, we introduce RFOFCI to fill this gap. RFOFCI is a fast causal structure learning algorithm to alternative FOFCI [12] for one who wants to deal with data sets that are too large or complex to be dealing within the best possible time. The RFOFCI algorithm differs from FOFCI for CML the part by ignoring the conditional independence tests given subsets of Possible-D-SEP sets.

In RFOFCI, CML part starts with checking of causal links in the separation sets before learning skeleton. If all or some causal links of the prior model still fit incoming data, we do not need to apply the independence tests which are required to find these causal links. Then, the structure learning algorithm finds the initial skeleton without searching Possible-D-SEP by starting with the graph obtained after this analysis and updates the separation sets at the same time.

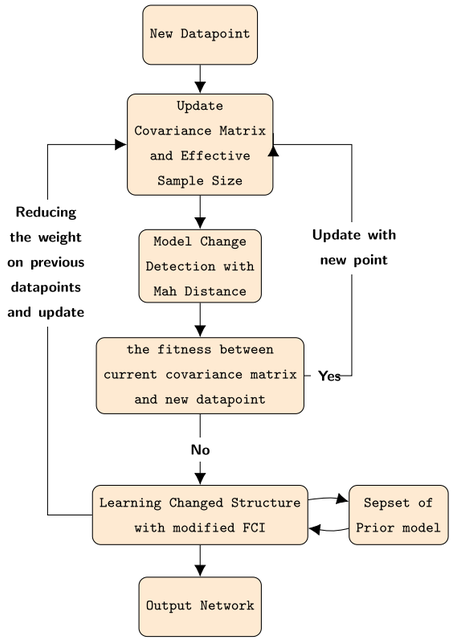


Figure 1: Basic flowchart of RFOFCI algorithm

After learning the causal model, the separation sets which are updated according to the new model are stored to use in the next change point. Therefore, we can start analysing on a more straightforward graph rather than starting from a complete graph like in the classic FCI. This simple Figure 1 represents a process of RFOFCI.

RFOFCI is faster than FOFCI and OFCI. Sometimes, the output of RFOFCI is slightly less informative than FOFCI and OFCI, but the causal interpretation of its output is still sound. As can be seen in the experimental results section, this sometimes reduces the independence test by ninety per cent.

Optionally, a probabilistic relearning scheduler is added to the algorithm, which utilises the pooled p-values calculated

in the CMCD module to determine when to relearn the causal model. For the first two parts (OCME and CMCD), in this study, we just described these parts and their functions. OCME and CMCD are identical in DOCL, OFCI, FOFCI and RFOFCI and defined by Kummerfeld and Danks. Therefore, in-depth mathematical knowledge (equations, properties, theorem and proof) of these parts can be found in Kummerfeld and Dankss works [16, 15].

5 Experimental Results

5.1 Synthetic Dataset Application

Synthetic datasets are used to verify the accuracy of our online algorithm inference approach when given a known ground truth network. Results are evaluated under the condition where the true partial ancestral graph is changed during the data collection process. We have created each synthetic dataset by following the same procedure by using the pcalg package for R [10]. First, we generated four random DAGs, which each DAG has the same number of nodes and is different from each other. Each random DAG is generated with a given number of vertices p0, expected neighbourhood size E(N) and sample size 10000. Next, we concatenated the data from these four different graphs that have the same characteristics (vertices, E(N) and sample size) to obtain a dataset with 40000 samples. Therefore the dataset is created by aggregating 4 different graphs distributions. That means there are three change points in each data. We do this to see the performance of OFCI, FOFCI and RFOFCI in the case where the causal structure is changed multiple times. We restrict each graph to have two hidden variables that have no parents and at least two children. (Selection variables are not considered in this study.)

Our goal is to present an alternative algorithm that works in real-world scenarios, that not only tracks the change of causal structure but also can compete with existing online (OFCI, FOFCI [12]) and batch structure learning algorithms concerning cost even when the causal structure does not change. The algorithms would relearn the causal model for each time when the data presents evidence that the underlying causal structure has changed and output a PAG. However, examining each PAG output for a data set which has 40000 sample size will not be possible. Since we already know the main structural change points in the data set, we applied to help of a relearning scheduler to the algorithm to see the performance at these points.

First, we investigated the performances of OFCI, FOFCI and RFOFCI, considering the number of differences in the output by comparing to the Markov equivalence class of the true DAG. We made a large scale schedule review. The relearning scheduler is scheduled for the main change points, which are 10000, 20000, 30000, 40000 for all algorithms. Instead, we also see from the Mahalanobis distance graph in Figure 4 that the algorithm detects these main change points even if we do not add this scheduler. So CMCD part determines change points of the data independently of the scheduler. If only a scheduler is added as an input (this is optional and allows to learn anytime we specified), the CML part of the algorithm will learn at the specified points of this scheduler by ignoring detected change point from CMCD.

We used two simulation settings: small-scale and large-scale. The simulation setting is as follows. For each value of $p^0 \in \{25, 30, 35, 40\}$ and $p^0 \in \{100, 125, 175, 200\}$, we generated 160 random DAGs with E(N) =2 (that means each node has a two neighbours at most). We generated a data set that has n =40000 sample size and the p-value for independence tests set to $\alpha = 0.05$.

In real-world datasets, some edges or adjacencies in the model may stay stable during data collection and learning process. Therefore, while generating random PAGs, we give attention to generate sample graphs which have local differences.

The results for the small and large scale graphs are represented in Figure 2. Figure 2 shows the average number of missing or extra edges over 5 replicates, and we see that this number was almost identical for all algorithms. As expected, OFCI and FOFCI perform the same and outperform RFOFCI to learn the true causal model in some cases. Zero means that there are no missing or extra edges, and the algorithm works correctly. High numbers represent the poor fit to the true causal model.

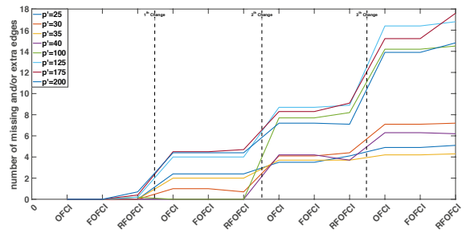


Figure 2: OFCI, FOFCI and RFOFCI, average number of miss and/or extra edges

Figure 3 shows the average percentage difference of independence test number in small and largescale settings. We first determined the number of necessary independence tests to learn the causal model for FCI, OFCI, FOFCI and RFOFCI. Then, we calculated the percentage difference of the independence test of these algorithms according to FCI. We see that RFOFCI requires significantly fewer independence tests compared to OFCI and FOFCI to learn the causal model for all the same parameter settings. In Figure 3, high numbers represent the success of the algorithm in pruning search space. RFOFCI outperforms OFCI and FOFCI in a large margin.

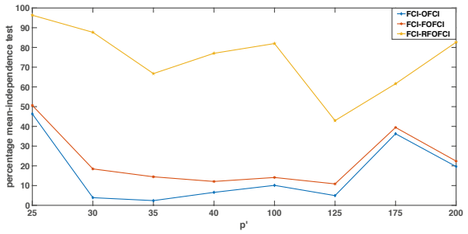


Figure 3: OFCI, FOFCI and RFOFCI average percentage reduction of independence test in contrast with FCI

Table 1: Average percentage reduction of Conditional Independence Test number performed by Algorithms

Dataset	OFCI	FOFCI	RFOFCI	Better?
Dataset (25)	46.3	50.6	96.3	√
Dataset (30)	3.8	18.4	87.6	√
Dataset (35)	2.3	14.4	66.7	√
Dataset (40)	6.5	12.0	77.0	√
Dataset (100)	10.1	14.1	81.9	√
Dataset (125)	4.9	10.8	42.9	√
Dataset (175)	36.2	39.4	61.6	√
Dataset (200)	19.6	22.4	82.6	√

It can be seen in detail in Table 1. We continued with a comparison of average percentage reduction of conditional independence test number performed by OFCI, FOFCI and RFOFCI (in seconds) under the same simulation settings. Table 2 shows the average running times in the small and large-scale setting.

Table 2: Average Running Time in seconds for 40000 sample size data

Dataset	OFCI	FOFCI	RFOFCI	Better?
Dataset (25)	16.3	14.8	13.8	√
Dataset (30)	19.5	18.9	15.4	√
Dataset (35)	26.0	21.8	20.9	√
Dataset (40)	33.4	28.0	22.1	√
Dataset (100)	113.8	101.6	98.6	√
Dataset (125)	171.1	154.8	148.0	√
Dataset (175)	338.5	307.9	298.3	√
Dataset (200)	501.4	377.7	373.3	√

Durdane Kocacoban
I am a PhD student in the Artificial Intelligence Group in the Computer Science Department at University of York since 2015. I am a graduate of the Middlesex University where I received an MSc in Operational Research, with Distinction in 2015. My earlier degree is BSc in Mathematics at Selcuk University in 2010. My main research interests are in the fields of online machine learning approaches particular Bayesian Networks and graphical models. Statistical methods in artificial intelligence.

James Cussens
I am a Senior Lecturer in the Artificial Intelligence Group in the Computer Science Department at University of York since 1997. Before that researcher at Kings College, London, Glasgow Caledonian and Oxford. I graduated from the University of Warwick with a BSc Computer Science degree in 1986. After graduating with a BSc in Computer Science, I received my PhD in Philosophy of Science at Kings College London in 1989. My main interest is machine learning particular Bayesian methods and graphical models. Applying discrete optimisation algorithms to machine learning. Statistical methods in artificial intelligence.

We see that RFOFCI is faster for all parameter settings. RFOFCI learned the causal models faster than OFCI and FOFCI. As the scale expands, the difference between them also grows.

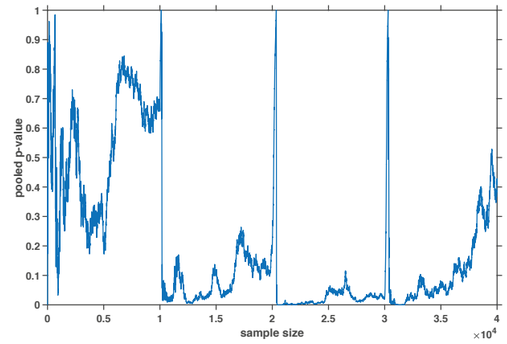


Figure 4: Pooled p-values

We also represented pooled p-values Figs 4. The datasets are a mix of four different distributions that indicate a large number of synthetic variables.

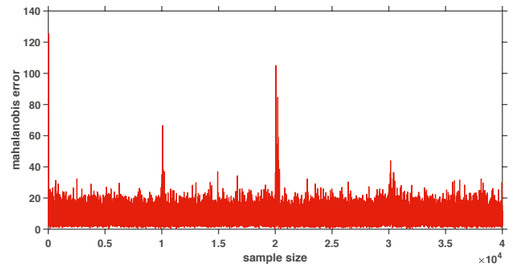


Figure 5: Mahalanobis distances

CMCD part of three algorithms measures significant Mahalanobis distance at changing datapoints as can be seen from the example in Figure 4. Therefore, it leads to higher weights and learns the new underlying causal structure.

The algorithm does not store any of datapoints coming sequentially. Its memory requirements are just for the estimated covariance matrix and sample size. Therefore, the algorithm has significant storage advantages for computational devices that cannot store all data.

5.2 Real-World Data Application

We have applied the CMCD part (it is same for three algorithms) of the online algorithms to seasonally adjusted price index data available online from the U.S. Bureau of Labor Statistics to confirm the efficiency of the change detection part of the online learning algorithms.

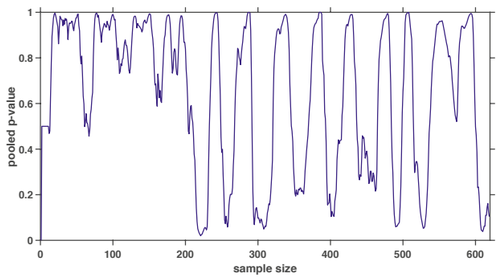


Figure 6: Pooled p-values

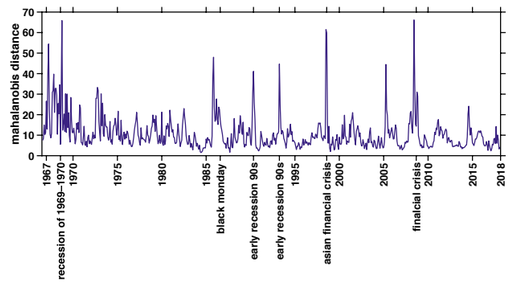


Figure 7: Mahalanobis distances

We have limited the data to commodities extending to at least 1967 and resulting in a data set of 6 variants: Apparel, Food, Housing, Medical, Other, and Transportation. Data were collected monthly from 1967 to 2018 and reached 619 data points. Due to significant trends in the indices over time, we used the month-to-month differences.

Figure 6 and Figure 7 show the drivers of these changes: the pooled p-value and Mahalanobis distance that has been collected for each month. Notably, the proposed algorithm detects a shift in the volatility of the causal relationships among these price indexes around recession of 1969-1970, the black Monday in 1987, 1990s early recession, Asian financial crisis in 1997 and Global financial crisis in 2007-2008.

6 Discussion and Future Research

In this paper, we introduce an alternative fast algorithm to the online algorithms previously proposed [12] for one who wants to deal with data sets that are too large or complex to be dealing with in best possible time for learning causal models, which is called RFOFCI. We evaluated the performance of this algorithm by testing them on synthetic and real data. The results show the efficacy of the proposed algorithms compared to other online algorithms proposed in previous work [12].

The outputs of OFCI, FOFCI and RFOFCI are almost identical to each other for most cases. Also, FOFCI requires substantially fewer conditional independence tests than OFCI and FOFCI to learn the causal model for both small and large numbers of variables. Additionally, we showed that RFOFCI is faster than OFCI and FOFCI concerning fewer conditional independence test number.

The online algorithm proposed here is good for learning changing causal structure. We showed that the algorithm is useful for tracking changes and learning new causal structure in a reasonable amount of time. However, the algorithms have limitations. Sometimes, the new model learning process of algorithms takes a long time because they require most of the data samples to learn the true model. This means that online algorithms will perform poorly if the causal structure changes rapidly. As can be seen from Figure 2, the output of RFOFCI is slightly less informative in some situations, regarding conditional independence information.

RFOFCI has a plug-and-play design. This feature allows for easy modification to use alternative algorithms. A range of alternative structure learning algorithms could be used for the learning part, constraint-based methods such as RFCI [10] and score-based methods such as greedy search algorithms, depending on the assumptions one can make. Thus, the developments in structure learning algorithms will automatically improve the performance of this online structure learning algorithm.

RFOFCI can track sufficient statistics for a linear Gaussian system efficiently. This problem is much harder for categorical/discrete variables or non-linear systems, as there will typically not be any compact representation of the sufficient statistics. One potential advantage of this approach is a way to learn conditional independence constraints in an online fashion, and then those constraints can be fed into any structure learning algorithm we want.

Acknowledgements

We want to thank Erich Kummerfeld, David Danks and David Heckerman warmly for valuable recommendations, helps and sharing their sources with us.

References: **[1]** Lipt´ak. On the combination of independent tests. Magyar Tud. Akad. Mat. Kutato Int. Kozl, 3:171–197, 1958. **[2]** David Maxwell Chickering and Christopher Meek. Finding optimal bayesian networks. Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence, pages 94–102, 2002. **[3]** Diego Colombo, Marloes H Maathuis, Markus Kalisch, and Thomas S Richardson. Learning high-dimensional DAGs with latent and selection variables. 2011. **[4]** A. Philip Dawid. An object-oriented bayesian network for estimating mutation rates. In AISTATS, 2003. **[5]** Nir Friedman, Michal Linial, Iftach Nachman, and Dana Pe'er. Using bayesian networks to analyze expression data. In Proceedings of the Fourth Annual International Conference on Computational Molecular Biology, RECOMB '00, pages 127–135, New York, NY, USA, 2000. ACM. **[6]** David Heckerman and Ross Shachter. Decision-Theoretic Foundations for Causal Reasoning. Journal of Artiicial Intelligence Research Submitted, 3(6695):405–430, 1995. **[7]** James Hensman, Neil D. Lawrence, and Magnus Rattray. Hierarchical bayesian modelling of gene expression time series across irregularly sampled replicates and clusters. In BMC Bioinformatics, 2012. **[8]** Harold Hotelling. The generalization of student's ratio. Ann. Math. Statist., 2(3):360– 378, 08 1931. **[9]** Markus Kalisch, Martin M¨achler, Diego Colombo, Alain Hauser, Marloes H Maathuis, and Peter Bu¨hlmann. More Causal Inference with Graphical Models in R Package pcalg. 2012. **[10]** Markus Kalisch, Eth Zurich, Martin M¨achler, Diego Colombo, Marloes H Maathuis, and Peter Bu¨hlmann. Causal Inference using Graphical Models with the Package pcalg. **[11]** Ron. Kenett and Yossi. Raanan. Operational risk management : a practical approach to intelligent data analysis. John Wiley & Sons, 2010. **[12]** Durdane Kocacoban and James Cussens. Online Causal Structure Learning in the Presence of Latent Variables. arXiv e-prints, page arXiv:1904.13247, Apr 2019. **[13]** Daphne Koller and Nir Friedman. Probabilistic Graphical Models. 2009. **[14]** Erich Kummerfeld. Theoretical Entities : Their Discovery and Justification. 2015. **[15]** Erich Kummerfeld and David Danks. Tracking Time-varying Graphical Structure. Advances in Nueral Information Processing Systems 26 (Proceedings of NIPS), pages 1–9, 2013. **[16]** Erich Kummerfeld, David Danks, and Machine Cognition. Online Learning of Timevarying Causal Structures. 2012. **[17]** Daniel Malinsky and Peter Spirtes. Estimating Causal Effects with Ancestral Graph Markov Models. 52:299–309, 2016. **[18]** Judea. Pearl. Causality : models, reasoning, and inference. Cambridge University Press, 2000. **[19]** Peter Spirtes. Causal Inference in the Presence of Latent Variables and Selection Bias, 1997. **[20]** Peter Spirtes, Clark Glymour, Richard Scheines, Stuart Kauffman, Valerio Aimale, and Frank Wimberly. Constructing Bayesian Network Models of Gene Expression Networks from Microarray Data. 2000. **[21]** Peter Spirtes, Christopher Meek, and Thomas Richardson. Causal Inference in the Presence of Latent Variables and Selection Bias1. Technical report, 1996. **[22]** Thomas Verma and Judea Pearl. Equivalence and Synthesis of Causal Models*. 1991. **[23]** L. F. Wright. Bayesian networks and probabilistic inference in forensic science. Journal of the Royal Statistical Society Series A, 170(4):1187–1187, 2007. **[24]** Sewall Wright. Correlation and Causation.pdf. 20:557–585, 1921.

Automatic Detecting Inconsistency between Diagnosis and Chief Complaint in Electronic Medical Records

Lufei Huang, Wei Yang, Ting Jiang, Li Tang, Fei Teng, Zheng Ma, Xuan Li

With the wide adoption of electronic medical records, quality checking is time-consuming and inefficient. The inconsistency between diagnosis and chief complaint is not only coupled with medical standards, but also may refer to medical disputes and ethics. This paper studied automatic methods to check the inconsistency between diagnosis and chief complaints in electronic medical records. As far as we know, we are the first to propose this task. This study explored a medical knowledge graph for clinic usage, linking diseases and symptoms directly from Chinese medical textbooks. Based on the knowledge graph, we proposed a method to detect the errors in real electronic medical records. The experimental results showed that this method had a good performance with F1 value of 0.837, which can improve the efficiency of quality control on electronic medical records in hospitals.

1 Introduction

Electronic Medical Records (EMRs) contain a rich source of clinical information for each patient encounter [1], including chief complaints, diagnosis, examination results, medication from doctors and treatments [2]. They are widely used in secondary applications related to quality of healthcare, clinical decision support, and reliable information flow among individuals and departments involved in patient care [3, 4]. In the past decades, EMRs adoption rates in China increase dramatically, and the volume of EMR continue to grow explosively. For example, a medium-sized hospital in China has more than 10 thousand discharged patients and 1 million outpatients one year. In that case, how to keep high quality of EMRs becomes a big challenge.

Manual checking EMR is time-consuming and inefficient. Very small percentage of EMRs are checked as samples, and most of errors are related to the formats, rather than to the contents. However, the quality of content is all that matters medical care, especially the inconsistency between diagnosis and chief complaint is not only coupled with medical standards, but also may refer to medical disputes and ethics. We are the first to propose this task that automatic detecting inconsistency between diagnosis and chief complaint. There are two main difficulties for automatic detecting inconsistency. First, it requires good knowledge of diagnostics, medicine and medical terminologies to identify whether there is relation between a disease (diagnostic) and a symptom (chief complaint). The current relational databases are unable to cope with the clinical application. Second, when writing diagnosis descriptions, physicians often utilize abbreviations and synonyms, which causes ambiguity and imprecision for matching diagnosis descriptions to standard medical terms. Although a few approaches have been developed for matching English medical text, very limited studies have focused on Chinese text.

To deal with the first problem, we apply knowledge graph to provide accurate sematic relations. Knowledge graph was officially proposed by Google to improve the search engine's query ability and user experience [5]. Through organizing the web content into conceptual graphs using ontologies and Resource Description Framework (RDF), knowledge graphs make it possible to understand the human knowledge and provide an efficient querying and reasoning framework for the vast web contents. DBPedia and YAGO are prominent examples in general domains. Considering the specific features of clinical requirements,we construct medical knowledge graph in a top-down way, including representation, extraction, fusion and reasoning of medical knowledge and quality assessment of medical knowledge graph.

For the problem of representing medical terms, we propose diagnosis linking algorithm to evaluate the similarity between style-free diagnosis and disease in medical knowledge graph. Through querying the corresponding symptoms of disease, all candidate symptoms are obtained. They are then assessed to identify relevance with diagnosis.

Table 1: Definition of data type properties

Data Type Property	Interpretation
name	Name of the knowledge element
alias	Alias name of knowledge element
icd10	ICD10 coding for knowledge element
department	Department to which the knowledge element belongs
stage	Stage of disease
condition	Conditions that cause changes in symptoms, such as time, personal behavior, etc.

There are two main contribution in this paper.

- We explore a novel model to organize disease and symptom into medical knowledge graph for clinic usage. This medical knowledge graph contains 37 departments including respiratory medicine, rheumatology and nephrology, including 6,687 disease entities, 7889 symptom entities, and entity relations more than 500,000.
- We propose a framework to automatically detect inconsistence between diagnosis and chief complaint. This method achieves a good inference result with precision and recall of 77% and 92%, respectively, which can avoid most of the meaningless inferences and greatly improve the efficiency of EMR quality checking.

The organization structure of this paper is as follows. Section 2 introduces the related work of knowledge graph in medical domain. Section 3 presents the construction process of Chinese medical knowledge graph in detail. Section 4 shows the framework of inconsistence detection with the aid of knowledge graph. Section 5 analyzes experimental setting and results. Section 6 summarizes the whole work and discusses the future research.

2 Related Work

It is difficult to identify the connotative errors in the writing of the chief complaint. Li Chunchang's work expounds the relationship between the chief complaint and diagnosis [6]. He proposed that the main symptoms should be grasped first, then the clinical characteristics should be analyzed and compared clearly, and finally the disease law should be analyzed. Only in this way can we ensure that the chief complaint is correctly written. If we want to construct medical knowledge graph, we should firmly grasp the information related to symptoms. Although a large number of knowledge bases have been published in the field of life sciences, there are few Chinese knowledge bases. For example, medical information published by Unified Medical Language System (UMLS) is in English. Most of the existing studies on medical knowledge graph are based on English.

In recent work, knowledge graph as a key technology plays a very important role in the research route. Youli Fang et al. diagnosed COPD(Chronic obstructive pulmonary disease) based on knowledge graph and integrated model [7], and provided doctors with clinical decision support. Emir Mun˜oz et al. used knowledge graph and multilabel learning models to predict adverse drug reactions [8]. They propose a specific method of generating different feature sets using knowledge graph, and show the good performance of the selected off-the-shelf multi-label learning model compared with existing works. Ying Shen et al. developed a drug similarity measurement system for drug substitution therapy based on knowledge graph [9]. A user interaction model was proposed to enable users to better understand drug characteristics from the perspective of drug similarity and to obtain insights that are not easily observed in individual drugs. Maya Rotmensch and others have explored an automated process to learn highquality knowledge base [10], linking diseases and symptoms directly from electronic medical records. They constructed the disease symptoms relationship graph, and evaluated and validated the knowledge graph. Similarly, in our study, we constructed a Chinese medical knowledge graph by extracting Chinese medical standard literature. And We use this key technology to evaluate the inconsistency between the chief complaint and the first diagnosis in the outpatient electronic medical record.

3 Construction of Medical Knowledge Graph

Knowledge graph (KG) is a directed label graph $G = (Es, Rs)$, where Es is the set of vertices of the knowledge graph, which is used to represent the set of entities; Rs is the edge set of the knowledge graph, which is used to represent the set of fact relationships between entities. In this paper, the top-down method is used to construct the medical knowledge map, that is, the ontology of the knowledge graph is constructed first, and then the instance is created according to the ontology.

3.1 Ontology Construction

Ontology is a clear specification that is defined as a conceptual model of shared knowledge [11], and can describe concepts in a domain and their relationships well. Combining the specific knowledge in the medical domain, we use the 7-step method and the skeleton method to construct the medical ontology.

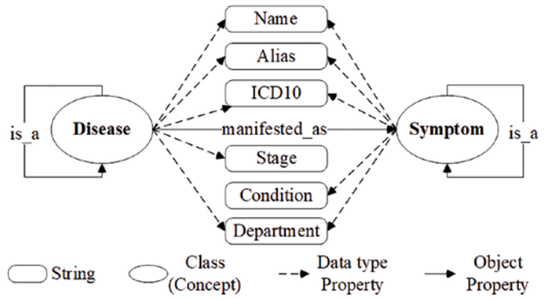


Figure 1: Ontology framework of medical knowledge graph.

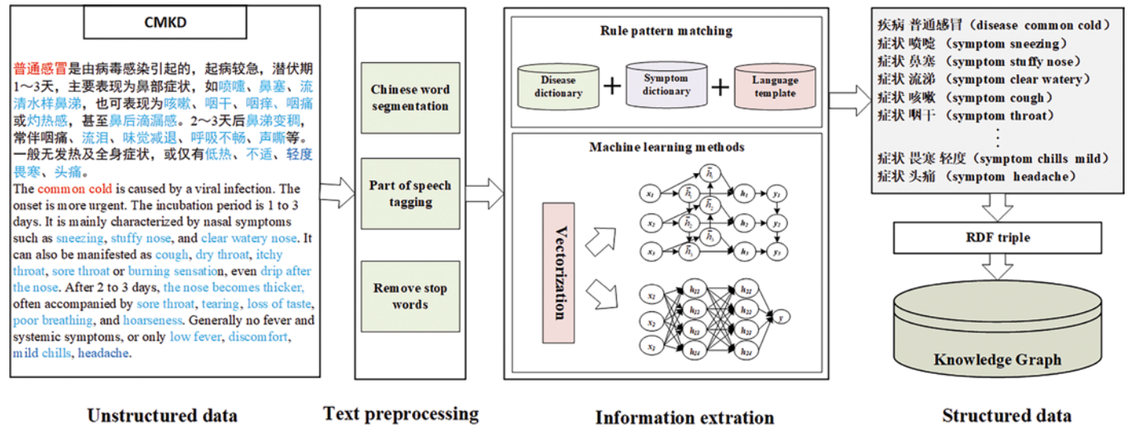


Figure 2: The creation process of instances in medical knowledge graph.The picture is in Chinese and English. Chinese characters are followed by English expressions of Chinese characters.

Figure 3: A visual display of the medical knowledge graph in Neo4j.

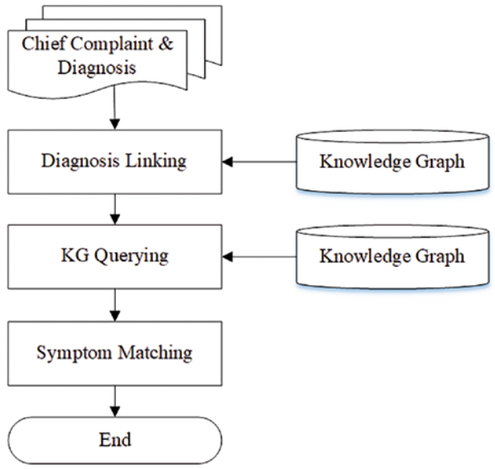


Figure 4: The consistency detection process.

The medical ontology in this paper is mainly composed of two core classes, disease and symptom, each class contains some specific properties. Property can be divided into data type property and object property, where the data type property is used to define the relationship between the class and the data describing a certain feature of the class, such as "*(disease) - name - cold*", the object property is used to define the relationship between classes, such as "*(disease) - manifested as - (symptoms)*". The definition of the detailed data type properties is shown in Table 1. In addition, three object properties are mainly considered, there are disease manifests as a symptom (*manifested as*), disease level relationship (*is a*), and symptom level relationship (*is* a). The structure of our medical ontology is shown in Figure 1.

3.2 Instance Construction

An instance is a concrete embodiment of a concept that inherits the attributes of a concept. When a large number of instances are constructed, a knowledge graph can be formed. The instance data source of our medical knowledge graph is Clinical Medicine Knowledge Interactive Platform (CMKD). Different from the common medical network encyclopedia, CMKD provides a scientific and rigorous library of clinical medicine materials, collating and compiling clinically relevant medical expertise, it can guarantee the accuracy of knowledge. CMKD elaborates on the disease in the form of unstructured text, which covers the properties of the disease, the symptoms exhibited by the disease, and the properties of the symptoms. We normalize the original unstructured data through natural language processing techniques such as named entity recognition and entity relation extraction to obtain structured data. Named entity recognition extracts the disease entity, the attributes of the disease entity, the symptom entity, and the attributes of the symptom entity; Entity relation extraction is used to determine the relationship between entity pairs in the text, and extract useful and uniform RDF triple from massive unstructured medical text.

In this paper, we use the method of rule pattern matching and machine learning to integrate knowledge to achieve

knowledge extraction [12,13]. The specific implementation process is shown in Figure 2.

Through knowledge extraction and knowledge integration, structured data for constructing medical knowledge graph is obtained. The medical knowledge graph is stored in Neo4j graph database that is currently popular and supports visual presentation. After statistics, the knowledge graph we constructed involved 37 departments including respiratory medicine, rheumatology and nephrology, including 6,687 disease entities, 7,889 symptom entities, and conceptual entity attribute instance RDF triples more than 500,000. Part of the knowledge graph is shown in Figure 3.

4 Automatic Detecting Inconsistency

The main aim of this paper is to detect the consistency between first diagnosis and chief complaint in EMRs. Research shows that the symptoms of the diagnosis should include the symptoms of the chief complaint when some hospitals judge the chief complaint. Based on the research, we proposed one solution and the detail is given as follows, firstly, extracting the keywords of the diagnosis in the medical records. Secondly, using the diagnosis search algorithm to search for the symptoms related to the diagnosis from the Chinese symptom knowledge graph. Finally, using the matching algorithm to match the symptoms obtained by searching with the symptoms of the chief complaint in the medical record and to score the symptoms obtained by searching. If the score exceeds the set threshold, the chief complaint in the medical record is considered to be consistent with the diagnosis. The consistency detection process is shown in Figure 4. The modules and algorithms involved in the process will be described in detail later.

4.1 Diagnosis Linking

For the standard diagnosis which means the diagnosis is the same as the standard disease name in the Chinese symptom knowledge graph, the symptom information corresponding to the standard disease can be obtained directly. However, the diagnosis name written by the doctor may be aliases or abbreviations in practice. Therefore, it is impossible to search the diagnosis name and get the corresponding symptoms in the knowledge map directly.

In order to solve the problem above, we propose the Diagnosis Linking algorithm. The Diagnosis Linking algorithm can select the disease with the highest degree between the disease names set in knowledge graph and diagnosis, that is the disease with the highest score.

The Diagnostic Linking algorithm is shown in Algorithm 1. In the Diagnosis Linking algorithm, the input *D* represents the diagnosis, *S* means the set of disease names in the knowledge graph, the output *O* of the algorithm indicates the disease name with the highest matching degree between the disease names set and the diagnosis, the initial value of *O* is null, and the *MaxScore* is defined as the highest score between diagnosis and disease name in the disease names set, the initial value is 0, *s* represents a disease name in the disease names set, *subD* and subs means sub-sequences of *D* and *s*, and *LensSet* indicates the set of lengths of the common subsequences of *D* and *s*. According to the characteristics of the diagnosis character sequence, the subsequence length weight *W* is defined as the ratio of the subsequence length *Slen* to the disease name s character length, and the score is determined by multiplying the weight W and the longest common subsequence length *Slen*. This makes it possible to reduce the default score of long sequences so that long sequences and short sequences can compete fairly. Traversing the disease s in the entire disease names set, the Diagnosis Linking algorithm can calculate the matching score of disease name in the disease names set and the diagnosis, and then the disease name with the highest score is obtained.

Algorithm 1 Diagnostic Linking

```
Input: Diagnosis in KG, D; Disease names in KG, S.
Output: Disease name with MaxScore matched
   with first diagnosis, O
   MaxScore ← 0;
   O ← null;
   for each s ∈ S do
       for subD s_i ∈ D and subs ∈ s do
           if subD = subs then
               LensSet ← Len(subD);
           end if
       end for
       Slen ← max(LensSet);
       W ← Slen/Len(s);
       if Slen*W > max then
           MaxScore ← Slen*W;
           O ← s;
       end if
   end for
```

Algorithm 2 Symptom Matching

```
Input: SymptomsList; Scontent
Output: Number of effective symptoms,
   EffectiveO
   EffectiveO ← 0;
   for each s ∈ SymptomsList do
       num ← 0;
       for each s_i ∈ s do
           if s_i exist in Scontent then
               num ← num+1;
           end if
           if num/Len(s) > θ then
               EffectiveO ← EffectiveO + 1;
           end if
       end for
   end for
```

4.2 KG Querying

In KG querying, the symptoms related to diagnosis can be obtained by using the cypher query language to query in the knowledge graph. In addition, the symptoms related to diagnosis can also be obtained by constructing the cypher query template to query in need.

4.3 Symptom Matching

Since the symptom information corresponding to the chief complaint content of the medical record text is unstructured data, which makes it impossible to directly judge that the symptom information shown in the diagnosis having an equal relationship or containment relationship with the chief complaint content or not. Moreover, effective evaluation of symptom consistency cannot get.

To solve the problem above, we propose the Symptom Matching algorithm. The Symptom Matching algorithm adopts the comparing results of the symptom information obtained by the previous stage of the diagnosis querying method and the chief complaint content to make a consistent evaluation. The Symptom Matching algorithm calculates the scores of all symptoms in the diagnosis querying result and the text to be recognized of the chief complaint and determines a symptom is valid or not based on the score. When effective symptoms exist, the chief complaint and the diagnosis are considered to be consistent. The specific experimental process and detailed results are in the following content.

The Symptom Matching algorithm is shown in Algorithm 2. In the Symptom Matching algorithm, all the symptoms in the diagnosis search result regard the dictionary library *SymptomsList*, the chief complaint as the text *Scontent* to be recognized, the effective symptom number is *EffectiveO* with the initial value 0, the character matched number is *num* with the initial value 0, the score for effective symptoms is the ratio of *num* and symptom *s*, moreover, θ is the score threshold for effective symptoms which is established by business, the threshold is set to 0.6 according to the characteristics of the medical record. Traversing each symptom *s* in the symptom candidate set *SymptomsList*. If the character s_i in the symptom is a substring of the chief complaint text *Scontent*, the matching number *num* is increasing. Then calculating the corresponding symptom score, when the score exceeds 0.6, the s_i symptom is considered to be a valid symptom, and *EffectiveO* increases. When the number of valid symptoms identified *EffectiveO* is greater than 0, the chief complaint is considered to be consistent with the first diagnosis.

5 Experiment

5.1 Dataset

The data of this experiment comes from a hospital in Sichuan, China that cooperates with us. We selected a respiratory doctor's one-month outpatient medical record and removed the electronic medical record samples with chief complaints missing, diagnoses missing, or both chief complaints and diagnoses missing, and obtained a total of 414 electronic medical records dataset. Every electronic medical record includes a clinical diagnosis column and a chief complaint column. In addition, all diagnosis information in the clinical diagnosis column of the electronic medical record is a bronchialrelated disease.

In the clinical diagnosis of electronic medical records, there is no single diagnosis on usual. In this experiment, we performed experiments by using the first diagnosis of cutting and extracting clinical diagnosis in the medical record.

5.2 Performance Metrics

We used three standard metrics to evaluate the results of the experiment: *P* (Precision), *R* (Recall), and *F*1. *P* represents precision which means the ratio of the number of samples predicted to be a positive example to the total number of samples. *R* represents the recall rate which represents the proportion of the number of samples predicted to be positive examples to the total number of positive samples. *F*1 expresses is the harmonic average of the precision and recall.

In this experiment, the positive cases are the wrong electronic medical records, while the negative cases are the correct electronic medical records. The calculation method is given by:

$$P = \frac{TP}{TF + TP} \quad (1)$$

$$R = \frac{TP}{TF + FN} \quad (2)$$

$$F1 = \frac{2 \times P \times R}{P + R} \quad (3)$$

Where *TP* represents the number of positive samples which are predicted correctly; *FP* denotes the number of negative samples for prediction errors; *TN* means the number of negative samples which are predicted correctly; *FN* indicates the number of negative samples with false predictions.

Table 2: Experimental Results

Result item	Value
P	0.767
R	0.920
F1	0.837

Experimental Results

The experiment has achieved good results, as shown in Table 2. The precision P is 0.767, the recall rate R is 0.92, and the F1 value is 0.837. The value of R is 0.92, which indicates that the algorithms proposed in this paper can find the vast majority of medical records with inconsistent between the chief complaint and the diagnosis. In practical applications, the medical staff can directly review the electronic medical records selected by the algorithms, which greatly reduces the workload of manual check. Furthermore, this indicates the feasibility of the proposed scheme based on the knowledge graph to detect the consistency between chief complaint and diagnosis.

5.3 Application

We designed and developed a medical record review system based on an application which is detecting the consistency between chief complaint and diagnosis in electronic medical records based on the knowledge graph. The user can use the system to upload electronic medical record files in batches, and the system will feedback the detecting results of the chief complaint and diagnosis to the user.

Figure 5 shows an example of an offline detecting of an electronic medical record. As can be seen from the figure, the symptoms described by chief complaint in the electronic medical record are lumbar spondylosis and severe osteoporosis. The first diagnosis is bronchitis, and the system gives the result that the main complaint is inconsistent with the first diagnosis.

The system verifies the feasibility and effectiveness of the algorithms we proposed, and gives the visualization effect intuitively, which brings great convenience to the medical staff to check the chief complaint and diagnosis consistency in the electronic medical record and improves the original artificial checking method. Moreover, it provides a good solution for matching chief complaint and diagnosis, which not only improves efficiency but also reduces the usage of human and material resources.

Figure 5: An example of an offline detecting. The picture is in Chinese and English. Chinese characters are followed by English expressions of Chinese characters.

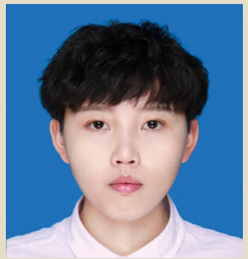
Conclusion

The inconsistency between diagnosis and chief complaint impact greatly the evaluation of doctors’ and even hospitals’ professional level. Manual detection requires a lot of time and labor concentration. It is urgent to detect the inconsistency between diagnosis and chief complaint in EMRs quickly and accurately. In the study, we first constructed the medical knowledge graph for clinic usage. The construction work is mainly divided into two parts, ontology layer and data layer, and finally acquire more than 500,000 entity relations between diseases and symptoms involved 37 departments. Based on the knowledge graph, we designed a method to detect inconsistency between diagnosis and chief complaints, and validated its performance with real cases from our partner hospital. The F1 value of the validation results reached 0.837, which showed that this method could be useful in quality control on electronic medical records.

In order to avoid unnecessary error losses, we plan to study in-depth network learning, from the end-to-end point of view, to reduce error transmission. In the future work, we also expand the medical knowledge graph according to the existing working flow and apply it to HIS (Hospital Information System) system. At the same time, we will also consider age, gender and other human attributes to detect the errors in electronic medical records. Simultaneously, we will also experiment on large-scale data sets to prove the applicability of this method.

References **[1]** Benjamin Shickel, Patrick Tighe, Azra Bihorac, and Parisa Rashidi. Deep ehr: A survey of recent advances in deep learning techniques for electronic health record (ehr) analysis. IEEE journal of biomedical and health informatics, 22(5):15891604, 2017. **[2]** Guthrie S. Birkhead, Michael Klompas, and Nirav R. Shah. Uses of electronic health records for public health surveillance to advance public health. Annual Review of Public Health, 36(1):345, 2015. **[3]** Raghupathy Anchala, Di Angelantonio Emanuele, Prabhakaran Dorairaj, and Oscar H Franco. Development and validation of a clinical and computerised decision support system for management of hypertension (dsshtn) at a primary health care (phc) setting. Plos One, 8(11):e79638–e79638, 2013. **[4]** Mark A. Musen, Yuval Shahar, and Edward H. Shortliffe. Clinical Decision-Support Systems. 2014. **[5]** Peliknov and Zuzana. Google knowledge graph [eb/ol]. Google, 2014. Accessed 16 July 2019, http://www. google.com/ insidesearch/features/search/knowledge. html. **[6]** Li Chunchang. The correct diagnosis comes from the identification and analysis of complaint symptoms. Chinese Community Physician, 35:17–17, 2011. **[7]** Youli Fang, Hong Wang, Lutong Wang, Ruitong Di, and Yongqiang Song. Diagnosis of copd based on a knowledge graph and integrated model. IEEE Access, 7:46004–46013, 2019. **[8]** E MuOz, cek V Nov, and P. Y. Vandenbussche. Facilitating prediction of adverse drug reactions by using knowledge graphs and multilabel learning models. Briefings in Bioinformatics, 20(1), 2017. **[9]** Shen Ying, Yuan Kaiqi, Dai Jingchao, Tang Buzhou, Yang Min, and Lei Kai. Kgdds: A system for drug-drug similarity measure in therapeutic substitution based on knowledge graph curation. Journal of Medical Systems, 43(4):92, 2019. **[10]** Maya Rotmensch, Yoni Halpern, Abdulhakim Tlimat, Steven Horng, and David Sontag. Learning a health knowledge graph from electronic medical records. Scientific Reports, 7(1), 2017. **[11]** Liu Q., Li Y., Duan H., Liu Y., and Qin Z. Knowledge graph construction techniques. Journal of Computer Research and Development, 2016. **[12]** Buzhou Tang, Xiaolong Wang, Jun Yan, and Qingcai Chen. Entity recognition in chinese clinical text using attention-based cnn-lstmcrf. BMC Medical Informatics and Decision Making, 19(S3):74, 2019. **[13]** Jingcheng Wang and Josiah Poon. Relation extraction from traditional chinese medicine journal publication. In IEEE International Conference on Bioinformatics and Biomedicine, 2017.

Lufei Huang



Wei Yang

Ting Jiang

Li Tang

Fei Teng

Zheng Ma
Xuan Li

KI-SIGS: Artificial Intelligence for the Northern German Health Ecosystem

Stefan Fischer, Martin Leucker, Christoph Lüth, Thomas Martinetz, Raimund Mildner, Dirk Nowotka , Frank Steinicke

KI-SIGS (“KI-Space für intelligente Gesundheitssysteme”, engl.: AI Space for Intelligent Health Systems), is an initiative to strengthen the Northern German health ecosystem to meet the challenges it is facing due to the rise of AI technologies. We present KI-SIGS by first describing the current state of the health ecosystem and the challenges it is facing. Then, we present our approach to meet those challenges, which basically consists of the interplay of three components: (i) an adaptive AI platform, (ii) an R&D (research and development) program using this platform, and (iii) the ecosystem itself. The initial partners of KISIGS are described, and finally, an outlook on sustainability options is given.

1 Introduction

Shaping the structural changes in the economy, labour market and society associated with digitisation and AI technologies represents a major challenge. This is especially true for the health care industry in Germany which traditionally has an SME-sized structure while being simultaneously exposed to strong international market competition. This challenge has been characterised by three decisive technological developments over the last ten years: major advances in machine learning, a rapid increase in data volumes and a massive increase in available computing power [1]. Taken together, these three developments lead to the breakthrough of AI and to disruptive innovations in our networked society, not only, but also and especially in the healthcare sector [2]. This in turn leads to considerable pressure for innovation in the industry. The increasing digitalization in the health care industry, medicine and medical technology leads to a rapid growth of the number of large, heterogeneous, complex and partly unstructured data sets in all areas of the life sciences. These data sets hold great potential for intelligent healthcare systems and adaptive selflearning AI technologies [3]. Moreover, comprehensive data exchange within and among institutions generate a multitude of new data-driven applications. The interoperability of medical devices and combining health data from distributed data sources of care and research open up a variety of possibilities for the exploitation of learning algorithms and adaptive AI systems. Intelligent assistive health technologies, robotic systems with assistive components up to humanoid robots and mobile health applications open up new fields of application also in nursing and rehabilitation.

The development AI in the healthcare industry must focus on people in order to comply with the regulatory framework of the industry. The high data protection requirements with regard to personal interests and the ethical requirements for the use of health data, which arise from the debates in our society, is particularly sensitive in the medical field. Medical technology companies and healthcare institutions are also faced with the question of new business models and their application in the situation of an ageing society [4].

First AI applications in the field show that their benefits are enormous (for a recent survey see [5], for a more general state of the art see [6]). Ranging from a better understanding of disease mechanisms through optimized diagnostics and therapy, increased efficiency in medical care, these applications cover a broad spectrum of value creation opportunities. However, there is a need for clarification as to how these benefits can be reconciled with the industry-specific challenges with regard to approval/regulation, data protection/ethics and financing/busi-

ness models under the conditions of international competition. This applies in particular to the explanation and transparency of the results, decisions and actions of AIsupported systems required in the healthcare sector.

With a gross value added of over 350 billion euros (2018), the core area of the healthcare industry is of considerable economic importance for Germany. This applies in particular to the North German economic area of Bremen/Hamburg/Schleswig-Holstein, in which the health industry plays an outstanding role as one of the central key industries. The industrial health sector makes a strong contribution to value creation and employment in the federal states (29.1% in Hamburg, 21.2% in Schleswig-Holstein and 16.9% in Bremen). With an annual growth rate of 4.1 percent, the sector has grown significantly faster than the gross domestic product over the past ten years (BMWI Health Economics Facts & Figures 2018). This strong starting position was created by decades of close cooperation between science and industry, which is reflected especially by the successful cluster network "Life Science Nord" (LSN) across the federal states. In recent years, this has resulted in a pulsating innovation ecosystem, from which numerous successful start-ups benefit in addition to regional universities and research institutions and large regionally based medical technology manufacturers such as Dräger, Philips, Söring, and Olympus.

Due to this favourable starting position, an AI-oriented strengthening of the North German health ecosystem offers the participants excellent opportunities to expand their position on the world market and to open up new business areas. Moreover, this may also serve as an example for other regions.

In the rest of this paper, we present the main conceptual ideas of KI-SIGS (Section 2) and present the partners participating in the project (Section 3). Section 4 looks at sustainability issues, always an important question for major infrastructural projects. Section 5 concludes and gives an overview of the future time plan.

2 KI-SIGS Concept

2.1 Objective

An AI facility focussed on intelligent health systems has not yet been established in Germany, despite its importance. In Northern Germany, however, there is a clear focus on AI-related R&D as well as application and start-up competencies in the field of health management, which form the basis of a cooperative initiative on the spatial axis Bremen/Hamburg/Schleswig-Holstein as the basis of a North German AI network in the sense of a critical mass of AI competence and market penetration. In KI-SIGS, the excellent competencies of the North German universities (in particular the universities in Bremen, Hamburg, Kiel and Lübeck) and research institutes (DFKI and Fraunhofer MEVIS) in the field of symbolic and statistical AI are to be brought into play in order to work together with the large regional companies (Philips, Stryker and Dräger) as well as small and medium-sized enterprises (e.g. Söring, Hugo Rost and apoQlar) from the health and AI sector to develop, test and deploy clinical-medical applications with the North German clinics (in particular UKSH and UKE).

Successful innovations in the high-technology sector require that in strongly connected research and development processes, with close involvement of users and in-depth knowledge of operational process framework conditions, different ideas and perspectives of a majority of participants and affected parties are implemented in new solutions. The solution approach to be developed in KI-SIGS and presented in the following therefore focuses on a decentralized AI network with a focus on knowledge gain, method competence, technology development, transfer and evaluation in the area of AI methods in the health care system. On the other hand, the focus is particularly on the challenges that the health sector brings with it, such as the problem of providing sensitive medical patient data across different locations, regulatory, approval and ethical issues.

2.2 Overall architecture

The above objectives are to be achieved through three main components to which all partners contribute in different ways:

1. development and deployment of the AI platform KI-Space with a variety of technical and nontechnical services for use by projects and ecosystems
2. definition of an R&D roadmap and its implementation in cross-location cooperative R&D projects that are constantly growing out of the ecosystem by partners from business, science and clinics using the AI platform and continuously improving it in a feedback process
3. improvement of the already excellent ecosystem in the North German health economy for the purpose of further strengthening its AI competence through the close cooperation of the project partners and the integration of further associated partners and external stakeholders on the basis of the platform, with the aim of repeatedly and increasingly producing AI-based innovations in new R&D projects and quickly translating them into products and services

All in all, the project will result in an interaction between the three components as shown in Figure 1. The platform and the R&D roadmap have to meet considerable requirements, because the roadmap must take the needs of the market and the competencies of the partners into account as much as possible, while the platform must support the development of the ecosystem - which in turn is to ensure the creation of new projects - and the most efficient implementation of the R&D projects. The design of the platform and the R&D program was therefore analysed in detail before the implementation; this process is described in the following section.

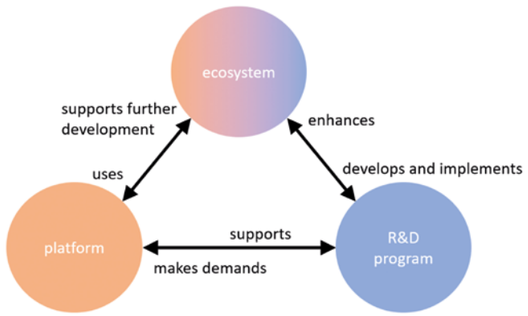


Figure 1: Interplay of the components

2.3 Requirements Analysis for R&D Program and the Platform

In order to identify and collect the requirements of the North German healthcare industry for an AI platform as well as the AI competencies and requirements of the North German healthcare industry in general, a comprehensive analysis was carried out before the implementation. In workshops, online surveys, interviews and focus groups, more than 80 companies, research institutes, universities and clinics were interviewed and the competencies and requirements of the relevant stakeholders from the economy and science of the health industry as well as the clinics in Northern Germany were collected and analysed in discussion rounds. The results make it clear that the topic of AI is already important or very important for the majority (approx. 55%) of the companies surveyed and that more than 67% assume that the topic will become significantly more important in the next 5-10 years. From the perspective of AI topics, the need to improve multimodal diagnostics through modern AI procedures, especially by means of biomedical signal processing as well as image and video processing (approx. 78%), is mentioned above all. Furthermore, the companies see a great need in the field of prediction and prognosis through better forms of data analysis but also through improvement of data management and access (approx. 78%). In addition, the companies see great potential for product innovations in the field of medical assistance systems, for example through new forms of human-machine interaction (approx. 67%), autonomous systems and robotics (approx. 57%), speech and text processing (approx. 57%) as well as virtual and extended reality (approx. 45%). While all the universities and research institutes surveyed are currently carrying out at least one AI project, less than a third of the companies surveyed have experience or expertise in the field of AI. Approximately 86% of the respondents stated that they currently have fewer than five employees working on AI projects. On the other hand, more than 71% of the companies surveyed currently have an immediate need for up to five additional employees in the field of AI. Less than 30% currently see no need or have no vacancies to fill. All respondents consider a regular exchange of knowledge to be important and would like to participate in workshops (100%), lectures (67%) and conferences (56%). 67% of the scientific partners already have or have access to specialized AI hardware, while less than one third of the SMEs have such access. Accordingly, the demand for further AI hardware is large (56%) to very large (23%), whereby GPUs and computing servers for training AI algorithms are particularly important. In the consortium, 89% of the partners would be willing to share their AI hardware.

Access to medical and clinical data as well as data sovereignty in the institutions themselves is considered essential for an AI platform. More than 78% of the partners already share data themselves or access shared data, primarily speech data, challenge data, measurement data, medical image and volume data, bio-signal and clinical data. All partners surveyed have a need for larger data sets, with anonymization and security of patient-related data as well as ethical issues having priority in the collection and processing of personal data.

2.4 R&D-Roadmap and Projects

For a targeted approach to the development of new products and services within the framework of a network, it is crucial to formulate content objectives and the path to achieving these objectives in an R&D roadmap. On the basis of the existing competencies and the requirements of the North German health sector described above, the following three future fields for R&D tasks in intelligent health systems were identified:

- Prediction and prognosis: The use of AI systems for clinical prediction and decision support benefits society and the healthcare industry equally. The application of predictive analysis functions to patient populations shows possible preventive measures, contributes to the reduction of health risks and helps to avoid unnecessary costs.
- AI-based multimodal diagnostics: With the help of AI-based multimodal diagnostics, large amounts of medical and clinical, but also behavioral, social and mobile health and care data can be combined to gain a holistic view of the patient. This supports the entire hospital and nursing staff in making diagnoses and decisions and optimizes the success of treatment.
- Medical assistance systems: Demographic change is leading to a growing demand for health services and support for nursing services. These include, for example, virtual support and robotic assistance systems that improve and facilitate training, therapy, rehabilitation or living at home in old age.

In this framework nine initial projects have been selected as the starting R&D program of KI-SIGS. Project 1 develops homecare devices for eye diagnostics for disease prevention as well as aftercare. Here, two start-up companies from the ecosystem are supported who develop and market an Optical Coherence Tomography (OCT) based measurement of macular degeneration as a homecare device.

In Project 2, both conventional X-ray images and 3D images from so-called Time-of-Flight cameras are analysed and evaluated using deep learning methods. An X-ray assistant is to be developed that also allows less qualified personnel to take high-quality X-ray images. Two highly innovative and globally active SMEs from the North see a high market potential here and are committed accordingly.

Patient data management systems (PDMS) in hospitals are increasingly recording essential patient data, which is to be evaluated and made usable with the help of AI. In Project 3, risk indicators for cardiopulmonary decompensation on intensive care units are derived from PDMS by monitoring vital parameters. In addition to the special methods for time series on the time scale of minutes, hours and days, the regulatory requirements for patient data management systems in connection with AI and the special needs of intensive wards will be incorporated into the platform.

Project 4 is a central project around AI for radiological imaging, especially in emergency and intensive care medicine. Radiological imaging is one of the main fields of application for AI because deep learning methods have been particularly successful on images. In emergency and intensive care medicine, AI-based diagnostic support systems are of greatest use, because they have to be diagnosed particularly quickly and carefully at the same time. For emergency and intensive care medicine, the regulatory system is also special and an exchange on the platform is particularly beneficial.

In Project 5, recurrent neural networks are used as in Project 3, but in a completely different context. This involves intelligent

Stefan Fischer
Stefan Fischer is a full professor in Computer Science at the University of Lübeck, Germany, and the director of the Institute for Telematics. He is also a Vice President of the university, responsible for technology transfer and digitization. His research interest is currently focused on network and distributed system structures such as ad-hoc and sensor networks, Internet of Things, Smart Cities and nano communications. He has (co-)authored more than 200 scientific books and articles.

Martin Leucker
Martin Leucker is currently a professor at the University of Lübeck, Germany heading the Institute of Software Engineering and Programming Languages. He obtained his Ph. D. at the RWTH Aachen, Germany and afterwards, he worked as a Postdoc at the University of Philadelphia, USA and at the Uppsala University, Sweden. He pursued his habilitation at the TU München, Germany. He is the author of more than 100 peer reviewed conference and journal papers ranging over software engineering, formal methods and theoretical computer science.

Christoph Lüth
Christoph Lüth is vice director of the rCyber-Physical Systems group at the German Research Centre for Artificial Intelligence (DFKI) in Bremen, and professor for computer science at the University of Bremen. His research covers the whole area of formal methods, from theoretical foundations to tool development and applications in practical areas such as robotics. He has authored or co-authored over eighty peer-reviewed papers, and was the principal investigator in several successful research projects in this area.

Thomas Martinetz
Thomas Martinetz is full professor of computer science and director of the Institute for Neuro- and Biocomputing at the University of Lübeck. He studied Physics at the TU Munich and the University of Illinois at Urbana-Champaign. Prior to Lübeck he developed Neural Networks for automation control at the Corporate Research Laboratories of the Siemens AG, was Managing Director and co-owner of the leading face recognition company in Germany, and professor for Neural Computation at the Ruhr-University of Bochum.

ultrasound aspirators, the online determination of the tissue type during the resection of brain and liver tumours. During resection, the aspirator becomes an intraoperative probe with the help of AI-based signal pattern recognition, which can be used to increase the extent of resection in tumour diseases and thus the life expectancy of cancer patients.

Project 6 is methodologically/technically closely related to Projects 2 and 9. 3D-Time-of-Flight cameras are used in all three projects. There will be an intensive exchange via the platform, also with Project 1, since this is also about the acquisition of time series of vital parameters, here especially about the acquisition of visual context information for the optimization of ventilation therapy. The patient's position is recorded with 3D cameras in order to be able to optimally adjust ventilation together with vital parameters.

Project 7 helps the surgeon in the OR by providing intervention support for the repositioning of bone fragments in pelvic fractures. The optimal repositioning of bone fragments in pelvic fractures shall be based on intraoperatively acquired 2D layer images and replace the time-consuming conventional 3D procedures. This requires learning how to map 2D layer images to 3D images.

With Project 8, we are slowly moving from therapy to aftercare. With the aging society, the number of hearing-impaired people who need hearing aids is constantly increasing. Hearing aids have to be adapted to the individual's specific hearing loss and preferences, which is difficult and time-consuming. An AI for individualized hearing aid fitting can therefore be of enormous benefit. Especially speech comprehension in acoustically challenging environments can be improved, which in turn increases wearing time and spontaneous acceptance. AI-methodically, it is closely related to Projects 4 and 6 and accordingly linked via the platform.

In cooperation with patients, therapists, physicians or nursing staff, intelligent virtual agent and robot systems for assistive movement training will be developed in Project 9. The movements are analysed by (depth) sensors and multimodal instructions (e.g. via speech and gestures) are given by agents/robots. If movements are detected correctly, human experts instruct and continuously improve the system using supervised learning.

2.5 The Platform of KI-SIGS

AI has the potential to play a decisive, even disruptive, role in the healthcare industry. This potential cannot be exploited by isolated projects in specific application areas. Rather, a uniform conceptual framework is needed that will tackle the following challenges common to all AI medical projects in a goal-oriented manner and pave the way for sustainable solutions:

- AI applications are usually data-driven, because the potential of these methods lies precisely in the discovery of unexpected, i.e. not hypothesis-driven, ideas through unsupervised learning methods. The latter, however, require large amounts of training data. Especially in health care, however, the explosive question arises as to who has sovereignty over patient data in which role. For understandable reasons, the authorities responsible for data processing (usually the clinics) are very reluctant to grant access to the data. Here, the AI space creates a framework that allows legally compliant access to patient data (data compatibility, security of access), supports its exchange between stakeholders and creates incentives for the provision of data.
- In medicine, applications are subject to strict regulatory requirements (such as legally compliant implementation in accordance with the applicable data protection regulations and the Medical Devices Act). The KI-Space supports developers in regulatory questions and helps to develop concepts for approval.
- Beyond the observance of legal rules and regulations, acceptance is also an obstacle to application that should not be underestimated, namely acceptance by the individual patient as well as by doctors, nursing staff and society as a whole. The KI-Space thinks about the ethical and social effects of the new development from the beginning.
- Furthermore, an interdisciplinary, collaborative framework must be created in which users, developers and users can build up the necessary competencies together to support a broad application and later enable the introduction into clinics.

KI-Space, the AI platform for intelligent healthcare systems, addresses these challenges and enables appropriate solutions. The AI Space has several components that address the above challenges:

- Central to the KI-Space is the collaboration platform, which on the one hand connects the participants with each other using technical aids such as portals and repositories, but on the other hand presents the KI-Space to the outside world with workshops, competitions and integration into other networks, thus making a massive contribution to the development and expansion of the ecosystem.
- On the one hand, the technical platform defines uniform communication standards, standardizes data access and provides the technical framework for services on this platform, and on the other hand it offers an application module which realizes hospital-based data processing "on premise".
- The regulatory platform ensures the conformity of the applications to be developed with legal, normative and other regulations and specifications.
- The responsible innovation platform explicitly addresses the ethical and social impacts of the use of AI technologies in the medical and health sector.

3 Partners

The following criteria have been used for the composition of the initial partner group:

1. Coverage of all links of the scientific-economic value chain in the North German health care industry
2. Integration of the main AI-competent scientific partners (universities and research institutions) with already transferable research results
3. Compilation of the economic partners as a prototypical image of the North German health care industry from corporations, SMEs and start-ups as well as clinics with high chances of success for the transfer of research results "onto the street".

The following partners have then been selected, according to the criteria:

- Universities: Bremen, Hamburg, Kiel, Lübeck
- Applied Research Institutes: DFKI Bremen, Fraunhofer MEVIS Bremen/Lübeck, UniTransferKlinik Lübeck
- Industry: Advanced Bionics (Hannover), apoQIar (Hamburg), Cellmatiq (Hamburg), Dräger (Lübeck), Gesundheit Nord (Bremen), Hugo Rost (Kiel), Image Information Systems (Rostock), mbits (Heidelberg), Philips (Hamburg), Söring (Quickborn), Stryker (Kiel), szenaris (Bremen)
- Hospitals: Universitätsklinikum Hamburg-Eppendorf, Universitätsklinikum Schleswig-Holstein Kiel/Lübeck

Raimund Mildner
Raimund Mildner is an Economist and Social Scientist. He currently is Project-Consultant to the University of Lübeck for diverse AI- and other technology- related projects, mainly in the clinical application domain of Medical Devices as well as in Logistics and Industry4.0. Before he was longstanding CEO of the Technology/Science-/StartUp-Center Lübeck and UniTransferKlinik Ltd. He has hold several supervisory board seats in companies and scientific institutions.

Dirk Nowotka
Dirk Nowotka leads the Dependable Systems group of the Computer Science department at Kiel University, Germany. Prior to joining Kiel as a HeisenbergProfessor in 2011, he was a research scientist at the Stuttgart University (2004-2011), Germany, where he gained his habilitation, and the ETH Zürich (2004), Switzerland. He completed his PhD in mathematics from the University of Turku (2004), Finland. Dirk's primary field of research is the theory and practice of automated mathematical and logical procedures for the safety and security analysis of software systems. One particular research interest of him is safety in the field of artificial intelligence.

Frank Steinicke
Frank Steinicke is full professor for Human-Computer Interaction at the Department of Informatics at the Universität Hamburg, and since 2018 he is head of the Department of Informatics. Before he joined Universität Hamburg, he was Professor at Julius-Maximilians-Universität in Würzburg (2011-2014), and Guest Associate Professor at the University of Minnesota (2009). His research is driven by understanding the human perceptual, cognitive and motor abilities and limitations in order to reform the interaction as well as the experience in computer-mediated realities.

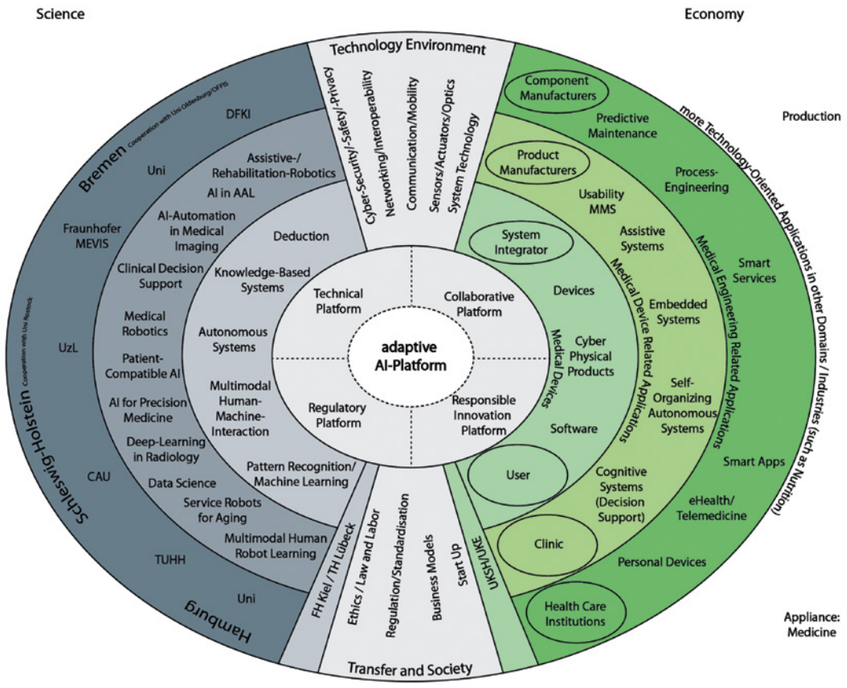


Figure 2: The ecosystem of KI-SIGS

The goal of KI-SIGS is the establishment of an ecosystem for the Northern German health system as it is depicted in Figure 2.

4 Sustainability

KI-SIGS requires considerable efforts on the part of the partners involved and the donors. It therefore pursues the objective of gradually consolidating itself in the form of a stable, permanent organisational form after a successful trial and funding phase lasting several years. This could, for example, be realised as a supra-regional (transfer) centre for innovations in the health industry, for example in the form of an R&D institution institutionalised by a sponsoring association or a non-profit limited liability company. The mandatory supra-regional cooperation then requires local transfer points as service and consulting facilities on site in order to guarantee short distances and fast reaction times for cooperation partners and customers. The costs of such a permanent network can be met in part by the contributions of the participating partner institutions (companies and scientific institutions) and by their own revenues (service revenues, workshops, training and further education), and will also require permanent co-financing from the public sector. This will provide the prerequisites for successful third-party funding by national and international funding providers. Taken together, the activities to be started now form a long-term perspective with prospects of success and enable the establishment of a sustainable transfer network for artificial intelligence in the health care system far beyond the North German region.

5 Conclusion and Outlook

In this paper, we have presented the project KI-SIGS the goal of which is the introduction of AI technologies into the products and services of the Northern German health industry in order to keep and extend the competitive advantages it has gained over the last years. We have introduced the major ideas as well as the consortium which is a nice representation of the Northern German health industry ecosystem. KI-SIGS has just recently been announced to be one of the 16 winners of the “Artificial Intelligence Idea Challenge for economically relevant ecosystems” of the German Ministry for the Economy [7]. The project will thus soon start its operational phase and will very quickly have a first version of the AI platform available. This will be the major base for a quick implementation of the R&D projects which are to start in mid 2020.

References: **[1]** LeCun Y, Bengio Y, Hinton G. Deep Learning. Nature 521, 436-444 (2015). **[2]** Esteva, A., Robicquet, A., Ramsundar, B., Kuleshov, V., DePristo, M., Chou, K., Cui, G., Corrado, G., Thrun, S., Dean, J. A guide to deep learning in healthcare. Nature Medicine, Volume 25, pp. 24–29, (2019). **[3]** Yu, K.-H., Beam, A. L., and Kohane, I. S. Artificial intelligence in healthcare. Nature Biomedical Engineering 2(10), 719-713, doi: 10.1038/s41551-018-0305-z (2018). **[4]** Strickland, E. IBM Watson, heal thyself: How IBM overpromised and underdelivered on AI health care, IEEE Spectrum, 56(4), 24-31 (2019). **[5]** Jiang F, Jiang Y, Zhi H, et al. Artificial intelligence in healthcare: past, present and future. Stroke and Vascular Neurology 2017;2:doi: 10.1136/svn-2017-000101, 2017. **[6]** Wang, F., Preininger, A. AI in Health: State of the Art, Challenges, and Future Directions. Yearbook of Medical Informatics, 28(01): 016-026 (2019) **[7]** Bundesministerium für Wirtschaft und Energie. Gewinner des KI-Innovationswettbewerbs stehen fest. [Online] 19. September 2019. https://www.bmwi.de/Redaktion/DE/Pressemitteilungen/ 2019/20190919-gewinner-des-ki-innovationswettbewerbs-stehen-fest.html (2019).

Trust in Numbers:

An Ethical (and Practical) Standard for Identity-Driven Algorithms

Mike Kiser

This paper seeks to advance the cause of ethics in artificial intelligence by proposing a practical and measurable standard for identity-driven algorithms. This standard may be used to improve internal ethical posture as well as foment comparison between existing implementations of artificial intelligence across various industries. Five different ethical axes are examined, and a graphical format is proposed that assists in the rapid evaluation and clear communication of relative strengths and weaknesses between algorithmic implementations.

1. Introduction

Who was the real Tarra Simmons? On November 16, 2017, she sat before the Washington State Supreme Court [1]. The child of addicts and an ex-addict and ex-felon herself, she had subsequently graduated near the top of her law school class. The Washington State Law Board had denied her access to the bar, fearing that the “old Tarra” would return. She was asking the court to trust her to become an attorney, and the outcome of her case rested whether or not her past could be used to predict her future.

Algorithms that use the past to predict the future are commonplace: they predict what we’ll watch next [2], or how financially stable we will be [3], or, as in Ms. Simmons’ case, how likely we are to commit a crime [4]. The assumption is: “with enough data, anything is predictable.” Over the last several years, however, headlines have repeatedly illustrated the influence of algorithms on human well-being, along with the inherent biases that affect many of them [3] [5]. Before we rush to embrace artificial intelligence algorithms, how can we ensure that they promote justice and fairness rather than reinforcing already existing inequalities?

Recent work by IBM [5], the IEEE [6], and the High-Level Expert Group on Artificial Intelligence (AI HLEG) [7] have helped to solidify an ethical approach to artificial intelligence. This paper seeks to set forth a practical and measurable standard for identity-driven algorithms that may be used to improve internal ethical posture as well as to foment comparison between existing implementations of artificial intelligence across various industries. Five different ethical axes are examined, and a graphical format is proposed to show improvement over time and to compare relative strengths and weaknesses between algorithmic implementations. This paper holds that an ethical approach to artificial intelligence has five key axes: (1) well-being, (2) accountability, (3) transparency, (4) fairness, and (5) user data rights.

2. Use Case

I. Recent Ethical Approaches to the use of Artificial Intelligence

Organizations such as IBM, the IEEE, and the EC’s AI HLEG have documented the application of ethics to artifici-

al intelligence, and they have provided a sound foundation to draw from. The five axes chosen as core to an ethical approach to artificial intelligence are shared by all three of these initiatives, with varying emphasis on each of them. For example, the IEEE calls out the impact of effectiveness misuse, and competence of a solution and provides a list of resources for further investigation. The AI HLEG provides a tiered model that describes how ethical principles translate into more practical activities. IBM’s design-emphasis emphasises the importance of establishing ethics as a key part of a solution’s design, rather than being an afterthought.

The five axes chosen in this paper reflect common elements of these approaches; by consolidating them into these five areas, visualization of a solution’s current ethical standing is simplified and a convenient structure to recommend tools and open-source resources emerges.

II. Ethical Maturity Graph

Any analysis or evaluation is only useful if it is easily understood by its intended audience. To that end, a radar chart (Fig. 1) is utilized to communicate the evaluation of any particular use of artificial intelligence and its associated algorithms. This allows for a means of evaluating both past ethical progress and the relative strengths or weaknesses of a particular implementation. Each of the five axes are evaluated and then plotted on a radar chart for quick visual comprehension (the center of the graph connotes complete ethical immaturity for that axis):

Figure 1: Ethical Maturity Graph

Since every organization seeking to utilize technology in an ethical manner may have requirements or needs unique to them, the creation of this graph is a process involving self-evaluation. At the same time, it is important to note that there are common processes and tools by all practitioners that are helpful for the evaluation of each axis.

It is the goal of this paper to standardize an approach, with easily accessible tools, and thus begin to create a practical method for ethical evaluation and comparison. To this end, each axis is examined, with important processes and best practices identified. Where possible, open-source tools or readily available techniques are highlighted, and a concise guide for self-evaluation along that particular axis is provided.

III. Ethical Axes and Evaluation

a. Well-Being

The first tenet of a practical ethical approach is that artificial intelligence must put human well-being first. This is not normally controversial assertion. The question, then, is what the term “well-being” signifies. Certainly, there is a core of universal truth that grounds well-being, as expressed in the Hippocratic oath: “do no harm,” but the concept must involve more than a mere defensive approach.

The previously cited case of Tarra Simmons is a prime example: whose well-being should be prioritized? Ms. Simmons’, who should be given a second chance and had potentially proven that she was different now? Or “society at large,” which might be negatively affected should she be imbued with legal power? Difficult choices must be made; to protect one class, another may be negatively impacted.

Determining well-being is no less tricky when developing artificial intelligence systems, as well-being is nuanced and often culturally dependent. This means exploring and documenting the interplay of interests for all parties. There may not be clear cut answers for difficult issues, but a methodological approach is essential for understanding the impact of decisions made in development of new solutions.

Using a tool known as an “ethics canvas,” these choices and outcomes may be explored and centrally documented [8]. Identification of the affected individuals, their relationships, and worldviews help to give concrete insight into what groups might be in conflict and what trade-offs are being implicitly agreed to with each decision or implementation choice. This allows for all participants — from designers to implementers — to contemplate the ramifications of their decisions, and to embed a mindset that seeks to underscore the morality of their actions more clearly. This central document establishes the ethical standard that all participants agree to abide by; it is a guiding force throughout the rest of the process. Fig. 2 provides a guideline for evaluating where an organization falls on the well-being axis:

Figure 2: Well-Being Axis Evaluation

b. Accountability

Once the ethical choices are laid out using the ethics canvas above, organizations must ensure that they are accountable for meeting the newly documented ethical standard. This is done in two primary ways. First, all ethical decisions must be documented as solutions are designed and architected. This encourages designers, architects, and implementers to make choices that are ethical and provides a documented record of why each choice was made. Second, a feedback loop must be built into the solution so that the end users (those directly impacted by the technology) have a method for holding the designers and creators of artificial intelligence systems accountable as well.

Documenting past choices reinforces the notion of responsibility for those decisions, something that is far too easily abdicated. Previous studies have found that as technology emerges, humans become more reliant on the technology rather than their own faculties. An easy illustration of this occurred as mobile devices became popular: before the rise of the mobile phone, most people had at least a score of numbers committed to memory to enable them to call friends and family. Subsequently, they relied on their devices — technology — to provide that number recall function. Research has shown that this process is writ large across society: technology such as search engines and the internet have shifted recall from the actual content (phone numbers, key facts, dates and locations) to the location to find that content online [9]. In short, that memory function has been abdicated to technology. There is an inherent danger that the rise of artificial intelligence will lead humanity to abdicate ethical choices, merely remembering “who to ask” rather than being held accountable for the decisions themselves. Documenting choices compels technologists to remember that choice (and its consequences) lies with them.

Accountability is more than internal processing, though. Giving those impacted by artificial intelligence provides essential feedback, illuminating the true effect of an algorithm. Features that allow comment and complaint into our systems in ways that are easy to use are not optional. Far from being an “add-on”, this is a requirement for accountability — if the goal is fairness, then those who feel that the decisions are unfair must be able to register their complaint. Fig. 3 provides a guideline for evaluating where an organization falls on the accountability axis:

c. Transparency

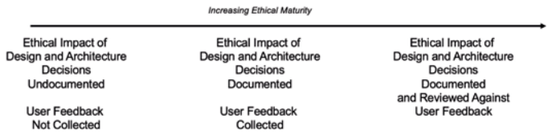


Figure 3: Accountability Axis Evaluation

For users to hold practitioners accountable, however, the reasons for those decisions must be transparent to those end users. True transparency not only answers the question of why, but it breaks down the how in a simple way. When applying this to artificial intelligence, it is critical to use language that is easily understandable to the end user, as technical jargon can quickly become complex. Imagine trying to explain why bubbles are round to a three-year-old child. A parent realizes that explaining the spherical shape via formal math is beyond the reach of their child, and so an alternate explanation is used. (This mental exercise is left to the reader.) Rather than burdening them with information, the parent seeks to equip the child to understand the world in useful, easy to understand ways.

With some forms of artificial intelligence, this simplicity is more easily achieved than others. Take machine learning, for example — how can one know what factors an outcome was based on when the artificial intelligence learns for itself? One way to do this is to use open source such as the Local Interpretable Model-Agnostic Explanations tool (LIME) (Fig. 4) [10], which can help identify the reasoning and essential factors for why a particular solution was chosen as the right one. Once the key features are identified, they may then be communicated to the end user (and the creators of the solution as well) so that they understand why the system chose that particular answer.

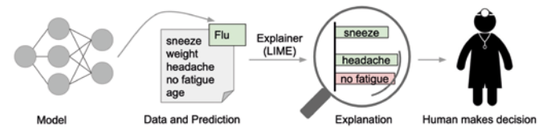


Figure 4: LIME tool for identifying features. Taken from [10].

This transparency is more than just disclosure for disclosure’s sake — it builds trust in artificial intelligence itself. Artificial intelligence that can explain itself binds machine learning to human learning — as humans learn the reasons behind identity-related decisions, they can learn to make better choices themselves. At the same time, humans can correct mistakes within the artificial intelligence’s choices. Thus, both human and machine learning are locked into a virtuous loop. Fig. 5 provides a guideline for evaluating where an organization falls on the transparency axis:

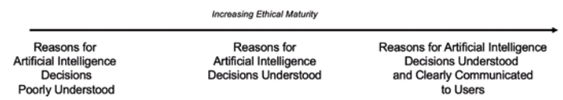


Figure 5: Transparency Axis Evaluation

This transparency has another benefit, however – it improves the ethical standing of the well-being axis by revealing any biases that have emanated from either the original dataset or the algorithm used by the artificial intelligence solution. In short, transparency ensures fairness.

d. Fairness

For artificial intelligence to promote fairness, it must expose its own biases. Detection is successful only when the full range of bias is understood, and organizations such as IBM have helpfully provided a taxonomy of bias [5], ranging from bias due to doing too much, too quickly (shortcut bias) — to false assumptions of sound judgement (impartiality bias) — to more direct prejudices (self-interest bias).

Identifying bias is sometimes straightforward, such as when an obviously flawed algorithm is clearly biased, even to outsiders with little knowledge of the system. These issues can be rapidly addressed and accounted for.

In other instances, it is more nuanced, and must be discovered through diligent examination of both the input data

and the process by which the machine learning assimilates and learns from that data set. Hiring or salary determination algorithms that use historical data, which tend to depress the value of women in the marketplace [11], are a well-known example of this. These biased data sets reflect past cultural trends in which fewer women advanced as far in their careers as their male colleagues, due to familial roles or even past cultural stereotypes. This poor data becomes a self-fulfilling prophecy that locks women into the patterns of the past.

To assist in this process, there are various open-source tools[1] that can assist in identifying biases and their prevalence both within an algorithm and its underlying data. The open-source bias audit toolkit Aequitas [12] bears special mention, as it includes python libraries, a command line tool, and a web- based audit tool that may be used to show ethical progress over time. These tools can quantify fairness for an individual algorithm, and may also be used to show the discrepancies in bias between various model or other implementation options. Fig. 6 provides a guideline for evaluating where an organization falls on the fairness axis:

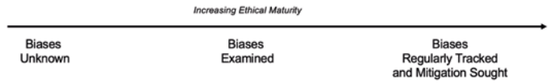


Figure 6: Fairness Axis Evaluation

What is really called for, though, is not merely algorithmic balance or a cleansing of bad data, but inclusion on a grander scale. A recent study indicated that only 18% of speakers at AI conferences [13] and only 20% of AI professors are women [14]. When historically disenfranchised groups lack a voice in the formation of AI systems, biases in the data remain uncorrected and the result is likely to perpetuate the institutionalized bias that must be eradicated. For bias to be addressed, diverse voices are needed to conduct research, to speak at conferences, and to lead the next wave of artificial intelligence development.

These various tenets of an ethical approach to artificial intelligence discussed thus far: well-being, accountability, transparency, and fairness – rely on a foundation of user data rights.

e. User Data Rights

The other tenets of ethics find their true expression in user data rights. Rather than a nice to have, privacy and control over the data that comprises identity is recognized as a fundamental human right. Article 12 of the Universal Declaration of Human Rights, adopted in 1948 by the General Assembly of the United Nations states, “No one shall be subjected to arbitrary interference with his privacy, family, home or correspondence, nor to attacks upon his honour and reputation. Everyone has the right to the protection of the law against such interference or attacks.” Over 160 countries have either enacted data privacy laws or are in the process of doing so, further reinforcing that the right to control of personal attributes and data is universal [15].

Technology that supports data user rights should be an automatic inclusion for anyone seeking to use identity and personally identifiable data in the creation of artificial intelligence solutions. These include standards that grant the user control over the use of their data as in the case of both User Managed Access (UMA) and consent management, as well as techniques that ensure personal data cannot be associated with individuals (data anonymization, pseudonymization, and differential privacy.) These techniques and tactics provide the firm grounding on which the ethical standards can be fulfilled. Fig. 7 provides a guideline for evaluating where an organization falls on the user data rights axis:

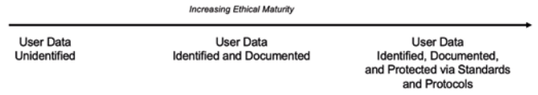


Figure 7: User Data Rights Axis Evaluation

IV. Utilizing the Ethical Maturity Graph

Once a self-audit is completed, then the ethics of the algorithm over time may be visualised:

Figure 8: Initial Assessment

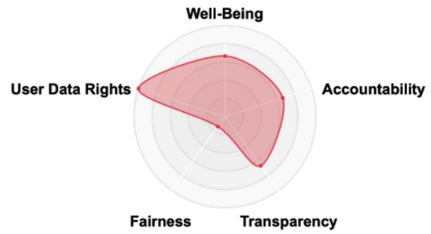


Figure 9: Secondary Review

In the figure above, a hypothetical organization already using AI completes the initial assessment in fig. 8. Well-being has been assumed rather than formally documented, resulting in a low score, and accountability is non-existent since it is unclear what the solution should be held accountable to. Later assessment finds that after work with an ethics canvas, the well-being and accountability score has improved; however, use of various tools increase transparency but also reveals previously unforeseen biases that weaken the assessment of fairness. Objectivity, however, is difficult. The recent emergence of open-source tools that give insight into algorithms (e.g. fairness) is helpful, but organizations still face challenges in quantifying more ineffable qualities such as their users’ understanding of choices made by these systems.

The goal, then, must be improvement over time — to be more aware of the impact of AI on human well-being, to be working for accountability internally and externally, to be increasingly transparent with our decision making, to be seeking out other voices that might go unheard, and to be using standards to improve how user data rights are protected.

True ethical advancement requires, however, that the journey not be a solitary one. By being able to compare various algorithms, their data, and how they are implemented, practitioners can begin to learn from one another. Lessons learned by one organization can be rapidly spread throughout the community, advancing the cause of ethics and embedding a morality-based mindset in the next wave of solutions.

Projecting several analyses of artificial intelligence and their algorithms on a single ethical maturity graph rapidly shows the relative strengths and weaknesses of each in an intuitive, visual way:

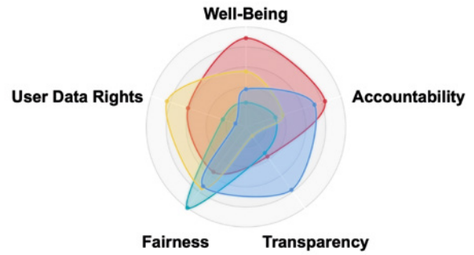


Figure 10: Ethical Maturity Graph: Multiple Algorithms

Rather than abstract discussions that are often vague, a visual representation of the ethical status of an algorithm provides a framework to facilitate interaction and allow for productive learning between organizations.

3. Conclusion

This paper has sought to further the cause of ethics within artificial intelligence by identifying a practical and measurable ethical standard, facilitated by readily available tools, for those involved in designing, directing, creating, and administering the next wave of technology. This standard was accompanied by a visual representation that rapidly communicates the ethical evolution of an algorithm, as well as highlighting the discrepancies between different solutions. The proposed standard is meant to further discussions between all who seek to embrace the next wave of technology.

Establishing and following an ethical standard for using artificial intelligence — before developing the technology or a specific use case — is essential. In Ms. Simmons’s case, the Supreme Court of Washington ruled in her favour and that she had demonstrated a change in her life for the better. She had made a clean break with her past; she was fundamentally different than she was before. As we race to embrace new technology and sources of data, and assume that with enough data, anything is predictable, we would do well to remember what the court wrote in their judgement [16]:

“We affirm this court’s long history of recognizing the one’s past does not dictate one’s future.”

Past patterns in aggregate can be helpful, but we must not lose sight of the individuals involved in the systems we build. Lest we be seduced by the power and potential of technology, we must not allow it to outpace our ethics – in short, we must strive to use artificial intelligence with our humanity intact.

References **[1]** S. Miletich, “Seattle law-school grad’s bright future outshines her rough past, state high court says,” 5 April 2018. [Online]. Available: https://www.seattletimes.com/seattle-news/seattle-law- school-grads-bright-future-outshines-her-rough-past-state-high-court-says/. **[2]** D. Jackson, “The Netflix Prize: How a $1 Million Contest Changed Binge-Watching Forver,” 7 7 2017. [Online]. Available: https://www.thrillist.com/entertainment/nation/the-netflix-prize. **[3]** K. Waddell, “The Atlantic,” 2 December 2016. [Online]. Available: https://www.theatlantic.com/technology/archive/2016/12/how-algorithms-can-bring-down- minorities-credit-scores/509333/. **[4]** J. Angwin, J. Larson, S. Mattu and L. Kirchner, “Machine Bias,” 23 May 2016. [Online]. Available: [1] ProPublica, “Machine Bias,” ProPublica, 23-May-2016. [Online]. Available: https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing. **[5]** A. Cutler, M. Pribić and L. Humphrey, “Everyday Ethics for Artifical Intelligence,” IBM Corp, New York City, 2019. **[6]** “The IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems,” 2019. [Online]. Available:https://standards.ieee.org/content/dam/ieee-standards/standards/web/documents/- other/ead1e.pdf. **[7]** AI HLEG, “Ethics Guidelines for Trustworthy AI,” European Commission, Brussels, 2019. **[8]** “The Ethics Canvas,” 2017. [Online]. Available: https://ethicscanvas.org. **[9]** B. Sparrow, J. Liu and D. M. Wegner, “Google Effects on Memory: Cognitive Consequences of Having Information at Our Fingertips,” Science, vol. 333, no. 6043, pp. 776-778, 2011. **[10]** S. S. C. G. Marco Tulio Ribeiro, “„Why Should ITrust You?“: Explaining the Predictions of Any Classifier,” in HLT-NAACL Demos, 2016. **[11]** C. O‘Neil, Weapons of Math Destruction, New York: Crown Publishers, 2016. **[12]** P. B. K. A. S. A. A. L. H. J. L. a. R. G. Saleiro, “Aequitas: A Bias and Fairness Audit Toolkit.,” ArXiv, vol. abx/1811.05577, 2018. **[13]** “Global AI Talent Report 2019,” jfgagne, 4 April 2019. [Online]. Available: https://jfgagne.ai/talent-2019/. **[14]** “Artificial Intelligence Index: 2018 Annual Report,” AI Index Steering Committee, Stanford University, Stanford, 2018. **[15]** D. Banisar, “Banisar, David, National Comprehensive Data Protection/Privacy Laws and Bills 2018 (September 4, 2018). Available at SSRN: https://ssrn.com/abstract=1951416 or http://dx.doi.org/10.2139/ssrn.1951416,” 4 September 2018. [Online]. Available: https://ssrn.com/abstract=1951416. **[16]** “Court Releases Opinion in Tarra Simmon‘s Case,” 5 April 2018. [Online]. Available: https://law.seattleu.edu/newsroom/2018-news/court-releases-opinion-in-tarra-simmons-case.

Mike Kiser
Mike Kiser is currently a Senior Strategist in the Office of the CTO. He has held a panoply of security roles over the past 20 years, in which he has designed, directed, and advised on large-scale security deployments for a global clientele. He is a member of several working groups and recently presented identity-related research at Black Hat and Def Con.

[1] Fairness-related tools include Aequitas, AIF360, Audit-AI, FairML, Fairness Comparison, Fairness Measures, FairTest, Themis™, and Themis-ML.

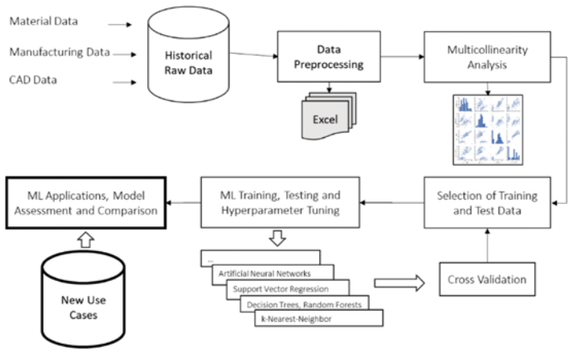
Machine Learning Based Cost Engineering of Automotive Parts – Lessons Learned

Frank Bodendorf, Jörg Franke

Strategic change management is one of the crucial fields of cost analytics in the automotive industry. Here, one important task is to explore potentials and propose measures to reduce costs of parts in the product development process. This is usually done within cost engineering and controlling departments of car manufacturing companies. The costs of part modifications are analyzed and specified usually by complex calculation methods based on a lot of input parameters describing the properties of a specific part. The calculation processes are mostly executed bottom-up. This means that the cost of a part is determined by going through many attributes and assigning and accumulating the corresponding cost estimations. These attributes comprise features of CAD designs, built-in materials as well as production sequences.

That is why the traditional calculation processes are effortful and time consuming. In addition, they are predominantly carried out manually with the help of simple calculation software like spreadsheets. Actually, Machine Learning (ML) algorithms and models are occupying center stage to cope with this problem and make the process more efficient. The overall goal is to predict costs of parts in the early phase of the product development process. In this early phase the expenditure of part modifications is low, but detailed information on the structure of the part is scarce. Machine Learning models are expected to be able to learn from a large amount of historical data and predict costs without using bottom-up calculation formulas. Also, Machine Learning algorithms are expected to find out the most influencing technical and economic parameters and in this way are able to reduce the considered input data to factors that are at hand in the early stage of the development process.

The figure below outlines the procedure model of a series of studies that has been conducted at a large automotive manufacturer in the state of Bavaria, Germany. These studies aim at developing different ML models to predict modification costs of automotive parts in an efficient way by learning from historical calculation data and results. The investigated ML models comprise Multilinear Regression, Artificial Neural Network, Support Vector Regression, Decision Tree, Random Forest, and k-Nearest Neighbor.



The developed ML models have been applied to a sample of use cases with given input parameters but unknown cost. In parallel, the use cases have been calculated traditionally and manually to provide traditional cost estimations. The ML models are then assessed and compared in regard to their predictive power and accuracy. At the same time, it is looked at the reduced input parameter set used and investigated how those input parameters can be obtained in the early phase of product development. The ML applications will be demonstrated and the results of the comparative studies will be explained.

Quality Whisperer

Britta Hilt

Self-learning Artificial Intelligence improves production quality in complex variant production:

Despite all quality checks during production, there are negative end of line tests. Even a small percentage is too much if you consider 11,000 products / day. Therefore, self-learning Artificial Intelligence discovers complex root cause in high variant production process. Thus, workers can stop disturbing factors, responsible for minor quality. Especially in 24 / 7 production, it is crucial to quickly discover and remove root cause for minor product quality.

Initial Situation

An international automotive supplier produces 11,000 car components daily in one plant, having 700 product variants. Each product which can consist of up to 600 parts is 100% tested in every technical detail before shipment. Although critical production process steps are checked again and again, and only those components continue following production line processing, which meet specification, there are test failures at end of line testing.

Project Description

The international automotive supplier´s intention with this digitalization approach is to deliver fast information to people who are responsible for decision processes to keep a plant in an optimal output with high quality products.

Solution

Self-learning Artificial Intelligence solution Predictive Intelligence discovers root cause for minor quality in a reliable and fast way. Speed is important because production runs 24 hours / 7 days a week. The sooner the real reasons for mal-functions are discovered, the sooner activities can be implemented to avoid bad quality. This saves a lot of time and reduces significant waste. The target is to reduce waste in certain manufacturing domains by 20%. The key success factor is the fast detection mechanism within the production chain delivered by Artificial Intelligence.

Complex root-cause findings can be reduced from several days to hours.

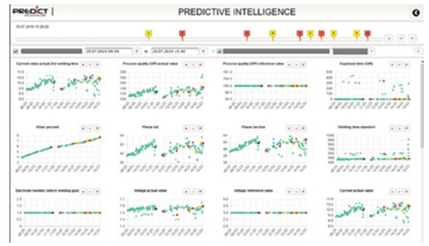


Figure 1: Self-learning Artificial Intelligence: Predictive Intelligence

Industry 4.0 –Aspects

End to end production process data is available in a data lake. This traceability is used by self-learning Artificial Intelligence algorithms to discover root cause for minor quality or waste. Continuous learning ensures that AI solution continues to be reliable, even though, production processes are changed.

Standardization Approach

Thanks to self-learning algorithms, which are realized by Artificial Intelligence, this is a highly scalable solution, even for other complex production processes with high variations.

Britta Hilt

Britta Hilt started at 1995 at IDS Scheer as international consultant. She worked for IDS Scheer until 2010, first as Core Service Manager for ARIS Software Trainings, then, as Senior Manager for ARIS Product Consulting and Solution Manager for ARIS Solution with top sales turnover. Then, Ms Hilt took over responsibility in Product Management for ARIS Platform products with top sales turnover and as Director in Product Management and Solution Marketing at IDS Scheer / Software AG. In 2011, she became co-founder of IS Predict and since then, has been Managing Director, being responsible for Marketing and Sales.

Using Existing **Reinforcement Learning Libraries** in Multi-Agent Scenarios

Carsten Hahn, Markus Friedrich

We propose an architecture enabling reinforcement learning libraries implemented for the single-agent case to be used in multi-agent scenarios. For this, we start each agent in a separate process and propose measures for process communication and synchronization. We compare our approach to the case where only a single agent is trained and its policy is distributed to all other homogeneous agents. Despite the non-stationary nature of the problem (other agents change their policy while one agent tries to find its best behavior in response), we show that our parallel architecture is able to learn a policy which outperforms policies found with the aforementioned policy distribution approach. We can explain this effect by the increased amount of training data per time step generated by multiple parallel learning agents in the simulated environment. Furthermore, we show that due to the parallelism of our architecture the number of steps done per time unit increases during the evaluation of the policy.

1 Introduction

In this paper, we propose an architecture for the extension of existing single-agent reinforcement learning libraries to support multi-agent scenarios. As mentioned in [7], multi-agent reinforcement learning yields a more natural decomposition of some problems as it is sometimes more efficient to model each individual in the environment as a separate agent instead of a single monolithic agent which observes and controls the whole environment. By this, an agent's observation, respectively action space is limited by the components it can actually perceive or control which is a far more realistic approach. Moreover, this enhances the scalability as the action and observation spaces of multiple simple agents have lower dimensionality compared to these of a single monolithic agent. The limited observation and action space of the simple agents also makes it possible to easily increase their number. Additionally, with multiple individual agents, the amount of training data of the environment generated per step increases.

The paper is structured as follows: Section 2 introduces the multi-agent reinforcement learning scenario that is considered in this paper as an example use-case. Section 3 discusses related approaches that were proposed for similar scenarios. Our architecture is described in Section 4. This is followed by its evaluation in Section 5 and a conclusion in Section 6.

2 Scenario

In this work, we exemplarily consider a predatorprey scenario in which multiple autonomous agents have to flee from a predator. It was already used in [5]. The agents shall learn to survive as long as possible before being caught. The predator acts on a static policy by following the nearest agent. In this scenario, the agents are selfinterested and each agent has its own reward signal depending on whether or not the agent has survived the last time step. This eases the credit assignment problem which occurs in cooperative multi-agent reinforcement learning as it is sometimes unclear how much a single agent contributed with its action to the achievement of the cooperative goal.

The environment is depicted in Figure 1 with the homogeneous agents in green and the predator in orange. All individuals can move freely in the two dimensional space. The space wraps around at the edges meaning that an agent leaving the space at the left border re-enters it on the right. The agents move with fixed speed and are able to influence their movement direction through their actions. The observation of an agent is restricted to itself, the predator and the n nearest neighboring agents. For every observable individual, the agent observes the euclidean between to the individual, the angle the agent would have to turn to face towards the observed individual and the absolute orientation of the entity in the environment.

3 Related Approaches

One approach to solve the described scenario could be to train a single monolithic agent which can observe and control every individual in the environment with reinforcement learning designed for single-agent use-cases. Assuming each individual has a different actions, then the combination of these actions (joint action) grows exponentially with the number of agents n: an. Thus, this approach does not scale well with the number of agents.

Egorov [4] proposes an approach for multi-agent reinforcement learning in which in a group of homogeneous agents, only one agent learns with DQN [10] while the other agents keep their policies fixed. After a certain number of iterations, the policy learned by the training agent is distributed to all other agents of its type.

Approaches exist for multi-agent reinforcement learning with implicit weight sharing between the policies or explicit communication between the agents in order to account for the non-stationarity of the problem. RLlib is a software library [8] which comes with implemented algorithms for the singleas well as for the multi-agent case and tries to ease the implementation of multi-agent algorithms through composable components, shared policies, hierarchical control and efficient parallelism. However, RLlib is quite sophisticated and complex and we argue that there are cases in which single-agent reinforcement learning algorithms can be easily expanded to the multi-agent case. We will present our architecture for this in the next section.

4 Multi-Agent Architecture

Reinforcement learning typically consists of an agent taking actions in an environment (based on the observed state of the environment) and receives a reward (depending on how useful its action has been) as well as an observation of the new state of the environment. OpenAI Gym [3] is a toolkit for developing and comparing reinforcement learning algorithms. It offers multiple environments which expose an interface that follows the previously described schema. An agent interacts with these environments by calling a step(action)-method with an action-parameter, which returns an observation of the new environment state as well as a reward. OpenAI Gym is designed for single-agent reinforcement learning and there are multiple reinforcement learning libraries like Keras-RL [11] or Tensorforce [6] which come with pre-implemented algorithms following this interface.

In our considered scenario with self-interested agents which try to avoid being caught by a predator, it would be possible to train only one agent and copy its learned policy to the other agents, for example after each episode (as shown in [5]). This eases the problem on the non-stationarity as n − 1 agents are static while one agent tries to find the best response to the behavior of the others (and the present predator of course) which leads to an emergent swarm. The described scenario can be straightforwardly implemented with existing reinforcement learning libraries and an environment that exhibits the OpenAI Gym interface as outlined in Figure 2.

If not all agents should follow a single copied policy, a monolithic learner which sees and controls all agents could be used. But a far more natural decomposition of the problem would be to train all agents in parallel although it has the property of non-stationarity. In the following, we will present an architecture that expands libraries for the single-agent case like Keras-RL or Tensorforce to multi-agent scenarios. This is done by starting multiple agents which concurrently invoke the environment's step(action)-method which makes synchronization of the agents necessary.

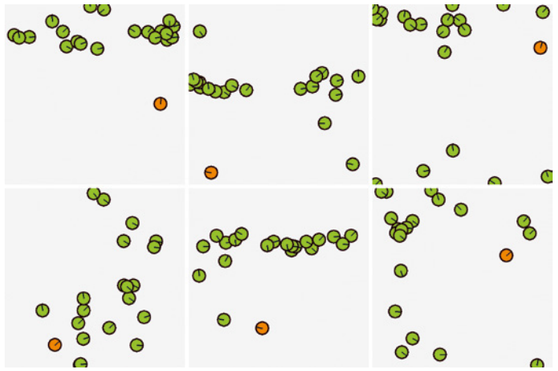


Figure 1: Six different states of the environment.

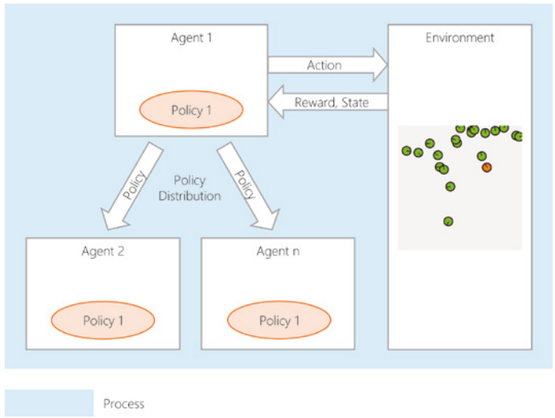


Figure 2: Architecture with a duplicated policy.

Keras-RL or Tensorforce come with preimplement reinforcement learning algorithms like DQN [10] or DDPG [9]. These algorithms use artificial neural networks to express their policies respectively to approximate the value of stateaction -functions. The neural networks are again implemented in Tensorflow [2] (possible as well in Theano [12] in the case of Keras RL). The Tensorflow Graphs Class is not thread-safe which makes it difficult to start each parallel agent in its own thread (in a shared memory space) as outlined in Figure 4. This would only be possible by starting a Tensorflow session respectively its own computational graph for each agent's neural network which makes changes in the pre-implemented libraries necessary and breaks the intended encapsulation.

This is the main reason for running each agent as well as the environment in a separated process (see Figure 3). In our architecture, all agents concurrently execute the environment's step(action)method with an action chosen based on their currently learned policies. This concurrent access on shared resources (the environment) requires synchronization no matter if threads or, as in our case, separated processes are used.

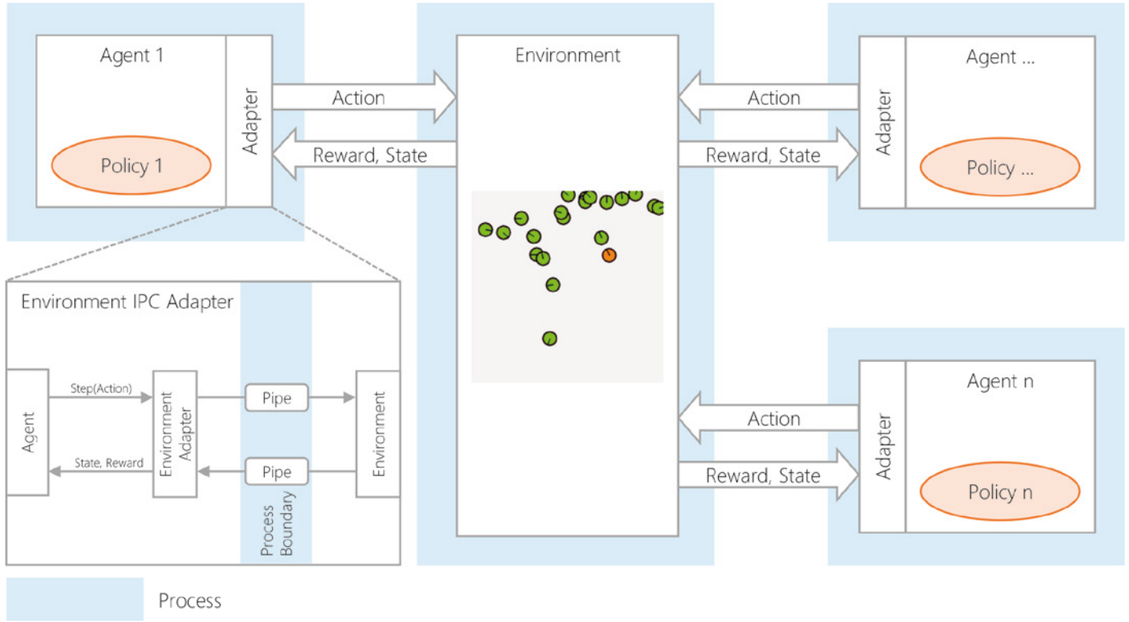


Figure 3: Architecture with agents as processes.

The synchronization is done in the step(action)-method and is explained later on. Before that, the communication between the separated processes is discussed.

4.1 Interprocess Communication

As we decided for separate processes for each agent and the environment itself in order to use Tensorflow or Theano in separated memory spaces, there is a need for interprocess-communication to pass the actions from the agents to the environment and individual rewards and observations from the environment back to the corresponding agents.

For this, we propose an extra abstraction layer implemented as an adapter design pattern which handles the communication between an agent and the environment. This decouples the agent and the communication logic which is now completely transparent. The agent, respectively the preimplemented reinforcement learning algorithm, only interacts with the adapter and still follows the step(action)-method interface as required by OpenAI Gym which does not break existing implementations. We propose pipes for interprocesscommunication but other communication schemes like message queues or sockets would be possible as well.

The environment adapter’s source code is depicted in Listing 1. An object of this class is given to each agent. It mimics an environment with an interface like OpenAI Gym and does the interprocess communication between agent and environment process by using a bi-directional pipe.

Listing 1 also shows some special communication which is further explained in the following section.

4.2 Synchronization

The synchronization inside of the step(action)method makes use of the blocking read feature of pipes in Python [1] respectively Linux. An environmental step is executed if and only if all agents have sent their actions for the next step. This on the other hand means, that all agents have to act on the same timescale which is a reasonable assumption since we talk about homogeneous agents. It also implies that the environment gets stuck if one agent stops its execution before others. This particular behavior can be circumvented by de-registering agents from the environment if their execution has been stopped (see Listing 1). This kind of signaling is also used for resetting an agent. In OpenAI Gym environments, an agent calls the reset-method in order to reset the whole environment. In our architecture, however, a reset is a special action “−1” sent to the environment which only resets the agent itself. The real environment on the other hand, can send a so-called done flag “−1” to the agent in order to end its execution.

The agent’s environment adapter receives a new observation and reward value through the duplex pipe after sending a reset indication or an action and passes these to the agent. An agent receives its observation after the actions of all agents have been executed in the environment. An alternative approach would be to send the observation of the environmental state to an agent right after its action has been executed which would result in a more sequential flow of execution. The agent’s environment adapter is blocked until new data has been received from the environment and so is the agent itself.

5 Results

In order to evaluate the performance of our proposed architecture, we first measured the number of steps that the environment executes per second. This was done in comparison to the architecture depicted in Figure 2 in which only one agent is

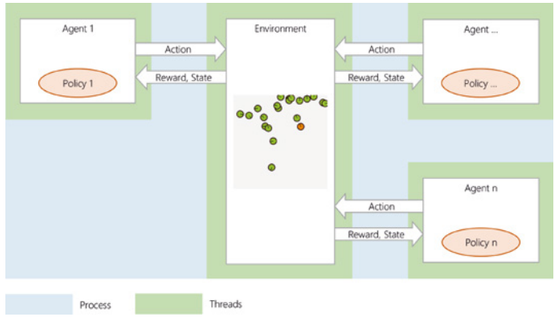


Figure 4: Architecture with agents as threads.

trained and its policy is distributed to all other agents after an episode ends. This strategy of policy distribution will be called “Clone” in the following experiments. Our approach of running multiple agents in parallel in distinct processes (as depicted in Figure 3) will be called “MultiProcess”. The comparison is done in the training stage as well as in the test phase in which the trained policy is evaluated. All experiments were done on an Intel® Core™ i7-5600U CPU which has two CPU cores and is able to execute four threads in parallel through Hyper-Threading.

Figure 5 shows the steps that are executed by the environment in case of 10 agents either training or only evaluating their policy. It is visible that during the training, more steps per second are executed for the “Clone”-strategy compared to the “Multi-Process”-strategy. But one has to keep in mind that in the latter case, 10 agents are trained in parallel instead of a single one (whose knowledge is distributed to the others) which altogether results in more training examples to learn from. During the execution of the learned policy (test) the “Multi-Process”-architecture clearly outperforms the “Clone”-architecture as it can exploit multiple processor cores without much synchronization overhead.

With 50 agents present in the environment (see Figure 6), the steps per second decline but the proportions between “Clone” and “Multi-Process” in training and test stay roughly the same. Of course, the “Multi-Process”-architecture would have benefited from a higher number of processor cores.

When training multiple agents in parallel (which breaks the Markov Property assumed for common reinforcement learning approaches), one has to investigate whether sensible behavior has been learned. Once again, the environment comprises of multiple agents which are trained to avoid being caught by a predator that might be distracted by multiple agents in its vicinity. For the case that the learned policy of one agent is distributed to all others (“Clone”) it has been already shown in [5] that a sensible behavior is learned as the agents tend to form a swarm in order to increase their survival chances. In Figure 7 the average survival time of agents trained in different architectures are depicted (with 10 agents in the environment). For the “Clone”-architecture, 100 training runs with different parameter settings were conducted and the best policy (in terms of

Listing 1: Python source code of the environment adapter

```
class EnvironmentAdapter():
   def __init__(self, real_env, env_connection):
      self.action_space = real_env.action_space
      self.observation_space = real_env.observation_
space
      self.env_conn = env_connection # duplex pipe to
the env

   def reset(self):
      observation, _, done, _ = None, None, None, None
      if self.env_conn.poll():
         observation, _, done, _ = self.env_conn.recv()
         if done == -1:
            sys.exit()
         else:
            raise Exception(,There should be nothing to
                              receive before we send
reset‘)

      self.env_conn.send(-1)
      observation, _, done, _ = self.env_conn.recv()
      if done == -1:
         sys.exit()
      return observation

   def step(self, action):
      observation, _, done, _ = None, None, None, None
      if self.env_conn.poll():
         observation, _, done, _ = self.env_conn.recv()
         if done == -1:
            sys.exit()
         else:
            raise Exception(,There should be nothing to
                              receive before we send an
action‘)

      self.env_conn.send(action)
      observation, r, done, info = self.env_conn.recv()
      if done == -1:
         sys.exit()
      return observation, r, done, info

   def deregister(self):
      self.env_conn.send(-2)

   def render(self, mode=‘human‘, close=False):
      pass
```

Carsten Hahn
Carsten Hahn is a researcher at the LMU Munich. His research interests are artificial intelligence and autonomous systems.

Markus Friedrich
Markus Friedrich is a researcher at the LMU Munich. His research focuses on 3D reconstruction based on point clouds/ image series and machine learning.

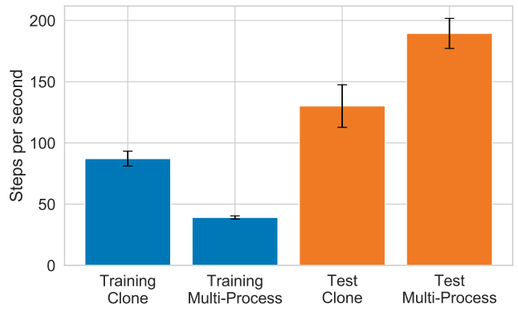


Figure 5: Steps per second of the environment with 10 agents present, either during training or test of the policy.

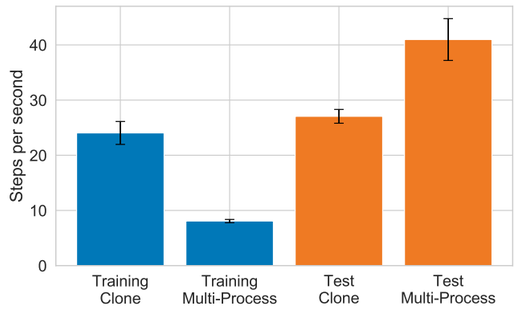


Figure 6: Steps per second of the environment with 50 agents present, either during training or test of the policy.

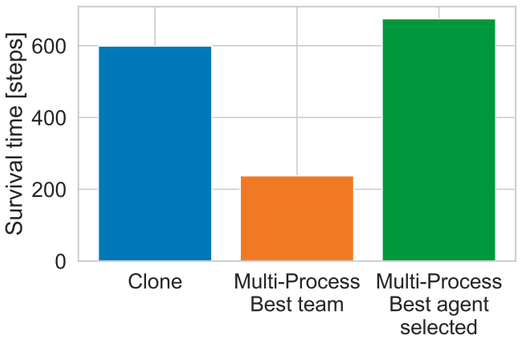


Figure 7: Average survival time of an agent with 10 agents present in the environment.

survival time) yields an average survival time of 599 steps. We also trained agents in parallel with our proposed architecture. For this, also 100 different training runs were executed which resulted in 1000 trained agents. We evaluated the performance of the agents that were actually trained together in a mutual run as a "team". These teams perform not that good, as the average survival time of an agent in the best "team" is only 238 steps. This is because in these teams, some agents learn quite sophisticated behavior patterns, like hiding from the predator behind other agents, while some agents in the team do not learn anything sensible and only turn around themselves. However, if we consider the average survival time of each individual agent trained in the "Multi-Process"-architecture (Figure 3), search for the best overall agent and distribute its learned policy to each other agent in the environment (like in the "Clone"-architecture in Figure 2), it results in a team with an average survival time of 675 time steps per agent. This outperforms the best survival time found in the "Clone"-architecture by 13%.

6 Conclusion

In this work, we proposed an architecture for the usage of libraries designed for single-agent reinforcement learning in a multi-agent case, as the decomposition into multiple autonomous agents is quite natural for many problems. We considered an example scenario in which multiple selfinterested agents try to escape from a predator in order to survive as long as possible. In our architecture, we propose to run multiple agents in parallel on this scenario in comparison to train only one single agent and copy its policy to all others. In our experiments we could find a behavior policy for the agents, which yields an about 13% higher survival time compared to the "Clone"architecture. Moreover, our parallel architecture shows better performance in the execution phase (after training) compared to sequentially evaluating a single policy for multiple agents. This could be enhanced even further on hardware with more CPU cores, resulting in a higher degree of parallelism. Justifiably, our experiments also prove that the training of multiple agents in parallel is slower than the training of only a single agent (with its policy being distributed to all others). However, again, this would benefit from a higher number of CPU cores.

References: **[1]** Python documentation. https://docs.python.org/2/library/multiprocessing.html#multiprocessing.Pipe. Accessed: 2019-07-16. **[2]** Mart´ın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, and Zhifeng Chen et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. **[3]** Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym, 2016. **[4]** Maxim Egorov. Multi-agent deep reinforcement learning. CS231n: Convolutional Neural Networks for Visual Recognition, 2016. **[5]** Carsten Hahn, Thomy Phan, Thomas Gabor, Lenz Belzner, and Claudia Linnhoff-Popien. Emergent escape-based flocking behavior using multi-agent reinforcement learning. In Conference on Artificial Life (ALIFE 2019), 2019. **[6]** Alexander Kuhnle, Michael Schaarschmidt, and Kai Fricke. Tensorforce: a tensorflow library for applied reinforcement learning. https://github.com/tensorforce/tensorforce, 2017. Accessed: 2019-07-16. **[7]** Eric Liang and Richard Liaw. Scaling multiagent reinforcement learning. https://bair. berkeley.edu/blog/2018/12/12/rllib/. Accessed: 2019-07-16. **[8]** Eric Liang, Richard Liaw, Philipp Moritz, Robert Nishihara, Roy Fox, Ken Goldberg, Joseph E. Gonzalez, Michael I. Jordan, and Ion Stoica. Rllib: Abstractions for distributed reinforcement learning, 2017. **[9]** Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. arXiv preprint arXiv:1509.02971, 2015. **[10]** Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. Nature, 518(7540):529, 2015. **[11]** Matthias Plappert. keras-rl. https:// github.com/keras-rl/keras-rl, 2016. Accessed: 2019-07-16. **[12]** Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. arXiv e-prints, abs/1605.02688, May 2016.

Towards a **Semantic of Intentional Silence** in Omissive Implicature

Alfonso Garcés-Báez, Aurelio López-López

In human communication, we often face situations where decisions have to be made, regardless of silence of one of the interlocutors, i.e. we have to decide from incomplete information, guessing the intentions of the silent person. Implicatures allow us to make inferences from what is said, but we can also infer from omission, or specifically from intentional silence in a conversation. In some contexts, not saying p generates a conversational implicature: that the speaker didn't have sufficient reason, all things considered, to say p. This behavior has been studied by several disciplines but barely touched in logic or artificial intelligence. After reviewing some previous studies of intentional silence and implicature, we formulate a semantics with five different interpretations of omissive implicature, in terms of the Says() predicate, and focus on puzzles involving assertions or testimonies, to analyze their implications. Several conclusions are derived from the different possibilities that were opened for analysis. Finally, we suggest a general strategy for the use of the proposed semantics.

1 Introduction

In human communication process, we often face situations where decisions have to be made regardless of omission or intentional silence of one of the interlocutors, doing implicatures, as often occurs in every day dialogues. The conversational implicature is a potential inference that is not a logical implication and is closely connected with the meaning of the word "says", as explained in [8]. A formulation of implicature goes as follows, with S the speaker and H the hearer (also referred as addressee) [15]. S conversationally implicates p iff S implicates p when:

1. S is presumed to be observing the Cooperative Principle (cooperative presumption);
2. The supposition that S believes p is required to make S0s utterance consistent with the Cooperative Principle (determinacy); and
3. S believes (or knows), and expects H to believe that S believes, that H is able to determine that (2) is true (mutual knowledge).

Where the Cooperative Principle (CP), as introduced by Grice [8], consists of the participants making their conversational contribution as required in the scenario in which this occurs for the accepted purpose of the speech exchange.

We also have to make decisions, possibly with implicatures, from incomplete information, guessing the intentions of the person in case of an omission or intentional silence. In this direction, Swanson [11] defines Omissive implicature as: "In some contexts, not saying p generates a conversational implicature: that the speaker didn't have sufficient reason, all things considered, to say p."

This phenomenon has been studied by several disciplines but barely touched in logic or artificial intelligence despite being an intelligent behavior in everyday human exchange. To the best of our knowledge, there is not an approach to formalize the use of omission or intentional silence in implicatures, in terms of logic, where the closest attempt was an "informal" logic in [9].

This paper offers a first logical approach to the study of omission or intentional silence in implicatures, analyzing the implications of five different interpretations of omission derived from two types of silence. For the analysis of these interpretations, we take as case study a couple of puzzles, formally expressed and previously solved, representing assertions in terms of the predicate Says defined as:

Definition 1.1 Says(X,Y) expresses that the agent X asserts predicate Y.

From now on, we employ the term omission or intentional silence, indistinctly.

The main contributions of this work consist of:

1. Semantics for two types of intentional silencewith five derived interpretations.
2. The use of intentional silence in implicaturesfrom agent assertions (testimonies).
3. Implementation of the semantics in Answerset Programming.
4. A general strategy for the application of thesemantics of intentional silence.

This paper is organized as follows: Section 2 shows some viewpoint related with the implicature and omission. Section 3 includes the semantics of two types of silences and five interpretations. Section 4 describes a case study consisting of two puzzles, and Answer-set Programming that served to implement and explore the semantics. Section 5 includes the five interpretations and their consequences in the case study. We conclude in Section 6, discussing in addition work in progress.

2 Implicature and Omission

The interpretation of omission must be contextual. For example, in a normal conversation, in court, intentional silence is interpreted in detriment of the person who is silent. The immediate reaction is that she hides something. For Kurzon [10], the silence is defined by language and points to three types of silence:

- Psychological silence. This silence is unintentional, and the help of a decoder is necessary.
- Interactive silence. It occurs as an intentional pause in the conversation allowing the other person to draw inferences related to the meaning of the conversation. Here, intentional silence is deliberate.
- Socio-cultural silence. When silence is interpreted based on specific cultural codes.

The intentional silence is also a sign of group loyalty. To interpret intentional silence, first we have to discard the modal "can" that may express unintentional silence, as in "I can not speak" [10]. Then, intentional silence can be interpreted with four manners: 1) I may not tell you; 2) I must not tell you; 3) I shall not tell you; 4) I will not tell you. Where manners 1 and 2 are intentional external silences "by order". And manners 3 and 4 are intentional internal silences "by will".

Bohnet and Frey [2] state that the variants for the interpretation of silence can be: anonymous and not anonymous, where the latter can be with identification and face-to-face. They also studied silence in the communication process, specifically in the context of prisoner's dilemma.

Umberto Eco [4] suggests that from the point of view of semiotics, silence is a sign. In a communication scheme that includes the interpretation of silence in its basic form, the speaker has to interpret the silence that the listener sends with a certain intention. The sender becomes the receiver of silence.

Walton [13] discusses the use of Gricean implicature [8] in the context of testimonies in law, specially in terms of enthymemes, i.e. arguments with a missing premise or conclusion. We present here a first approach to bring implicature to cases with missing testimonies (silence) and a strategy for analysis.

The conventional (or systemic) implicature is related to the semantic view point for the widely understood linguistic meaning, and is related with the "unsaid". According to the concepts expressed by Fern´andez [5] related to conversational and conventional implicatures, we work on the conversational implicature (i.e. on "what is said") represented by the starting specification of the puzzles taken as a case study. On the other hand, the conventional implicature involves the interpretation of "what is not said", similar to the assumption of omission of statements by one or more of the agents involved in the case study.

2.1 A Classification of Silence

After reviewing some previous studies on implicature and silence, we now detail a classification that serve to understand the types interpreted in this research.

Kurzon [10] proposes a first taxonomy that served to further development. We start from such taxonomy to include some contexts from our findings during the literature review as well as our proposed semantics, as Fig. 1 illustrates. Silence is a diverse and intangible object that we learn to interpret within the context in which it appears. In this first classification, we show the following three contexts: Socio-cultural, in the Arts, and Interactive. Silence in the socio-cultural context is related to religion or the beliefs of ethnic groups [1]. Silence in the interactive context or omission refers to what could have been said and for some reason was not said [11]. Silence in the arts context is related with the temporal arts, such as music and literature, as developed by Khatchadourian [9]. Interactive silence or omission can be:

- Unintentional. The reasons for the nonintentionality of silence in the communicative process can be:
  – The noise. For example, the hearer could perceive a "silence" because of the engine of a vehicle passing by [4].
  – The forgetfulness. When unintentionally, we give incomplete information.
  – Psychological. Possibly caused by a trauma.
- Intentional. Intentional silence is sent voluntarily and intentionally received [4].
  – In group. This silence occurs when several people are involved in a situation or interaction and is not dialogic.
  – Face to face. This silence is dialogical, i.e. between two agents and intentional silence is understood as Yes, No [10], or something that requires clarification. This is called contextualized polarity.
  – Evasive. Characteristic of politicians who prefer to talk about anything else, instead of answering directly a question.

The In-group silence is the context where we place our interpretations, for instance in group chat or a crime case involving testimonies of several people. The defensive silence has two variants linked to the "right to remain silent", the acquiescent silence has three variants linked with the old saying "silence is consent" and the pro-social silence is related to the silence of employees for loyalty in the context of a group or company [3].

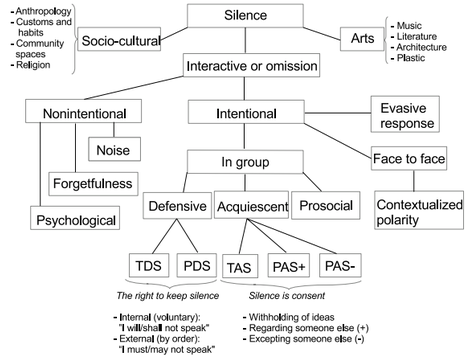


Figure 1: Some contexts where silence appears.

3 Semantics of Silence

For our definitions, we assume the following. P is a logic program or knowledge base, an n number of interacting agents, and X_{A_i} = Says(A_i,*) is all that the Ai agent Says, i.e. asserts.

3.1 Defensive Silence

This silence is an intentional and proactive behaviour that is intended to protect the self from external threats, described in [3] as follows. Withholding relevant ideas, information, or opinions as a form of self-protection, based on fear. Formally, we express it straightforwardly as:

Definition 3.1 P_{A_i} is Total Defensive Silence (TDS) of A_i understood as:

$$P_{A_i} = P - X_{A_i}$$

Where ($1 \leq i \leq n$).

Definition 3.2 $P_{A_i,pj}$ is Partial Defensive Silence (PDS) of A_i understood as:

$$P_{A_i,pj} = P - \{Says(A_i, p_j)\}$$

Where ($1 \leq i \leq n$); A_i is an agent; ($1 \leq j \leq m$); with m the number of assertions p_j done by A_i.

3.2 Acquiescent Silence

The next three interpretations of silence are related with the old saying "silence is consent", expressing a passive disengaged attitude, that is explained in [3] as follows. Acquiescent Silence: Withholding relevant ideas, information, or opinions, based on resignation. In our semantics, the following three interpretation are based on this type of silence.

Definition 3.3 P'_{A_i} is Total Acquiescent Silence (TAS) of Ai understood as:

$$P'_{A_i} = P_{A_i} \cup (\{Says(A_j, *)\} \circ \lambda)$$

Where $i \neq j$, ($1 \leq i,j \leq n$); P_{A_i} is TDS for A_i; $\lambda = \{A_j/A_i\}$, and the operator $\circ$ with λ substitution denotes the replacement of Aj for Ai on Says subset of agent A_i.

Definition 3.4 P'_{A_i,A_k} is Partial Acquiescent Silence of Ai relative to Ak (PAS+) understood as:

$$P'_{A_i,A_k} = P_{A_i} \cup (\{Says(A_k, *)\} \circ \lambda)$$

Where $i \neq k$, ($1 \leq i \leq n$); Ak is the agent supported; P_{A_i} is TDS for A_i; $\lambda = \{A_k/A_i\}$, and the operator $\circ$ with λ substitution denotes the replacement of A_k for A_i on Says subset of agent A_i.

In contrast to PAS+ where an agent relies on the assertions of other, the following partial acquiescent silence relies on the assertions of the rest of agents involved except another.

Definition 3.5 P'_{A_i,A_k} is Partial Acquiescent Silence of A_i excepting A_k (PAS-) understood as:

$$P'_{A_i,A_k} = P_{A_i} \cup (\{Says(A_j, *)\} \circ \lambda)$$

Where $A_j \neq A_k$, ($1 \leq i,j \leq n$); A_k is the unreliable agent; P_{A_i} is TDS for A_i; $\lambda = \{A_j/A_i\}$, and the operator $\circ$ with λ substitution denotes replacement of A_j for A_i on Says subset of agent A_i.

Once the semantics for the two types of silence have been stated, we elaborate on how to assess the consequences of silence in some situations, allowing to achieve implicatures.

4 Experimental Setting

Given the complication to access a collection of actual testimonies or dialogues where to evaluate the different interpretations, we recur to puzzles that include some kind of testimonies of agents and require reasoning for their solution.

4.1 A Poisoning

There is a puzzle taken from [14], here on referred as Poisoning, with the attributes to model and explore our interpretations of silence. In this puzzle, a mystery related to the murder of a person is raised, where one can assume that an officer requests and records the testimony of three suspects: Four men were eating dinner together in a restaurant when one of them suddenly struggled to his feet, cried out "I've been poisoned," and fell dead. His companions were arrested on the spot and under questioning made the following statements, exactly one of which is false in each case: Watts: 1) I didn't do it. 2) I was sitting next to O'Neil. 3) We had our usual waiter today.

Rogers: 1) I was sitting across the table from Smith. 2) We had a new waiter today. 3) The waiter didn't do it.

O'Neil: 1) Rogers didn't do it. 2) It was the waiter who poisoned Smith. 3) Watts lied when he said we had our usual waiter today.

Assuming that only Smith's companions and the waiter are implicated. Who was the murderer?

To find the solution of Poisoning, a matrix is used in [14], where possibilities are discarded while adhering to the constraint that only one statement (S) of each suspect is false. The combination of truth values False (F) and True (T), with no contradiction, allowing to reach the solution (O'Neil), is presented in Table 1.

Table 1: Solution of Poisoning puzzle

Suspect	S_1	S_2	S_3
Watts	T	T	F
Rogers	F	T	T
O'Neil	T	F	T

4.2 A Mystery

There is a second puzzle, previously modeled and solved in [6], that includes testimonies of different people, and also allows to model and explore our interpretations of silence and variants. In this puzzle, a mystery related to a murder is raised: Vinny has been murdered, and Andy, Ben, and Cole are suspects.

Andy said: 1) He did not do it. 2) Ben was the victim's friend. 3) Cole hated the victim.

Ben said: 1) He was out of town the day of the murder. 2) He didn't even know the guy.

Cole said: 1) He is innocent. 2) He saw Andy and Ben with the victim just before the murder.

We must assume that all the people involved tell the truth except, possibly, the murderer (Ben in this case).

4.3 Implementation

Even though the two puzzles share some common features (testimonies of some agents involved in a murder), they have different structure requiring separate reasoning for their solution.

The case of Poisoning is represented for the analysis in terms of the predicate Says(), as previously defined, with a third parameter that affirms (1) or denies (0) the assertion, according to the original statement. This predicate is interpreted in the sense of Grice's conversational implicature. The statements are represented as follows:

W_1:says(watts;murderer(watts); 0).
W_2:says(watts; sitting next(watts; oneil); 1).
W_3:says(watts; new(waiter); 0).
R_1:says(rogers; sitting next(rogers; smith); 0).
R_2:says(rogers; new(waiter); 1).
R_3:says(rogers; murderer(waiter); 0).
O_1:says(oneil; murderer(rogers); 0)
O_2:says(oneil; murderer(waiter); 1)
O_3:says(oneil; new(waiter); 1)

From this set of statements, we can easily identify the following contradictions: $\{W_3,R_2\} \vdash \perp$; $\{R_3,O_2\} \vdash \perp$; and $\{W_3,O_3\} \vdash \perp$

For the second puzzle, Mystery, we employed the Answer-Set Programming (ASP) paradigm to explore the implications of omission, given that is closely related to intuitionist logic, i.e. based on the concept of proof rather than truth (previously shown in [7] for intuitionist logic). This is a logical programming branch that computes stable models for difficult problems [6], where a stable model is a belief system that holds for a rational agent.

Clingo (https://potassco.org/clingo/) is an implementation of ASP that allows to find, if there exists, the answer set or stable model of a logic program. Clingo is used to generate answer sets for the problem of a case study, and explore the implications of the silence interpretations formulated. Python is used to update the program [12].

5 Interpreting Silence in Omissive Implicature

Based on the formulation of the puzzles previously described, we proceed to consider and explore five interpretations of intentional silence linked to such context. As defined in omissive implicature, the omitted predicate corresponds to p, assuming that at some point agent a could have uttered p, i.e. Says(a,p).

The first two interpretations are variants of Defensive Silence, i.e. an agent intentionally simply remains silent, mainly by fear, for which we take the Poisoning and Mystery puzzles. While the remaining three correspond to Acquiescent Silence that is adequate for the Mystery puzzle, understood as asserting with silence what others have said, commonly by resignation. We explore in all the cases, the consequences of the interpretation, assuming that a particular kind of silence occurs.

5.1 Defensive Silence in Poisoning

If an agent investigating a case faces this kind of silence of one or more of those involved, he can not count on their testimonies. So, for our case study, we have to remove the declaration of those people, as a rule in an analysis.

So, if we ignore for a moment the central constraint of the puzzle (i.e. that one of statements is false), and we assume that somebody decide to defend himself by remaining silence. Then, to proceed we have to consider each person giving testimony, silence him, i.e. bring into consideration a total defensive silence (TDS). This does not lead to a solution, given the contradictions that emerge, as shown below:

1. When Watts is silent: $\{R_3,O_2\} \vdash \perp$
2. When Rogers is silent: $\{W_3,O_3\} \vdash \perp$
3. When O'Neil is silent: $\{W_3,R_2\} \vdash \perp$

However, by recalling the central constraint that requires to consider statements in pairs, we can explore the combinations, as depicted in Figure 2, for the TDS of Rogers. In this forest, we can observe cases where possible solutions can be reached, such as W_1W_2 with O_1O_2, and other leading to contradictions, such as W_1W_3 with O_1O_3.

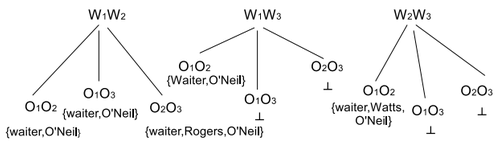


Figure 2: Pair combinations for TDS of Rogers.

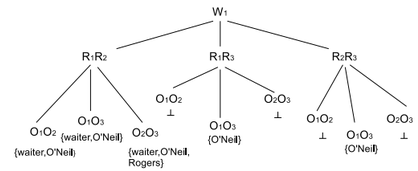


Figure 3: PDS for Statements 2 and 3 of Watts.

The main constraint of the puzzle allows devising a strategy based on partial defensive silence (PDS) to reach solutions. We have to silence one or two out of three statements, of each of those involved in Poisoning (i.e. 3). We will have a total of 216 (6^3) possible combinations, where one of them is presented in Fig. 3. However, we can reduce the number to 3^3 by taking pairs, for instance, analyzing for Watts, we can note that the PDS for W_1 and the PDS for W_2 there is no solution due to contradictions. On the other hand, the PDS for W_3 does have models that include the solution to the original puzzle and whose path is the dotted line in the tree in Fig. 4, where the omitted statements are W_3 of Watts, R_1 of Rogers, and O_2 of O'Neil. Interestingly, a second way to find O'Neil as causing the poisoning emerged, by replacing statement R_2 by R_1, that is false but does not contradict any other statement. Notice also that the other models reached in the analysis (non-contradiction leaves in the tree) involve, besides O'Neil, to the waiter.

Figure 4: PDS for Statement 3 of Watts.

As a consequence of bringing the semantics of defensive silence, specifically PDS, in this puzzle, we reached the solution summarized in Table 2.

Table 2: Solution Models for Poisoning

Suspect	S_1	S_2
Watts	W_1	W_2
Rogers	R_1/R_2	R_3
O'Neil	O_1	O_3

5.2 Two Types of Silence in Mystery

Moving now to analyze the Mystery considering the omission of testimonies, we can wonder; what would happen if silence with common sense is presented as a possibility? What conclusions the interrogator or judge can reach if some of the suspects decide to intentionally shut up?

If an agent investigating a case faces this kind of silence of one or more of those involved, he can not count on their testimonies. So, we have to remove the declaration of those people, as stated in the definition of TDS.

Applying this first rule to each person giving his testimony and executing the corresponding programs, we get those presumable guilty. That is, as a result of the silence of a person, we can analyze who becomes a candidate to blame. The possible outcomes (guilty) when a one or more suspects decide intentionally to omit their testimonies are presented in Table 3, illustrating the right to remain silent. In this, we can notice that the culprit can be anyone depending on who decides to shut up. For the possibilities, we can comment:

1. {} corresponds to the original scheme where nobody is silent, i.e. every testimony is taken into account. The only model for this case is Ben, as expected.
2. When Andy is silent, the offender turns outto be either Ben or Cole.
3. When Ben is silent, any of the three suspects may be guilty. Intuitively we can think that Ben's silence has more decision capability since anyone involved can turn out as guilty.
4. Cole's silence can turn Andy or Ben guilty.
5. With the remaining possibilities, related tomore than one person, any of the three involved may be guilty.

Table 3: TDS models for agent

Silent agent(s)	Culprit
{}	{ben}
{andy}	{ben, cole}
{ben}	{cole, andy, ben}
{cole}	{andy, ben}
{andy, ben}	{cole, ben, andy}
{ben, cole}	{cole, andy, ben}
{andy, cole}	{cole, ben, andy}
{andy, ben, cole}	{cole, ben, andy}

We can further detail the analysis by considering PDS, for this variant of silence, what would happen if only part of the information about the case is omitted. At the atomic level, which of the arguments of each one of the suspects has more impact on their total silence? That is, we can bring into consideration Partial Defensive Silence as detailed in Definition 3.2. Here we briefly provide some results:

1. In the Andy case, with the silence of his firstor second statement the culprit can be Ben, with the silence of the third one, Cole also appears as presumable guilty. Table 4 details the consequences of partial silence of Andy.

Table 4: PDS models for Andy

Silenced testimony (predicate)	Culprit
{says(andy, innocent(andy))}	{ben}
{says(andy, hated(cole,vinny))}	{ben}
{says(andy, friends(ben,vinny))}	{ben, cole}

2. Ben is the most affected with his silence, either total or partial since he comes out in every model, as Table 5 shows.

Table 5: PDS models for Ben

Silenced testimony (predicate)	Culprit
{says(ben,out-of-town(ben))}	{andy, ben}
{says(ben,know(ben,vinny))}	{ben, cole}

3. Cole can also decide, without incriminatinghimself, whom to reveal as guilty. Table 6 shows the different answers obtained.

Table 6: PDS models for Cole

Silenced testimony (predicate)	Culprit
{says(cole,innocent(cole))}	{ben}
{says(cole,together(andy,vinny))}	{ben}
{says(cole,together(ben,vinny))}	{andy, ben}

The second type of silence has three variants related with the old saying "silence is consent". This semantics is implemented by omitting the whole person's testimony and inserting new assertions related with what is implicitly assuming with silence. Table 7 shows the solutions reached for the puzzle when one or several persons are silenced under the interpretation of TAS. Again, the first line corresponds to the original situation where everybody has declared, leading to Ben as the muderer. Notice that there is no model (solution) in cases 2 and 4, where UNSATISFIABLE is obtained. These situations can be interpreted that there is no evidence to blame any of the suspects, possibly leading to a mistrial. So, under this scheme, Andy and Cole are those who could benefit from remaining silent. In cases 5, 6 and 7, the person who speaks is out of suspicion. In the latter case, as expected from common sense, when everybody is silent (no one has revealed any information), anyone can be the culprit.

Table 7: Total Acquiescent Silence for agent

Silent agent(s)	Culprit
{}	{ben}
{andy}	UNSATISFIABLE
{ben}	{ben}
{cole}	UNSATISFIABLE
{andy, ben}	{ben, andy}
{ben, cole}	{ben, cole}
{andy, cole}	{cole, andy}
{andy, ben, cole}	{cole, ben, andy}

The direct interpretation of Total Acquiescent Silence seems unintuitive since there could exist declarations of other people involved that are unknown to the person remaining silent. For this reason, the acquiescent silence is referred or relative to some other agent testifying, as stated in the PAS+ variant.

Table 8: PAS+ for agent

Silent agent	Regarding	Culprit
andy	ben	{cole}
andy	cole	{ben}
ben	andy	{cole, ben, andy}
ben	cole	{andy, ben, cole}
cole	andy	{ben}
cole	ben	{andy}

Table 8 contains the possible combinations for the three agents. Notice that this interpretation generates models in all the case, and leads to only one suspect in most cases, except when Ben recurs to this kind of silence. We can also observe that PAS+ is more advantageous to Ben than TAS.

As a complement of PAS+, a partial acquiescent silence that omits the testimony of some other agent (witness), possibly by his untrustworthiness, is PAS-. This variant of silence was given above as Definition 3.5. Table 9 details the consequences of this variant of silence if it occurs in the mystery.

Table 9: PAS- for agent

Silent agent	Excluding	Culprit
andy	ben	{ben}
andy	cole	{cole}
ben	andy	{ben, andy}
ben	cole	{ben, cole}
cole	andy	{andy}
cole	ben	{ben}

Alfonso Garcés-Báez
Alfonso Garcés-Báez is PhD student since 2018 at National Institute of Astrophysics, Optics and Electronics (INAOE). México. Professor since 1998 at the Faculty of Computer Science of the Benemérita Universidad Autónoma de Puebla. ICPC-ACM Couch in Shanghai-2005 and Tokyo-2007.

Aurelio López-López
Aurelio López-López got his Ph. D. degree in Computer and Information Sciences from Syracuse University, Syracuse NY, U.S.A. Before joining the Computational Sciences Department at INAOE, Prof. López-López worked in the national petroleum research center and academia. His areas of interest are: knowledge representation, information extraction and retrieval, text mining, natural language processing, human language technologies, and education applications.

Similar as PAS+, PAS- generates solution models in any circumstance, and leads to one suspect in most of the cases. However, the models of Table 9 differ of those of Table 8, having less suspects when Ben decides to remain silent. Notice also that the agent that recurs to this variant of silence incriminates to the agent whom untrusts.

5.3 Discussion

From assuming that silence occurs in the puzzles and interpreting it in different ways, their solution can be reached along some other, as expected given that there is less information available. Surprisingly, in particular for the poisoning puzzle, other solution was revealed with PDS.

Also, this formulation as speech acts (expressed in terms of the predicate "Says") under the assumption of silence of one or more of the interlocutors is an example of non monotonicity, since allows to draw tentative conclusions (implicatures), in particular:

- Under Total Defensive Silence (TDS), with the silence of Andy, the first model found included Ben as a solution. In this case, we have three utterances less in the knowledge base (P).
- Under Total Acquiescent Silence (TAS), with the silence of Ben, the culprit comes out also as Ben. In this case, we require four additional utterances, i.e. two less of Ben (since he is silent), three more coming from what Andy is asserting, and three more from the assertions of Cole.
- The relationship of cardinalities is the following: $|P_{TDS}| \leq |P| \leq |P_{TAS}|$.

We can now formulate a strategy for bringing intentional silence in the analysis of problems involving testimonies. Assuming that testimonies of different people involved are already available in a knowledge base (KB), the strategy is formulated in detail as follows: 1) Identify agents and predicates. 2) Formalize the statements using the predicate "Says". 3) Add definitions and common sense rules according to the problem at hand. 4) Identify the types of silence occurring in the problem. 5) Generate a KB to model the problem, including agent statements, common sense knowledge, and identified types of silence. Depending on the type of silence of the agents, define the knowledge base accordingly i.e. KB_{New}. 6) Apply ASP to get the models on KB_{New}. 7) Analyze the different scenarios obtained for decision making.

6 Conclusions

Intentional silence embodies an economy of representation given that less knowledge is expressed for reasoning, as shown in our case study. Contrasting the types of silence, we can notice that Total Defensive Silence opens possibilities while Total Acquiescent Silence restricts them. People who recur to Acquiescent Silence tend to appear as guilty, except for cases where no solution is found. Silence expresses valuable information that can be employed for decision making. In particular, when the intentional silent is interpreted according to its context, we achieve implicatures.

Understanding and modeling the implications of silence can be useful in agent interaction, either human or virtual. We foresee semantics and a useful analysis of different scenario for legal cases involving testimonies and different types of silence. We are in the process of analyzing scenarios where the two kinds of silence are displayed by participants, e.g. one is recurring to defensive silence and other to acquiescent silence. As future work, we plan to extend the interpretations to incorporate prosocial silence, i.e. retaining work-related information or opinions with the goal of benefiting other people or an organization. It remains to bring the interpretations of silence to a more general framework for agent interaction, beyond answer set programming. Also in this direction, we are exploring to consider payoffs of agents involved in the interaction, as well as to the predicates to know who or what has more gains with silence, as an instrument in making decisions.

References: **[1]** Keith H Basso. "to give up on words": Silence in western apache culture. Southwestern Journal of Anthropology, 26(3):213–230, 1970. **[2]** Iris Bohnet and Bruno S Frey. The sound of silence in prisoner's dilemma and dictator games. Journal of economic behavior & organization, 38(1):43–57, 1999. **[3]** Linn Van Dyne, Soon Ang, and Isabel C Botero. Conceptualizing employee silence and employee voice as multidimensional constructs. Journal of Management Studies, 40(6):1359–1392, 2003. **[4]** Umberto Eco. Signo. Labor, 1994. **[5]** Graciela Fernández Ruiz. Decir sin decir: Implicatura convencional y expresiones que la generan en español. Colegio de México, 2018. **[6]** Michael Gelfond and Yulia Kahl. Knowledge representation, reasoning, and the design of intelligent agents: The answer-set programming approach. Cambridge University Press, 2014. **[7]** Kurt G¨odel and Anzeiger Akademie der Zum intuitionistischen Aussagenkalku¨l. Wissenschaften wien, math.-naturwissensch. Klasse, 69:65–66, 1932. **[8]** H Paul Grice. Logic and conversation. Syntax and semantics: Speech acts, Cole et al., 3:41– 58, 1975. **[9]** Haig Khatchadourian. How to Do Things with Silence, volume 63. Walter de Gruyter GmbH & Co KG, 2015. **[10]** Dennis Kurzon. The right of silence: A sociopragmatic model of interpretation. Journal of Pragmatics, 23(1):55–69, 1995. **[11]** Eric Swanson. Omissive implicature. Philosophical Topics, 45(2):117–138, 2017. **[12]** Jake VanderPlas. Python data science handbook: essential tools for working with data. O'Reilly Media, Inc., 2016. **[13]** Douglas Walton. Witness Testimony Evidence: Argumentation, Artificial Intelligence, and Law. Cambridge University Press, 2008. **[14]** Clarence Raymond Wylie. 101 puzzles in thought and logic, volume 367. Courier Corporation, 1957. **[15]** Edward N Zalta, Uri Nodelman, Colin Allen, and John Perry. Stanford encyclopedia of philosophy, 2003.

Uncertainty-Based Out-of-Distribution Detection in Deep Reinforcement Learning

Andreas Sedlmeier, Thomas Gabor, Thomy Phan, Lenz Belzner

We consider the problem of detecting out-of-distribution (OOD) samples in deep reinforcement learning. In a value based reinforcement learning setting, we propose to use uncertainty estimation techniques directly on the agent's value estimating neural network to detect OOD samples. The focus of our work lies in analyzing the suitability of approximate Bayesian inference methods and related ensembling techniques that generate uncertainty estimates. Although prior work has shown that dropout-based variational inference techniques and bootstrap-based approaches can be used to model epistemic uncertainty, the suitability for detecting OOD samples in deep reinforcement learning remains an open question. Our results show that uncertainty estimation can be used to differentiate in- from out-of-distribution samples. Over the complete training process of the reinforcement learning agents, bootstrap-based approaches tend to produce more reliable epistemic uncertainty estimates, when compared to dropout-based approaches.

1 Introduction

One of the main impediments to the deployment of machine learning systems in the real world, especially in safety critical areas, is the difficulty to show that the system will continue to reliably produce correct predictions in all the situations it encounters in production use. One of the possible reasons for failure is so called out-of-distribution (OOD) data, i.e. data which deviates substantially from the data encountered during training. As the fundamental problem of limited training data seems unsolvable for most cases, especially in sequential decision making tasks like reinforcement learning, a possible first step towards a solution is to detect and report the occurrence of OOD data. This can prevent silent failures caused by wrong predictions of the machine learning system, for example by handing control over to a human supervisor [1]. In this paper, we propose to use uncertainty estimation techniques in combination with value-based reinforcement learning [27] to detect OOD samples. We focus on deep Q-Learning [20], integrating directly with the agent's valueestimating neural network.

When considering to use uncertainty estimation in order to detect OOD samples, it is important to differentiate two types of uncertainty: aleatoric and epistemic uncertainty. The first type, aleatoric uncertainty models the inherent stochasticity in the system and consequently cannot be reduced by capturing more data. Epistemic uncertainty by contrast arises out of a lack of sufficient data to exactly infer the underlying system's data generating function. Consequently, epistemic uncertainty tends to be higher in areas of low data density. Qazaz [25], who in turn refers to Bishop [2] for the initial conjecture, showed that the epistemic uncertainty σepis(x) is approximately inversely proportional to the density p(x) of the input data, for the case of generalized linear regression models as well as multi-layer neural networks:

$$\sigma_{epis}(x) \propto p^{-1}(x) \qquad (1)$$

This also forms the basis of our proposed method: to use this inverse relation between epistemic uncertainty and data density in order to differentiate in- from out-of-distribution samples.

2 Related Work

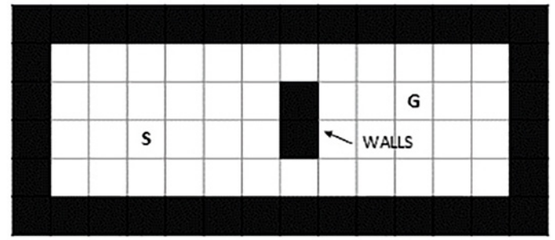
A systematic way to deal with uncertainty is via Bayesian inference. Its combination with neural networks in the form of Bayesian neural networks is realised by placing a probability distribution over the weight-values of the network [19]. As calculating the exact Bayesian posterior quickly becomes computationally intractable for deep models, a popular solution are approximate inference methods [9, 12, 3, 7, 13, 17, 8]. Another option is the construction of model ensembles, e.g., based on the idea of the statistical bootstrap [6]. The resulting distribution of the ensemble predictions can then be used to approximate the uncertainty [22, 15]. Both approaches have been used for tasks as diverse as machine vision [14], disease detection [16], or decision making [5, 22].

For the case of low-dimensional feature spaces, OOD detection (also called novelty detection) is a well-researched problem. For a survey on the topic, see e.g. Pimentel et al. [24], who distinguish between probabilistic, distance-based, reconstruction-based, domain-based and information theoretic methods. During the last years, several new methods based on deep neural networks were proposed for high-dimensional cases, mostly focusing on classification tasks, e.g. image classification. Hendrycks and Gimpel [11] propose a baseline for detecting OOD examples in neural networks, based on the predicted class probabilities of a softmax classifier. Liang et al. [18] improve upon this baseline by using temperature scaling and by adding perturbations to the input. These methods are not directly applicable to our focus, value-based reinforcement learning, where neural networks are used for regression tasks. Other methods, especially generativeneural-network-based techniques [26] could provide a solution, but at the cost of adding separate, additional components. Our approach has the benefit of not needing additional components, as it directly integrates with the neural network used for value estimation.

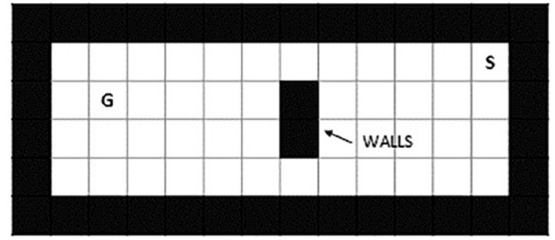
3 Experimental Setup

One of the problems in researching OOD detection for reinforcement learning is the lack of datasets or environments which can be used for generating and assessing OOD samples in a controlled and reproducible way. By contrast to the field of image classification, where benchmark datasets like notMNIST [4] exist that contain OOD samples, there are no equivalent sets for reinforcement learning. As a first step, we developed a simple gridworld environment, which allows modifications after the training process, thus producing OOD states during evaluation.

For our experiments, we focus on a simple gridworld pathfinding environment. During training, the agent starts every episode at a random position in the left half of the 12 × 4 grid space. Its goal is to reach a specific target position in the right half of the grid, which also varies randomly every episode, by choosing one of the four possible actions: {up-,down,left,right}. A vertical set of walls separates the two halves of the environment, acting as static obstacles. Each step of the agent incurs a cost of −1 except the target-reaching action, which is rewarded with +100 and ends the episode. This configuration of the environment is called the train environment. For evaluating the OOD detection performance, we flip the start and goal positions, i.e. the agent starts in the right half of the environment and has to reach a goal position in the left half. This so called mirror environment produces states which the agent has not encountered during training. Consequently, we expect higher epistemic uncertainty values for these OOD states. Figure 1 shows two example initialisations of the train and test environment, with label S marking the start and label G the goal positions.



(a) Example initialisation of the train environment.



(b) Example initialisation of the mirror environment.

Figure 1: Example initializations of the gridworld pathfinding environments called (a) train and (b) mirror. The S label marks the agent's start position, G the goal position. In the train environment, the start position is randomly set in the left half of the environment, in the mirror environment in the right half. The same principle is applied for the goal position: It is randomly set in the right half of the environment for train and the left half of the environment for mirror.

Note that training is solely performed in the train environment. Evaluation runs are executed independently of the training process, based on model snapshots generated at the respective training episodes. Data collected during evaluation runs is not used for training. The state of the environment is represented as a stack of three W ×H feature planes (W being the width, H the height of the grid layout) with each plane representing the spatial positions of all environment objects of a specific type: agent, target or wall.

We compare different neural network architectures and their effect on the reported uncertainty values as the networks are being used by the RL agent for value estimation. The Monte-Carlo Dropout network (MCD) uses dropout variational inference as described by [14]. Our implementation consists of two fully-connected hidden layers with 64 neurons each, followed by two separate neurons in the output layer representing μ and σ of a normal distribution. Before every weight layer in the model, a dropout layer with p = 0.95 is added, specifying the probability that a neuron stays active. Model loss is calculated by minimizing the negative log-likelihood of the predicted output distribution. Epistemic uncertainty as part of the total predictive uncertainty is then calculated according to the following formula:

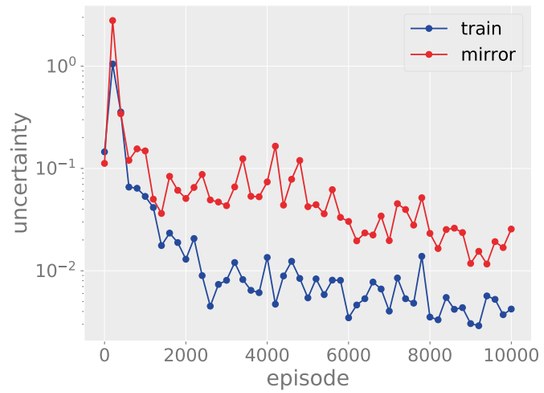
$$\mathrm{Var}_{ep}(y) \approx \frac{1}{T}\sum_{t=1}^{T}\hat{y}_t^2 - (\frac{1}{T}\sum_{t=1}^{T}\hat{y}_t)^2 \qquad (2)$$

with T outputs $\hat{y}_t$ of the Monte-Carlo sampling. Gal et al. [8] suggested an improvement to the default Monte-Carlo dropout method called Concrete Dropout which does not require a prespecified dropout rate and instead learns individual dropout rates per layer. This method is of special interest when used in the context of reinforcement learning, as here the available data change during the training process, rendering a manual optimization of the dropout rate hyperparameter even more difficult. Our implementation of the Monte-Carlo Concrete Dropout network (MCCD) is identical to the MCD network with the exception that every normal dropout layer is replaced by a concrete dropout layer. For both the MCD and MCCD networks, 10 Monte-Carlo forward passes are performed.

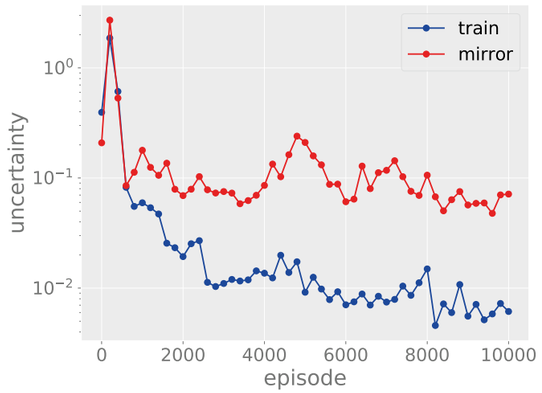
The Bootstrap neural network (BOOT) is based on the architecture described by [22]. It represents an efficient implementation of the bootstrap principle by sharing a set of fully-connected hidden layers between all members of the ensemble. The hidden layers are followed by an output layer of size K, called the bootstrap heads. Our implementation consists of two fully-connected hidden layers with 64 neurons each, which are shared between all heads, followed by an output layer of K = 10 bootstrap heads. For each datapoint, a Boolean mask of length equal to the number of heads is generated, which determines the heads this datapoint is visible to. The mask's values are set by drawing K times from a Bernoulli distribution with p = 0.2.

The Bootstrap-Prior neural network (BOOTP) is based on the extension presented in [21]. It has the same basic architecture as the BOOT network but with the addition of a so-called random Prior Network. Predictions are generated by adding the output of this untrainable prior network to the output of the different bootstrap heads before calculating the loss. Osband et al. [21] conjecture that the addition of this randomized prior function outperforms ensemble-based methods without explicit priors, as for the latter, the initial weights have to act both as prior and training initializer.

For all networks, ReLU is used as the layers' activation function, with the exception of the output layers, where no activation function is used.



(a) BOOT



(b) BOOTP

Figure 2: Per-episode mean uncertainty of chosen actions as predicted by the bootstrap-based methods on the in-distribution train environment (blue) and out-of-distribution mirror environment (red). Reported values are averages of 30 runs. Note: y-axis log-scaled.

4 Results

Figures 2 and 3 present the average uncertainty of the chosen actions over 10000 training episodes of the different network architectures. As there is a certain amount of randomness in the evaluation runs, caused by the random placement of start and goal positions, the plots show averages of 30 evaluation runs.

According to the concept of epistemic uncertainty, we would expect a decline in the absolute value of reported epistemic uncertainty in the train environment over the training process, as the agent collects more data. Interestingly, only the bootstrap-based methods BOOT (Figure 2a) and BOOTP (Figure 2b) reliably show this behaviour. The dropout-based methods do not show consistent behaviour in this regard. For these methods, the predicted uncertainty sometimes even increases along the training process as can be seen in Figure 3b.

Regarding the OOD detection performance, the methods are required to predict higher epistemic uncertainty values for OOD samples than for indistribution samples. Here also, the bootstrapbased methods outperform the dropout-based ones. For all bootstrap methods, over the complete training process, the predicted uncertainty values in the "out-of-distribution" mirror environment are higher than the values in the train environment. Consequently, it would be possible to detect the OOD samples reliably, for example by setting a threshold-based on the lower uncertainty values predicted during training. Figure 2b shows that the addition of a prior has a positive effect on the separation of in- and out-of-distribution samples, as the distance between the predicted uncertainty values increases.

Our results for the dropout-based techniques are not as positive. As can be seen in Figure 3a and 3b, neither of the tested Monte-Carlo dropout methods consistently outputs higher uncertainty values for the OOD states of the mirror environment over the complete training process. Although there are episodes, especially in the beginning, where the mirror environment's uncertainty values are higher, there is a reversal during the training process. As a consequence, it would not be possible to reliably differentiate between in- and out-of-distribution samples at every point in time.

5 Discussion Work and Future

The results we obtained from the bootstrap-based methods show the general feasibility of our approach, as they allow for a reliable differentiation between in- and out-of-distribution samples in the evaluated environments. Declining uncertainty values over the training process also conform to the expectation that epistemic uncertainty can be reduced by collecting more data. For the dropout-based techniques, it remains to be seen if our results show a general problem of these methods in sequential decision problems, or whether the results are a consequence of our specific environments. According to Osband et al. [21] the observed behaviour is to be expected for the basic Monte-Carlo dropout method (MCD) as the dropout distribution does not concentrate with observed data. Consequently, we expected different results from the concrete dropout method (MCCD) as it should be able to adapt to the training data. Nevertheless, this did not lead to decreasing uncertainty estimates over the training process or a reliable prediction of higher uncertainty for OOD samples. We are currently working on extending our evaluations to more environments in order to evaluate if these results generalize. This will include stochastic domains, where it is necessary to differentiate between aleatoric and epistemic uncertainty in order to correctly detect OOD samples. It will also be very interesting to compare the performance of the proposed uncertainty-based methods to methods based on generative models. Another interesting aspect which could further improve the OOD detection performance of the ensemble methods is the choice of prior [10] and a newly proposed method called Bayesian Ensembling [23], which could bridge the gap between fully Bayesian methods and ensembling methods.

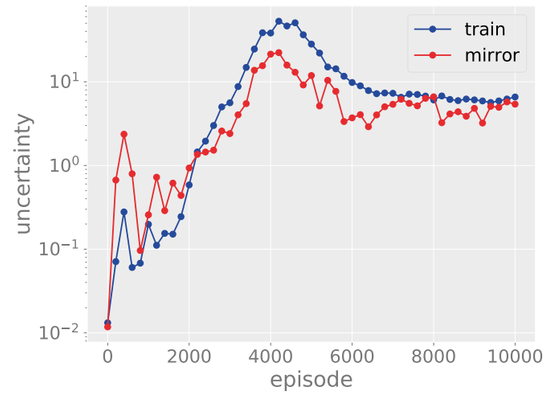
(a) MCD

Andreas Sedlmeier
Andreas Sedlmeier is a researcher at the LMU Munich. His research interests are in the area of artificial intelligence and machine learning. With a focus on questions relating to uncertainty and dependability of autonomous systems, he is researching problems that must be solved so that these systems can be safely deployed in the real world.

Thomas Gabor
Thomas Gabor is a researcher at the LMU Munich. Coming from foundations of optimization techniques in natural computing, he is currently working on topics from both quantum computing and artificial intelligence. He is a fellow of the QARLab and collaborates in the recently announced Plan-QK project, which aims to make quantumsupported AI accessible to industry applications.

Thomy Phan
Thomy Phan is a researcher at the LMU Munich. His research focuses on artificial intelligence and autonomous systems.

Lenz Belzner
Lenz Belzner leads the data science & artificial intelligence team at MaibornWolff. His research focuses on autonomous system engineering and multi-agent reinforcement learning, with applications to operations research, logistics and production. He obtained his PhD at LMU Munich in 2016. As a postdoc, he researched engineering of statistical learning for distributed production systems in collaboration with Siemens and Infineon.



(b) MCCD

Figure 3: Per-episode mean uncertainty of chosen actions as predicted by the dropout-based methods on the in-distribution train environment (blue) and out-of-distribution mirror environment (red). Reported values are averages of 30 runs. Note: y-axis log-scaled.

References: **[1]** D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Man´e. Concrete Problems in AI Safety. ArXiv e-prints, June 2016. **[2]** C. M. Bishop. Novelty detection and neural network validation. IEE Proceedings Vision, Image and Signal Processing, 141(4):217–222, Aug 1994. ISSN 1350-245X. **[3]** C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra. Weight Uncertainty in Neural Networks. ArXiv e-prints, May 2015.**[4]** Yaroslav Bulatov. Notmnist dataset. Google (Books/OCR), Tech. Rep.[Online]. Available: http://yaroslavvb.blogspot.it/2011/09/notmnistdataset.html, 2011. **[5]** S. Depeweg, J. M. Hern´andez-Lobato, F. Doshi-Velez, and S. Udluft. Learning and Policy Search in Stochastic Dynamical Systems with Bayesian Neural Networks. ArXiv e-prints, May 2016. **[6]** Bradley Efron. Bootstrap Methods: Another Look at the Jackknife, pages 569–593. Springer New York, New York, NY, 1992. ISBN 978-1-4612-4380-9. **[7]** Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In international conference on machine learning, pages 1050–1059, 2016. **[8]** Yarin Gal, Jiri Hron, and Alex Kendall. Concrete dropout. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, Advances in Neural Information Processing Systems 30, pages 3581–3590. Curran Associates, Inc., 2017. **[9]** Alex Graves. Practical variational inference for neural networks. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, Advances in Neural Information Processing Systems 24, pages 2348–2356. Curran Associates, Inc., 2011. **[10]** D. Hafner, D. Tran, A. Irpan, T. Lillicrap, and J. Davidson. Reliable Uncertainty Estimates in Deep Neural Networks using Noise Contrastive Priors. ArXiv e-prints, July 2018. **[11]** D. Hendrycks and K. Gimpel. A Baseline for Detecting Misclassified and Out-ofDistribution Examples in Neural Networks. ArXiv e-prints, October 2016. **[12]** J. M. Hern´andez-Lobato and R. P. Adams. Probabilistic Backpropagation for Scalable Learning of Bayesian Neural Networks. ArXiv e-prints, February 2015. **[13]** JM Herna´ndez-Lobato, Y Li, M Rowland, D Hern´andez-Lobato, TD Bui, and RE Ttarner. Black-box α-divergence minimization. In 33rd International Conference on Machine Learning, ICML 2016, volume 4, pages 2256–2273, 2016. **[14]** Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, Advances in Neural Information Processing Systems 30, pages 5574–5584. Curran Associates, Inc., 2017. **[15]** Bala ji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, Advances in Neural Information Processing Systems 30, pages 6402–6413. Curran Associates, Inc., 2017. **[16]** Christian Leibig, Vaneeda Allken, Murat Se¸ckin Ayhan, Philipp Berens, and Siegfried Wahl. Leveraging uncertainty information from deep neural networks for disease detection. Scientific reports, 7(1):17816, 2017. **[17]** Y. Li and Y. Gal. Dropout Inference in Bayesian Neural Networks with Alphadivergences. ArXiv e-prints, March 2017. **[18]** S. Liang, Y. Li, and R. Srikant. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. ArXiv eprints, June 2017. **[19]** David J. C. MacKay. A practical bayesian framework for backpropagation networks. Neural Computation, 4:448–472, 1992. **[20]** V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller. Playing Atari with Deep Reinforcement Learning. ArXiv e-prints, December 2013. **[21]** I. Osband, J. Aslanides, and A. Cassirer. Randomized Prior Functions for Deep Reinforcement Learning. ArXiv e-prints, June 2018. **[22]** Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems 29, pages 4026–4034. Curran Associates, Inc., 2016. **[23]** T. Pearce, M. Zaki, A. Brintrup, and A. Neel. Uncertainty in Neural Networks: Bayesian Ensembling. ArXiv e-prints, October 2018. **[24]** Marco A.F. Pimentel, David A. Clifton, Lei Clifton, and Lionel Tarassenko. A review of novelty detection. Signal Processing, 99:215 – 249, 2014. ISSN 0165-1684. **[25]** Cazhaow S. Qazaz. Bayesian error bars for regression. PhD thesis, Aston University, 1996. **[26]** Thomas Schlegl, Philipp Seeb¨ock, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In IPMI, 2017. **[27]** Richard S Sutton and Andrew G Barto. Introduction to reinforcement learning, volume 135. MIT press Cambridge, 1998.

Joint Action Learning for Multi-Agent Cooperation using Recurrent Reinforcement Learning

Jorrit E. Posor, Lenz Belzner, Alexander Knapp

Many approaches in applying Q-learning to multi-agent Systems consider other agents as part of the environment and use one shared policy network for all agents. These independent learners (IL) show good results for noncooperative tasks. For cooperative tasks, IL report convergence to good but not always optimal policies. Studies show that better performance can be achieved through agents that learn the effect of joint actions in many scenarios like cooperative games. We propose a recurrent joint action learner (RJAL) using Q-learning and a recurrent neural network and evaluate it in various games against IL. The games are designed to require cooperation and the constant action of all agents at different levels. Our evaluation shows that RJAL can use cooperation to dominate IL in a mixed cooperative-competitive battle game. On the other hand, since IL does not suffer from lazy agent behavior, it leads in the case of the pursuit game that punishes lazy agents directly.

1 Introduction

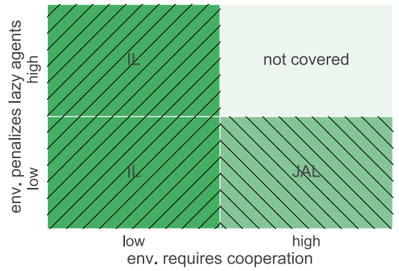


Figure 1: Independent learners (IL) perform well in environments where cooperation can not be leveraged. Furthermore, IL are not prone to the "lazy agent" problem. A joint action learner (JAL), shows superior performance when cooperation is needed.

In addition to successes in single-agent reinforcement learning, interest in cooperative multiagent reinforcement learning (MARL) is increasing. Recent work in the field of cooperative MARL has dealt with independent learning, an approach originating from noncooperative single-agent reinforcement learning, and applied it to cooperative settings [25, 3]. Further work uses centralized approaches [6, 28, 22, 18], hybrid approaches [5, 7], and value decomposition networks [24, 19]. Due to incomplete or noisy observation capabilities of an agent, partial observability of the environments state is a realistic setup for real-world applications of MARL [17]. Hence, we propose an approach that operates in a partially observable environment and avoids the use of the true state of the environment. We show the proposed recurrent joint action learner (RJAL) forms superior cooperation strategies under these circumstances. The evaluation of competitive games against an independent Q-learner [26] reports high win rates for RJAL although independent Q-learning is less affected by lazy (i.e. standing idle for multiple steps) agent behavior [24] than RJAL. To evaluate this, we use games that require cooperation to varying degrees or punish lazy agents to varying degrees.

2 Related Work

Independent Learning applies approaches to single-agent reinforcement learning by handling each agent of the multi-agent environment as a single agent. Other agents are usually treated as part of the environment. The actions that other agents perform are not explicitly taken into account. A commonly applied method is *Independent Q-learning* [26]. Through Q-learning [29] or using a deep Q-network [16], decentralized action-value functions are learned independently for each agent [25]. Unfortunately, concurrent exploring and learning of independent agents make the environment non-stationary and break theoretical convergence guarantees [15, 4]. Nevertheless, the approach of decentralized independent learning has been successfully applied for various tasks like mixed and competitive games [25, 13]. Deep distributed recurrent Q-networks were successfully used to solve communication-based coordination tasks [3]. The network weights are shared among all agents who operate in a partially observable environment. To approximate agents' actionobservation history, the network input does also contain the last action of each agent.

Centralized Learning of Joint Actions mitigates the non-sta-

tionarity issue in multi-agent settings, but many approaches are limited to a certain number of agents [19]. This limitation has its origin in the fact that joint action spaces grow exponentially with the number of agents [17]. Q-learning using the joint action space converges to an optimal solution if the state of the environment is known to all agents [28]. For Dec-POMDPs, where the state is unknown, no methods that guarantee convergence have been established yet [17]. In [6] the joint value function is approximated as a linear combination of local value functions called agentbased decomposition. An agents' local value function depends on the action selection of neighboring agents in a coordination graph to enable cooperative action selection. Thereby the global coordination problem is reduced to several local coordination problems. For densely connected graphs, the size of local value functions still grows exponentially with the number of neighbors [11]. Another limitation of this approach is that the dependencies between agents need to be pre-supplied.

With an edge-based decomposition of the joint value function, local value functions are no longer based on individual agents' actions and those of neighboring ones but the actions of the two agents forming an edge [11]. This approach scales linearly with the number of neighbors but, similar to the work of [6], it also depends on pre-supplied information to create the coordination graph. Traffic light control is an application for which coordination graphs were used to estimate joint actions by sending locally optimized messages among connected agents [1]. A bidirectional RNN in an actor-critic [12] architecture can be used to enable communication between agents [18]. This approach yields better performance than independent learning. Unlike this paper, [18] uses the environment state instead of partial observations.

In [22], a centralized network that takes individual observations of all agents as input is used to form joint action. Gradients to the model are derived through a policy gradient. Evaluations on diverse tasks show that the model outperforms independent approaches.

Besides centralized and independent approaches, *Hybrid Approaches* employ both concepts. They learn joint actions in a centralized way and train another policy that executes actions in a decentralized manner. COMA [5] is an actor-critic [12] approach where a single centralized critic for all agents is used during training and decentralized actors are used during execution. An agents' actor conditions on its action-observation history to compute actions and the centralized critic conditions on joint actions and the environments state to estimate Q-values. The actors are optimized by following a gradient estimated by the critic. Furthermore the centralized critic is used to mitigate the credit assignment problem by implementing difference rewards [27, 30]. An approach similar to COMA is [14]. The key difference is that it uses a centralized critic for each agent and considers environments with explicit communication between agents. Further, COMA operates in environments with communication constraints and learns continuous policies rather than discrete policies. However, on-policy learning, like the two approaches above, can be sample-inefficient, and the centralized critic can become a bottleneck with an increasing number of agents [19].

Value Decomposition Networks (VDNs) [24] sum up individual value functions to represent a joint action-value function. To optimize the individual value functions, VDNs utilize losses calculated with the joint action-value function. To address the credit assignment problem and resulting lazy agent behavior, a VDN is used in the scope of [24]. The proposed architecture is based on a DQN [16] and strongly outperforms centralized and decentralized methods in their experiments. Since the VDN of [24] can represent only a limited class of centralized action-value functions, QMIX [19] is developed to represent a much richer class of action-value functions. QMIX applies a mixing network for this goal, which maintains a certain restriction regarding the monotonous relationship of individual Q-values and the sum of Q-values. QMIX improves the performance over [24] as well as independent Q-learning.

3 Background

A cooperative multi-agent task in an environment with a limited view range per agent can be modeled as decentralized partially observable Markov decision process (Dec-POMDP) [17]. A Dec-POMDP consists of a tuple $G = \langle A,S,U,T,O,O,R,h,\gamma \rangle$.

$A = \{1,...,n\}$ is the set of n agents. $s \in S$ is the state of the environment. At each time step t, each agent $a \in A$ performs an action $u_t^a \in U$. All agents of a team form the joint action $\mathbf{u} \in \mathbf{U} \equiv U^n$. According to the transition probability function T that specifies $P(s_{t+1}|s_t,\mathbf{u}) : S \times \mathbf{U} \times S \rightarrow [0,1]$, the environment transitions into the next state s_{t+1}. For each state s_t, each agent a receives an observation $o_t^a \in O$ according to the observation function $O(s,a) : S \times A \rightarrow O$. $R(s,\mathbf{u}) : S \times \mathbf{U} \rightarrow \mathrm{R}$ is a reward function, shared among all agents. h is the horizon, specifying the number of time steps during which the agent will interact with the environment before it terminates. $\gamma \in [0,1)$ is a discount factor. Each agent conditions a deterministic policy $\pi^a : \overline{O}^a \rightarrow U$ on its observation history $\overline{o}_t^a = (o_0^a, \ldots, o_t^a)$.

4 Joint Action Learning

We propose a joint action learner that consists of an RNN that maps an input sequence of spatial observations and non-spatial features to an output sequence of Q-values. The central idea of this approach is that the individual Q-values condition not only on the hidden states of the RNN but also on the Q-values of the previous agent.

At each time step t, the input sequence contains n elements. For agent $a \in A$, (o_t^a, f_t^a) is an element of the input sequence, where o_t^a is the spatial individual observation. Non-spatial individual features are defined as:

$$f_t^a = (u_{t-1}^a, r_{t-1}^a, a, p_t^a, Q_t^{a-1}) \qquad (1)$$

where u_{t-1}^a and r_{t-1}^a denote the last action and last reward, respectively. a is the agent ID and p^a is the relative position. Q_t^{a-1} denotes the Q-values calculated for agent $a-1$.

Entailing the last action u_{t-1}^a into the input sequence for an RNN, is a practical approach to approximate action-observation histories [3, 8, 24]. Although agents who follow a deterministic policy can derive their actions from their observation history alone [17], we add the past actions to the input sequence as this stabilizes the training of the recurrent architecture in our case. The order of the input elements is fixed and determined by the list of agents (1,...,n). Figure 2 shows the RNN,

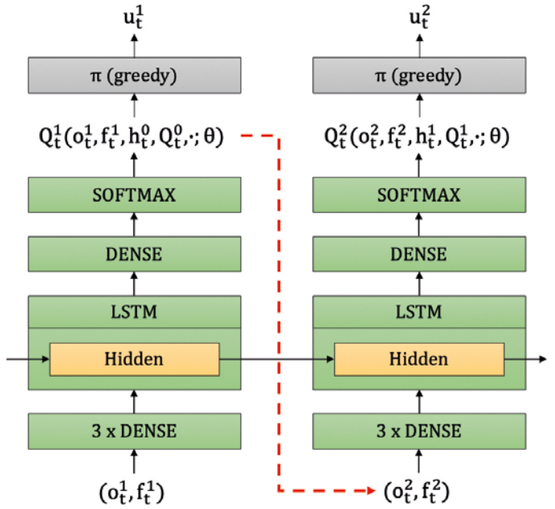


Figure 2: The unrolled network of the joint action learner for two agents. The features f contain the Q-values of the previous agent (dashed arrow). Q^0 is initialized with uniformly distributed values. u_t^1 and u_t^2 form a joint action. The RNN uses LSTM

unrolled over two agents. The red arrow shows the passing on of the Q-values to the following agent.

For each agent $a \in A \equiv \{1,...,n\}$, at time step t, the output sequence contains Q-values $Q_t^a(o_t^a, f_t^a, h_t^{a-1}, Q_t^{a-1}, \cdot; \theta)$, conditioned on the inputs o_t^a and f_t^a, the hidden state h_t^{a-1} of the preceding agent, the state action-values Q_t^{a-1} of the preceding agent and network parameters θ. Due to weight sharing between agents, the network parameters θ do not condition on a. To select actions $(u_t^1,...,u_t^n)$, that will be carried out to transition into the next state, $\mathrm{argmax}_{\mathbf{u}}\, Q_t^a(o_t^a, f_t^a, h_t^{a-1}, Q_t^{a-1}, u; \theta)$ is performed for all agents.

Losses are calculated slightly different to the standard DQN loss [16], because we distinguish between global and local rewards for the evaluation and do not exclusively use squared differences. The joint action learner is trained by minimizing following loss for batches of size B:

$$loss(v_t^a, v_t^{a;\mathrm{target}}) = \sum_{b=1}^{B} \sum_{a=1}^{n} z^a \qquad (2)$$

where z^a is the Huber loss [10] function:

$$z^a = \begin{cases} 0.5(v_t^a - v_t^{a;\mathrm{target}})^2 & \text{if } |v_t^a - v_t^{a;\mathrm{target}}| < 1 \\ |v_t^a - v_t^{a;\mathrm{target}}| - 0.5 & \text{otherwise} \end{cases} \qquad (3)$$

The target values $v_t^{a;\mathrm{target}}$, are computed by:

$$v_t^{a;\mathrm{target}} = GR + \gamma v_{t+1}^{a;-} \qquad (4)$$

In case of a training on local rewards, GR is replaced by reward r_t^a for agent a. Correlations between Q-values and target values can lead to a unstable training [16]. Therefore, we utilize a copy of our original network whose network parameters θ^- are frozen and updated every 5 episodes by copying the parameters θ of the original counterpart [16]. $v_{t+1}^{a;-}$ denotes the value, calculated by the frozen network (θ^-) and selected by the action v_{t+1}^a, that the original network (θ) would have selected for the same observation:

$$v_{t+1}^{a;-} = Q_{t+1}^a(o_{t+1}^a, f_{t+1}^a, h_{t+1}^{a-1}, u_{t+1}^a; \theta^-) \qquad (5)$$

5 Experimental Setup

This section describes the three partially observable games used to evaluate RJAL against an IL.

As an environment, we use MAgent [32], an open-source research platform with a focus on MARL. MAgent builds on a gridworld as the fundamental environment and provides a reward description language and agent configurations to enable a flexible game design. Like previous the work [31], we use this environment to evaluate our approach against an IL.

For the evaluation, we use three games, which are implemented using MAgent. The three games are designed to highlight the ability of the proposed algorithms to cooperate and to point out proneness to lazy agent problems.

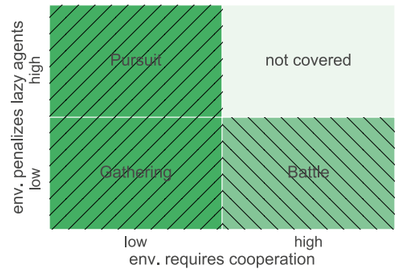


Figure 3: A qualitative classification of the games with regard to cooperation and lazy agents.

The goal of the gathering game is to capture randomly moving prey within a maximum of 400 steps. Hence agents of the predator group receive reward through attacking and destroying the prey. The predator group is controlled by RJAL or IL, while the other group ignores the reward setting and samples actions from a uniform distribution over the action space to perform random actions. Since the predator group receives a negative reward for every step, it can maximize its total reward by destroying the opponents as fast as possible. As Figure 3 illustrates, the gathering game entails low penalties for lazy agents and requires low amounts of cooperation. Therefore independent learners are expected to find suitable solutions for this game. A joint action learner should also achieve at least equivalent results, as this game does not have properties that can be exploited explicitly by IL.

The battle game is a hybrid of cooperation and competition that consists of two agent groups [32]. The algorithm that destroys all agents of the opponent wins the game. Both agent groups share the same attributes and reward setting. The ability

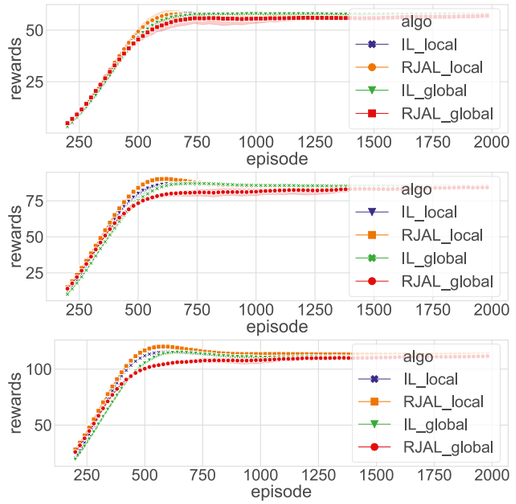


Figure 4: Gathering 4 vs. 8, 6 vs. 12 and 8 vs. 16 (downwards). Averaged total reward plus corresponding 95% confidence intervals (shaded areas) of ten training runs per algorithm over the course of 2000 episodes.

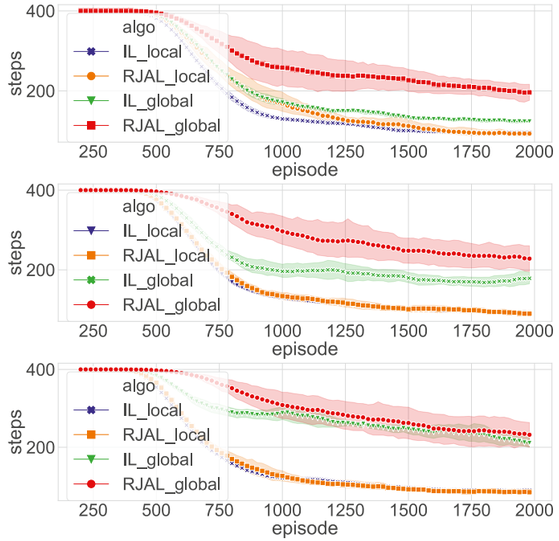


Figure 5: Gathering 4 vs. 8, 6 vs. 12 and 8 vs. 16 (downwards). Averaged total steps plus corresponding 95% confidence intervals (shaded areas) of ten training runs per algorithm over the course of 2000 episodes. A gathering game stops after 400 steps, or as soon as the entire prey has been caught.

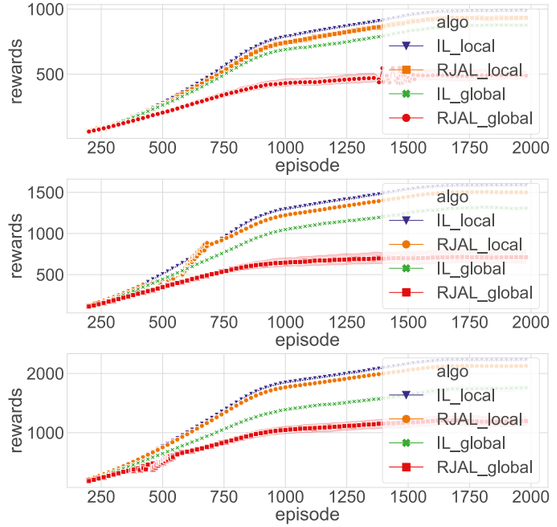


Figure 6: Pursuit 4 vs. 8, 6 vs. 12 and 8 vs. 16 (downwards). Averaged total rewards across setups over the course of 2000 episodes. Shaded areas show the 95% confidence intervals of ten training runs per algorithm.

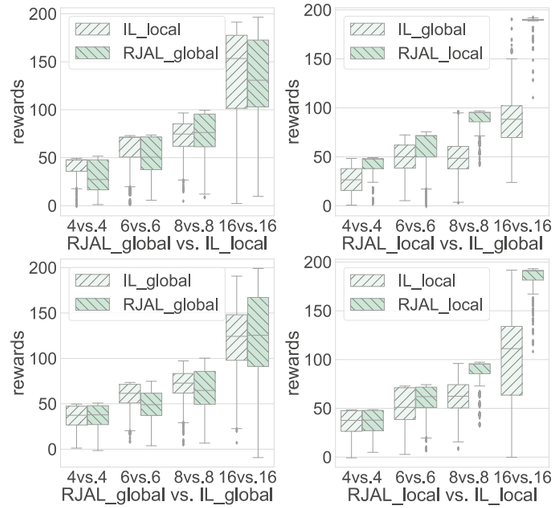


Figure 7: Rewards for the battle games.

to cooperate, for instance, through coordinated attacks and positioning in formations, is beneficial in the case of the battle game. Even if a small fraction of the agents in a team shows lazy behavior, the ability to cooperate and the game design are expected to mitigate this issue. Both circumstances should result in a superior performance for a joint action learner. By increasing the number of the agents in a team, more teammates of an individual agent are out of its range. Therefore, it should become more difficult to deduce which observations the teammates have at a specific time step only from the individual observation of an agent. Hence, we expect the performance difference between RJAL and IL to increase with the number of agents.

Similar to the gathering game, the pursuit game consists of a predator and a prey agent group. Again, the prey group chooses its actions based on uniformly sampling from the action space. The prey can't be destroyed in this game. The only way to generate rewards is to hit the prey as often as possible. Hence, to maximize the total reward, the predator group has to follow and hit the prey consistently. Consequently, the pursuit game directly penalizes lazy agents. All agents are required to hit the prey throughout a game continually, and therefore every lazy agent instantly minimizes the maximum achievable total reward. We expect that there will always be a significant performance difference between RJAL and IL in favor of IL.

For the Gathering and Pursuit game, RJAL and the IL are trained against randomly moving prey in three setups. The setups consist of 4, 6, or 8 agents controlled by RJAL/IL. The prey consists of twice the number of agents. Within the battle game, RJAL/IL are first trained in self-play against the same number of agents. For the evaluation, RJAL and IL play against each other in teams of the same size. The setups for the battle game consist of 4, 6, 8, and 16 agents. Each setup is performed on a map size of 20 times 20. At each time step, an agent can perform one of nine actions. After the execution of actions, each agent receives a local (individual) reward. RJAL and IL are trained and evaluated using both local and global rewards. A global reward corresponds to the sum of the individual rewards of the agents. Local rewards correspond to the individual rewards of the agents. The reward setting is: 5 when destroying an enemy/prey, 0.2 when hitting an enemy/prey, −0.5 when being destroyed, −0.1 when performing an attack, −0.005 for every move. Agents can move to horizontally and vertically neighboring hexes or attack these hexes. In combination with the idle action, this results in an action space action of size 9. Partial observability is realized by the restricted field of vision of the agents. At any time, an agent can only access one observation of dimension $[9 \times 9 \times 7]$.

6 Results

Figures 4 and 5 show the averaged total rewards and averaged total steps for various numbers of agents (4, 6, 8) controlled by RJAL/IL and corresponding preys (8, 12, 16) in the gathering game. The suffixes local and global mark the algorithms that are trained on the individual or global rewards, respectively. Ten training runs are conducted for each of the four algorithms, respectively. A training run consists of 2000 episodes. As Figure 4 shows, all algorithms converge to nearly the same values regarding total reward. Furthermore, total rewards' standard deviations do not significantly differ. IL local and RJAL local can destroy the prey faster than IL global, followed by RJAL global. Total steps' standard deviations of both global approaches are significantly higher than the standard deviations of the local approaches.

Ten training runs encompassing 2000 episodes per setup for every algorithm (IL local, RJAL local, IL global, and RJAL global) are conducted for the pursuit game. Each episode terminates at the specified maximum episode length of 400 steps. Consequently, we don't report results regarding total steps for this game. In contrast to the gathering game, IL local outperforms all other algorithms across setups. Figure 6 clearly illustrates a systematic gap in the total rewards. The approaches trained on global reward become worse when the number of agents is increased. Although RJAL local also suffers from lazy agents, its results with additional agents remain stable.

Since the optimal solution requires all agents to hit prey every step, this game is useful revealing lazy agent [24] behavior. Evaluating replay videos of the 8 vs. 16 setup, we observed that regularly up to four (depending on the algorithm) agents show lazy behaviors. This observation explains the systematic difference of total reward between the algorithms. Centralized approaches are eventually prone to the lazy agent problem [24]. Some agents might be discouraged from learning if random actions worsen the total reward of the team. As a result, a suboptimal policy might be learned. Data and visual analyses support this claim. Furthermore, the lazy agent problem intensifies when an algorithm is trained on global rewards or when the number of agents increases. If the global reward is used to calculate the target values, information may be lost that would otherwise be helpful to evaluate the individual observations of the agents differently.

The pursuit game shows that RJAL cannot reach the performance of the individual learner when lazy agents are severely punished. On the other hand, the gathering game clarifies that RJAL can achieve an equal reward when relaxing the punishment for lazy agents.

Figure 7 shows the rewards achieved in the battle game for different combinations of approaches trained on local and global rewards. Following previous work [20, 21, 23, 14, 2], the corresponding approaches were trained using self-play before evaluation. RJAL global fails to form superior cooperation strategies. Therefore both IL approaches slightly dominate RJAL global for most setups. Visual analyses of the games entailing RJAL local reveal that this approach behaves in a more coordinated manner. It stays in a formation longer and attacks opponents at the same time. Hence, RJAL local outper-

		Setup (# learning agents)			
	Algorithm	4	6	8	16
Gathering	IL_local	**1.3**	**2.4**	**2.8**	
	RJAL_local	7.1	10.7	8.0	
	IL_global	1.5	3.5	4.6	
	RJAL_global	6.7	27.2	13.5	
Battle	IL_local	**1.1**	**0.7**	**1.7**	2.1
	RJAL_local	1.8	2.1	2.6	2.6
	IL_global	1.4	1.5	2.0	**1.7**
	RJAL_global	6.2	3.3	2.8	5.0
Pursuit	IL_local	**4.5**	5.5	**6.2**	
	RJAL_local	9.5	13.7	43.9	
	IL_global	**4.5**	**5.4**	**6.2**	
	RJAL_global	25.1	14.3	17.1	

Figure 8: Averaged training durations (hours) for a training run over the course of 2000 episodes.

forms IL global and IL local in all setups. This result becomes clearer as more agents are part of a team. RJAL requires more training time than the IL for all games. This is mainly due to the sequential calculation of the Qvalues. Figure 8 shows the corresponding training durations for all approaches.

7 Conclusion

This paper presented an approach to joint action learning for MARL and compared it to a state of the art approach. Our approach is based on an RNN and the additional transfer of Q-values to other agents in strict order. RJAL was evaluated using three games to demonstrate the influence of lazy agents and cooperation. The results show that RJAL is particularly suitable for problems that require cooperation. Although the pursuit game shows that RJAL suffers from lazy agents, RJAL (local) wins 80.0% (against IL local) and 93.1% (against IL global) for the largest setup (16 vs. 16) in the battle game. In addition, the gathering game has shown that RJAL delivers similar performance to IL when lazy agents are not punished and cooperation can only be marginally exploited. However, in this case IL should be chosen due to the shorter training times. For future work, we plan a more dynamic distribution of the Q-values, as well as the implementation of a joint action learner, which includes the latest breakthroughs of transformer architectures from the NLP domain.

Acknowledgements

We would like to thank Thomy Phan for helpful comments and discussion.

References **[1]** Bram Bakker, Shimon Whiteson, Leon Kester, and Frans CA Groen. Traffic light control by multiagent reinforcement learning systems. In Interactive Collaborative Information Systems, pages 475–510. Springer, 2010. **[2]** Trapit Bansal, Jakub Pachocki, Szymon Sidor, Ilya Sutskever, and Igor Mordatch. Emergent complexity via multi-agent competition. arXiv preprint arXiv:1710.03748, 2017. **[3]** Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. Learning to communicate with deep multiagent reinforcement learning. In Advances in Neural Information Processing Systems, pages 2137–2145, 2016. **[4]** Jakob Foerster, Nantas Nardelli, Gregory Farquhar, Triantafyllos Afouras, Philip HS Torr, Pushmeet Kohli, and Shimon Whiteson. Stabilising experience replay for deep multiagent reinforcement learning. In Proceedings of the 34th International Conference on Machine Learning-Volume 70, pages 1146–1155. JMLR. org, 2017. **[5]** Jakob N Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In Thirty-Second AAAI Conference on Artificial Intelligence, 2018. **[6]** Carlos Guestrin, Daphne Koller, and Ronald Parr. Multiagent planning with factored MDPs. In Advances in neural information processing systems, pages 1523–1530, 2002. **[7]** Jayesh K. Gupta, Maxim Egorov, and Mykel Kochenderfer. Cooperative multi-agent control using deep reinforcement learning. In Autonomous Agents and Multiagent Systems, volume 10642, pages 66–83. Springer International Publishing, 2017. **[8]** Matthew Hausknecht and Peter Stone. Deep recurrent Q-learning for partially observable MDPs. In 2015 AAAI Fall Symposium Series, 2015. **[9]** Sepp Hochreiter and Ju¨rgen Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997. **[10]** Peter J Huber. Robust estimation of a location parameter. In Breakthroughs in statistics, pages 492–518. Springer, 1992. **[11]** Jelle R Kok and Nikos Vlassis. Collaborative multiagent reinforcement learning by payoff propagation. Journal of Machine Learning Research, 7(Sep):1789–1828, 2006. **[12]** Vijay R Konda and John N Tsitsiklis. Actorcritic algorithms. In Advances in neural information processing systems, pages 1008–1014, 2000. **[13]** Joel Z Leibo, Vinicius Zambaldi, Marc Lanctot, Janusz Marecki, and Thore Graepel. Multi-agent reinforcement learning in sequential social dilemmas. In Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, pages 464–473. International Foundation for Autonomous Agents and Multiagent Systems, 2017. **[14]** Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In Advances in Neural Information Processing Systems, pages 6379–6390, 2017. **[15]** Laetitia Matignon, Guillaume J Laurent, and Nadine Le Fort-Piat. Independent reinforcement learners in cooperative markov games: a survey regarding coordination problems. The Knowledge Engineering Review, 27(1):1–31, 2012. **[16]** Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. Nature, 518(7540):529, 2015. **[17]** Frans A Oliehoek, Christopher Amato, et al. A concise introduction to decentralized POMDPs, volume 1. Springer, 2016. **[18]** Peng Peng, Ying Wen, Yaodong Yang, Quan Yuan, Zhenkun Tang, Haitao Long, and Jun Wang. Multiagent bidirectionally-coordinated nets: Emergence of humanlevel coordination in learning to play starcraft combat games. arXiv preprint arXiv:1703.10069, 2017. **[19]** Tabish Rashid, Mikayel Samvelyan, Christian Schroeder De Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. QMIX: monotonic value function factorisation for deep multi-agent reinforcement learning. arXiv preprint arXiv:1803.11485, 2018. **[20]** Martin Riedmiller, Thomas Gabel, Roland Hafner, and Sascha Lange. Reinforcement learning for robot soccer. Autonomous Robots, 27(1):55–73, 2009. **[21]** David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of Go with deep neural networks and tree search. nature, 529(7587):484, 2016. **[22]** Sainbayar Sukhbaatar, Rob Fergus, et al. Learning multiagent communication with backpropagation. In Advances in Neural Information Processing Systems, pages 2244– 2252, 2016. **[23]** Sainbayar Sukhbaatar, Zeming Lin, Ilya Kostrikov, Gabriel Synnaeve, Arthur Szlam, and Rob Fergus. Intrinsic motivation and automatic curricula via asymmetric self-play. arXiv preprint arXiv:1703.05407, 2017. **[24]** Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z Leibo, Karl Tuyls, et al. Value-decomposition networks for cooperative multi-agent learning based on team reward. In Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, pages 2085–2087. International Foundation for Autonomous Agents and Multiagent Systems, 2018. **[25]** Ardi Tampuu, Tambet Matiisen, Dorian Kodelja, Ilya Kuzovkin, Kristjan Korjus, Juhan Aru, Jaan Aru, and Raul Vicente. Multiagent cooperation and competition with deep reinforcement learning. PLOS ONE, 12(4):1–15, 2017. **[26]** Ming Tan. Multi-agent reinforcement learning: Independent vs. cooperative agents. In Proceedings of the tenth international conference on machine learning, pages 330–337, 1993. **[27]** Kagan Tumer and Adrian Agogino. Distributed agent-based air traffic flow management. In Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems, page 255. ACM, 2007. **[28]** Nikos Vlassis. A concise introduction to multiagent systems and distributed artificial intelligence. Synthesis Lectures on Artificial Intelligence and Machine Learning, 1(1):1–71, 2007. **[29]** Christopher John Cornish Hellaby Watkins. Learning from delayed rewards. King’s College, Cambridge, 1989. **[30]** David H Wolpert and Kagan Tumer. Optimal payoff functions for members of collectives. In Modeling complexity in economic and social systems, pages 355–369. World Scientific, 2002. **[31]** Yaodong Yang, Rui Luo, Minne Li, Ming Zhou, Weinan Zhang, and Jun Wang. Mean field multi-agent reinforcement learning. arXiv preprint arXiv:1802.05438, 2018. **[32]** Lianmin Zheng, Jiacheng Yang, Han Cai, Ming Zhou, Weinan Zhang, Jun Wang, and Yong Yu. MAgent: A many-agent reinforcement learning platform for artificial collective intelligence. In Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

Jorrit E. Posor
Jorrit E. Posor studied information systems at the Technical University of Darmstadt and the University of Augsburg. Since 2018 he has been working in the data science & artificial intelligence team at MaibornWolff as a machine learning engineer.

Dr. Lenz Belzner
Dr. Lenz Belzner leads the data science & artificial intelligence team at MaibornWolff. His research focuses on autonomous system engineering and multi-agent reinforcement learning, with applications to operations research, logistics and production. He obtained his PhD at LMU Munich in 2016. As a postdoc, he researched engineering of statistical learning for distributed production systems in collaboration with Siemens and Infineon.

Dr. Alexander Knapp
Dr. Alexander Knapp works on formal methods, semantics of modelling and programming languages, constraint optimisation, and the interplay between quality assurance and machine learning. In 2000 he received his PhD from the LudwigMaximilians-Universit¨at Mu¨nchen. Since 2007 he is professor for the foundations of software and systems engineering at Augsburg University.

Distance-based classifier on the **Quantum Inspire**

Robert Wezeman, Niels Neumann, Frank Phillipson

With quantum computers entering a new phase of maturity and quantum algorithms covering a growing range of applications, it is important that the two meet at some point in the middle.

Currently, the two meet only by toy-problem implementations for some algorithms. In this work, we extend a recently proposed distance-based classifier and give explicit quantum circuit implementations, and we give a method to overcome restrictions on the data, present in the original work.

1 Introduction

Extracting meaningful information from data seems simple. Even making decisions based on this information appears easy, we humans do it constantly. However, training a computer or an algorithm to do precisely that, is in general more complex. Machine learning deals precisely with this topic: Learning a computer to extract meaningful information from data and in some cases take subsequent actions based on it.

Examples of problems especially well suited for machine learning are pattern detection [1], image classification [2] and anomaly detection [3]. However, machine learning can also be used to play games such as chess [4] and more recently the game of Go [5]. The machine learning algorithm does not exist. Instead, it is a class of algorithms, each working in its own specific way, aimed at solving specific types of problems it is optimized for.

There are, however, also challenges within machine learning that might limit its applicability on today’s computers. For instance, the large training times or high data demands. Quantum computing might give a solution, as preliminary work already showed that quantum computing can give an exponential or quadratic speed-up [6, 7] and training might require less training samples [8, 9]. Even though the available quantum hardware is not yet sufficiently mature, theoretical results are already available [10]. Also, implementations on noisy intermediate-scale quantum (NISQ) computers are already possible and give a good insight in the potential of quantum computing [11]. An example of such an algorithm is a four qubit distance-based classifier proposed in [12], where a test point is classified in one of two classes.

In this work we also focus on the distance-based classifier and give extensions on the implementations needed for larger problem instances than considered in the original work. Some assumptions on the data set are made in the original work, for instance that the number of data points is a power of two and only two classes are possible. Our work will give solutions for those assumptions and will give explicit quantum circuits for them. The extensions are implemented on the Quantum Inspire platform [13] and can be found on Github [14].

In the remainder of this paper we explain the algorithm and give an implementation on the Quantum Inspire platform [13] in Section 2. Afterwards, in Section 3 we explain how to deal with data sets with larger feature spaces. Section 4 explains how to deal with with arbitrary number of data points. Section 5 gives an approach for the situation where the data has more than two classes or when the classes are unbalanced. Section 6 explains how to deal with data sets consisting of concentric circles, as the original work in [12] does not work for such data sets. Final remarks are given in Section 7.

2 Problem description

In this section we first explain how the distancebased classifier works, after which we give an implementation on the Quantum Inspire platform.

2.1 The distance-based classifier

Pattern recognition and image classification are tasks well-suited for machine learning. In this work we consider a supervised image classification algorithm with binary input images as first formulated in [12]. The original algorithm considered a distance-based classifier taking M data points $\mathbf{x}_i \in \mathrm{R}^N$ with binary labels $y_i \in \{-1,1\}$. The task is to assign a label $\tilde{y}$ to a new data point $\tilde{\mathbf{x}}$ based on the M data points. Classification is done using the classification function

$$\tilde{y} = \operatorname{sgn}\left(\sum_{m=0}^{M-1} y_m \left[1 - \frac{1}{4M}\left|\tilde{\boldsymbol{x}} - \boldsymbol{x}_m\right|^2\right]\right). \quad (1)$$

Note that this function uses a kernel method [15]. Kernel methods are used to classify data points based on some definition of similarity, distance in this case, with closer points being more similar.

Here, $\left[1 - \frac{1}{4M}|\tilde{\boldsymbol{x}} - \boldsymbol{x}_m|^2\right]$ is the kernel method. The classifier can be rewritten to

$$\tilde{y} = \operatorname{sgn}\left(\sum_{m=0}^{M-1} y_m \left|\tilde{\boldsymbol{x}} + \boldsymbol{x}_m\right|^2\right), \qquad (2)$$

as long as the data is normalized and both classes contain the same number of data points, meaning M/2 data points per class and hence $\sum y_m = 0$.

Each data point can be encoded in qubits. For data points of length N, we use $n_d = [\log_2 N]$ qubits to encode the data point:

$$\boldsymbol{x} = (x_0, \ldots, x_{N-1})^{\mathrm{T}} \mapsto |\psi_{\boldsymbol{x}}\rangle = \frac{1}{\|\boldsymbol{x}\|^2} \sum_{i=0}^{N-1} x_i |i\rangle, \qquad (3)$$

with $|i\rangle$ the i-th computational basis state on n_d qubits. In the following the normalization constant is neglected as normalized data is assumed.

Quantum states are either given in the computational basis $|i\rangle$ on n qubits, or as $|i_{n-1}\rangle...|i_0\rangle$, with i_{n-1} ...$i_1 i_0$ the binary representation of i. In the remainder, the chosen representation should be clear from the context. The left-most qubit $|i_{n-1}\rangle$ is the most significant qubit, and $|i_0 i$ the least significant qubit.

Classification comes down to evaluating the classification function of Equation (1). When doing so on a quantum computer, this can be done by preparing the following quantum state

$$|\mathcal{D}\rangle = \frac{1}{\sqrt{2M}} \sum_{m=0}^{M-1} |m\rangle \left(|0\rangle |\psi_{\tilde{\boldsymbol{x}}}\rangle + |1\rangle |\psi_{\boldsymbol{x}_m}\rangle\right) |y_m\rangle, \qquad (4)$$

where $|m\rangle$ is used to index training input m. The second register, consisting of a single ancillary qubit, is entangled with the third register containing the new data point, entangled with state $|0\rangle$, and training point m, entangled with state $|1\rangle$ of the second register. The fourth register contains the label of data point m, encoded via $y_m = -1 \; 7\rightarrow |0\rangle$ and $y_m = 1 \; 7\rightarrow |1\rangle$.

In total, this state requires $n_M + 1 + n_d + 1$ qubits, with $n_M = d\log_2 Me$ qubits used for the number of data points, a single ancilla qubit, n_d used for the features of a data point and a single qubit used for the label. Note that this state can be prepared explicitly, but could also be the result of a different quantum process.

Given this quantum state, the quantum algorithm consists of only three operations: First, a Hadarmard-gate on the second register, giving

$$\frac{1}{2\sqrt{M}} \sum_{m=1}^{M} |m\rangle \left(|0\rangle |\psi_{\tilde{\boldsymbol{x}}+\boldsymbol{x}_m}\rangle + |1\rangle |\psi_{\tilde{\boldsymbol{x}}-\boldsymbol{x}_m}\rangle\right) |y_m\rangle,$$

with $|\psi_{\tilde{x}\pm xm}\rangle = |\psi_{\tilde{x}}\rangle \pm |\psi_{xm}\rangle$; Secondly, a measurement on the second register, where execution is aborted if state $|1\rangle$ is measured. Otherwise, we obtain

$$\frac{1}{2\sqrt{Mp_{acc}}} \sum_{m=0}^{M-1} \sum_{i=0}^{N-1} |m\rangle \left(\tilde{x}^i + x_m^i\right) |i\rangle |y_m\rangle, \qquad (5)$$

where $p_{acc} = \frac{1}{4M}\sum_M |\tilde{\boldsymbol{x}} + \boldsymbol{x}_m|^2$ is a normalization factor and the probability that state $|0\rangle$ is measured. Hence, it is refered to as the 'acceptance probability'. Thirdly, a measurement on the fourth register.

The true probability that the result q of this last measurement is k ϵ {0,1}, is given by

$$\mathbb{P}[q = k] = \frac{1}{4Mp_{acc}} \sum_{m|y_m=k} |\tilde{\boldsymbol{x}} + \boldsymbol{x}_m|^2$$

and the label $\tilde{y}$ should be assigned based on

$$\tilde{y} = \arg\max_k \mathbb{P}[q = k].$$

Quantum computing is probabilistic, hence assigning a label based on a single evaluation of the algorithm might give incorrect results. Instead, the label should be assigned based on a majority vote over a number of evaluations. The number of shots required for a certain accuracy depends heavily on the data set. The process of measuring can be seen as sampling from a Bernoulli distribution. Let X be the statistical variable describing how often $|1\rangle$ was measured in R shots and let p be the true underlying probability. The probability that, after R shots, we assign the data point label −1 is given by the cumulative probability $\mathbb{P}(X \leq \frac{1}{2}R)$. For large R and p not close to zero or one, the binomial distribution can be approximated by a normal distribution. See Figure 1, the probability of a correct classification is shown for different number of shots R and underlying true probabilities p. The errors in classification can be shown to decrease with O(R) [12].

The complexity of this distance-based classifier depends on whether the initial state is given, for instance as a result from a different quantum process or loaded from a quantum RAM [16]. If we have to explicitly construct the state, each data point has to be encoded and the complexity is given by $O(Me^{nd}) = O(MN)$, see also Section 3.

If, however, the initial state is given, the algorithm consists of only a Hadamard gate and measurements on the second and fourth register, independent on the number of data points. Thus giving a complexity of O(1).

Note that the availability of the initial quantum state heavily determines whether this quantum distance-based classifier has an advantage over classical classification algorithms such as Support Vector Machines ($O(N^3M^3)$) and decision trees (O(MN logN)).

2.2 Implementing the distancebased classifier

In [12], the distance-based classifier algorithm is implemented on a 5-qubit quantum chip by IBM. Due to physical restrictions, only five qubits and at most eighty quantum gates could be used. Therefore, for the implementation both the number of features N and the number of data points M were fixed to 2. The extensions and modifications given in this paper are implemented on the Quantum Inspire simulator by QuTech [13]. This allows for simulations using more than five qubits, with no limitations on quantum circuit length and qubit connectivity. Furthermore, it supports full quantum-state evaluation and hence, exact probability distributions can be extracted from the Quantum Inspire platform.

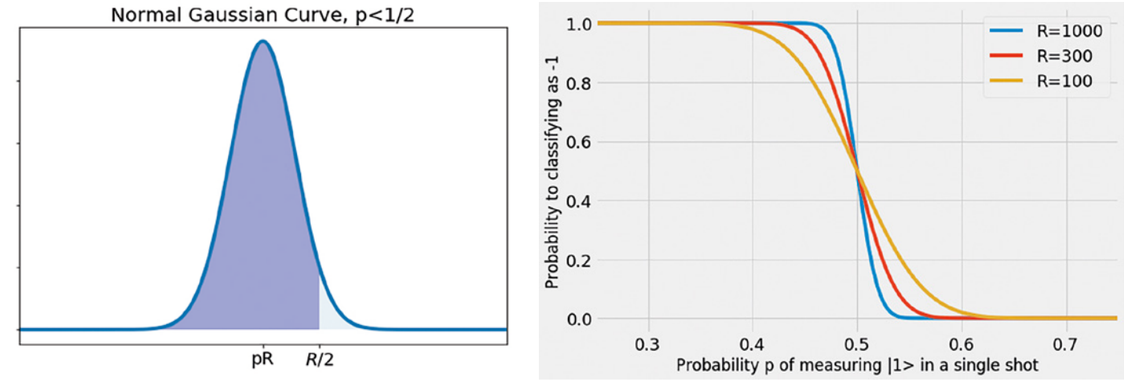


Figure 1: In the left figure the normal distribution is given for some p < ½. The shaded area corresponds to the probability of a correct prediction. The right figure shows the probability of classifying the data point with label –1 for different true probabilities p and different amount of shots R.

First, we implement the algorithm for N = 2 = M using four qubits. In the next sections, implementations of state-preparation will be given for each of the extensions. A normalized data point is encoded in a qubit via

$$R_Y(\theta_{\boldsymbol{x}})|0\rangle = |\psi_{\boldsymbol{x}}\rangle, \qquad (6)$$

with $|\psi$xi as in Equation (3). The state preparation of the algorithm can be split into different parts as shown in Figure 2. Firstly, the first two registers are brought in a uniform superposition using two Hadamard operations. Secondly, the new data point $\tilde{x}$ is encoded in the circuit using a controlled-RY rotation. Part C and D entangle both training data points x_0 and x_1 with the excited state in the second register and the corresponding index in the first. Finally, a CNOToperation is used to set the label in the fourth register for each data point.

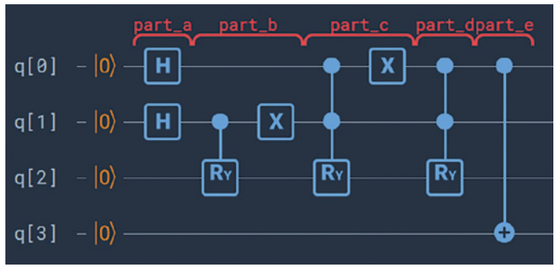


Figure 2: Circuit for the state preparation of the distance-based classifier.

A single Hadamard-operation on the second register and a measurement on the second and fourth register give the final result, these are not shown in the figure. Due to the probabilistic nature of quantum mechanics, this circuit should be evaluated multiple times to obtain the true label with high probability.

Quantum Inspire does not yet support direct implementations of the controlled-R_Y operations, hence it is decomposed in supported gates, as explained in [14].

The algorithm is tested by applying it on the Iris-data set [17]. This data set contains 150 data points, equally distributed among three classes and each data point having four features. For each evaluation, we randomly selected a data point from each of the first two classes as test point. Furthermore, we only considered the first two features. To encode the data points in qubits, we first standardized and then normalized the data set, thereby obtaining data points on the unit circle. The performance is analyzed based on how well a new, randomly chosen data point is classified. Figure 3 shows the resulting true probability distribution of the algorithm for two specific data points and a test point. Based on the given probabilities, the test point would be classified with label 1, which is indeed correct.

Naturally, one would expect that more data points or more features lead to better classifications. A classical analysis, shown in Table 1 indeed empirically proves this. Here, the classification function (1) is evaluated for different combinations of training points and test points. The training points are randomly sampled, as well as the test point. Either 1, 2 or 4 data points are drawn from each class. Furthermore, either the 2, 3 or 4 features are considered. This is done for 10,000 random samples and performance is given as percentage of correctly classified test points. As one would expect, using more training data or data with more features results in better predictions.

Table 1: Classical evaluation of the classification function (1) for different number of training points and features taken from the Iris data set. 10,000 random samples are used and the results are percentages of correctly predicted labels.

Number of data points	Number of features		
	2	3	4
2	94.26	98.70	99.40
4	97.35	99.33	99.86
8	98.03	99.75	99.98

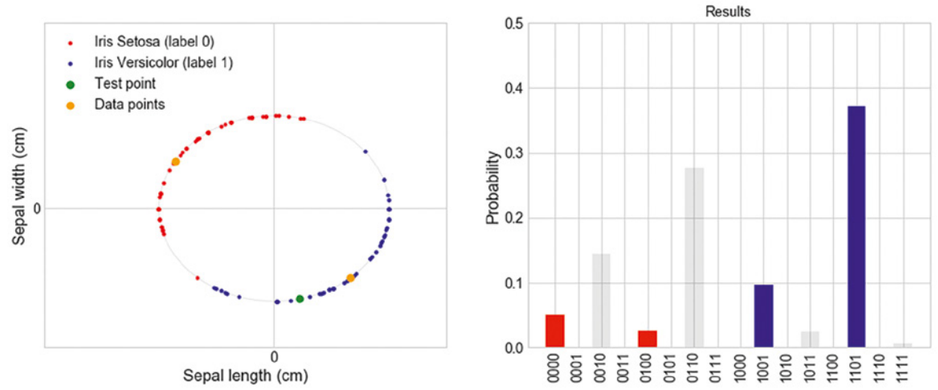


Figure 3: On the left: all normalized data points of the two classes of the iris data set, together with the selected data points (orange) and the test point (green). On the right: the probability distribution obtained with the algorithm. Red bars correspond to label 0, blue bars to label 1. Grey bars correspond to states with second register in |1⟩ and hence to the case we have to restart the algorithm.

3 Increasing the feature space

When the data allows for it, increasing the considered amount of features improves the quality of the classifier. For every qubit we add the amount of supported features are doubled. From Table 1 we see that, at least for the Iris data set, the performance increases more when doubling the considered features, than when we increase the total number of considered data points.

Let us consider the case of data points with $N \leq 2^{nd}$ features. The algorithm for classification stays the same, we only need to take into account that the data points are now encoded in n_d qubits. State preparation, if needed, becomes a little bit more involved as the data points are no longer encoded in a single qubit, but in n_d qubits. This requires a generalization of Equation (6) to be able to encode data in multiple qubits, i.e. an operation R_x is needed, such that $R_x |0...0\rangle = |\psi x\rangle$, with R being dependent on the data point.

In the literature, different implementations are presented [18, 19], the first implicitly using ancilla qubits, the second explicitly. Approximate methods also exist [20]. Our solution is inspired by the article of Long and Sun [18].

We show how to do this for N = 4 features. We refer to [14] for the analogous approach for more features and the precise implementation for N = 4 features. We want to construct some operation Rx such that

$$R_{\boldsymbol{x}} |00\rangle = a_0 |0\rangle + a_1 |1\rangle + a_2 |2\rangle + a_3 |3\rangle = |\psi_{\boldsymbol{x}}\rangle . \tag{7}$$

A possible way to construct R_x is by using three R_Y (α_i) rotations, two of which are controlled, see Figure 4, where the angles αi are given by

$$\alpha_1 = \arctan \sqrt{\frac{|a_2|^2 + |a_3|^2}{|a_0|^2 + |a_1|^2}},$$
$$\alpha_2 = \arctan \frac{a_1}{a_0}, \quad \alpha_3 = \arctan \frac{a_3}{a_2}. \tag{8}$$

Note, by setting $a_3 = 0$, we directly have a solution for N = 3 features.

4 Increasing the data space

In [12] an implementation is given for only two data points and a test point. Often however, interesting classification problems deal with data sets containing much more than two data points. One way of dealing with this is given in [21] and is based on sampling, using only two qubits, which is especially useful for NISQ devices. Here, only the second and third quantum register are encoded in qubits. In this approach, state preparation is less complex and less qubits are required. However, in general more quantum states need to be prepared and more circuit evaluations are needed. In Section 4.1 we will explicitly give the state preparation for the case that the number of data points is a power of two, while in Section 4.2 we will give it for the other case.

4.1 Number of data points a power of two

As Table 1 shows, using more data points increases the performance of the distance-based classifier.

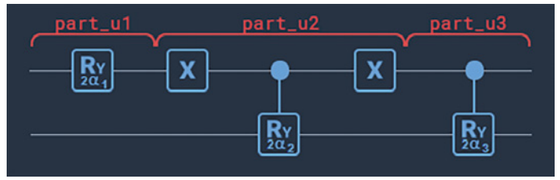


Figure 4: A quantum circuit to encode a normalized data point with N = 3 or N = 4 features in n = 2 qubits. Angles αi depend on the data point.

While the workings of the distance-based classifier itself does not change when more data points are used, state preparation becomes more complex.

To encode M data points, we need n_M qubits in the index register. For two features per data point, this gives a total of n_M +3 required qubits. In the remainder, the subscript M will be dropped.

We use Equation (6) to encode all data points in the initial state. For the test point, we use a controlled operation, with the second register as control. For the data points, again a controlled rotation is used, however now the first and the second register are used as control. In the following only the first and third register are considered.

Hence, state $\frac{1}{\sqrt{M}} \sum_{m=0}^{M-1} |m\rangle |0\rangle$, must be mapped to $\frac{1}{\sqrt{M}} \sum_{m=0}^{M-1} |m\rangle |\psi_{\boldsymbol{x}_m}\rangle$, using unitary operations, with $|\psi_{\boldsymbol{x}_m}\rangle$ as in Equation (3).

Define an n qubit multi-controlled rotation by

$$C^n \mathrm{R_Y}(\theta_{\boldsymbol{x}}) |m\rangle |0\rangle = C^n \mathrm{R_Y}(\theta_{\boldsymbol{x}}) |m_{n-1} \dots m_0\rangle |0\rangle$$
$$= |m_{n-1}\rangle \dots |m_0\rangle \, \mathrm{R_Y}^{m_{n-1} \cdot \dots \cdot m_0}(\theta_{\boldsymbol{x}}) |0\rangle \tag{9}$$
$$= \begin{cases} |m\rangle |\psi_{\boldsymbol{x}}\rangle & \text{if } m_i = 1 \, \forall i \\ |m\rangle |0\rangle & \text{otherwise} \end{cases} \tag{10}$$

where $m_n \dots m_0$ is the binary representation of m. Using adjoining X-operations around control qubit i, we can flip a control, i.e., the term m_i in Equation (9) gets replaced by $(1 - m_i)$. Note, to minimize the number of X-operations, data points should be ordered according to the Gray code [22]. Decompositions for controlled- and doublecontrolled-R_Y-operation are given in [14]. Multicontrolled-R_Y-operations can be constructed using a controlled-R_Y operations, ancilla qubits and Toffoli-operations.

4.2 Number of data points not a power of two

Often, the number of data points is not a power of two, which complicates the preparation of the first register, whereas, if the number of data points is a power of two, a single Hadamard operation on each qubit is sufficient. Again, techniques proposed in [18, 19, 20] can be used to initialize the first register to a uniform superposition over the first M states, however, each has drawbacks that nullify the benefits of using a quantum algorithm.

Instead, it is also possible to construct a uniform superposition over all 2^n states, and classically post-process the results to filter out measurement results not corresponding to the first M states. Implementation-wise this is easier, while it gives at most a factor 2 overhead in the number of circuit evaluations. For M data points, states |i⟩ in the first register with $i \geq M$ are not relevant for the algorithm, and hence affect the absolute probabilities of other states. However, the relative probabilities for states |j⟩, j = 0,...,M − 1 are not affected. Therefore, in a post-processing step, all measurements corresponding to states |i⟩ for $i \geq M$ can be discarded, thereby giving the same probability distribution as when an initial state was prepared only for the state |0⟩,...,|M − 1⟩. As $2^{n-1} < M < 2^n$, there are at most $2^{n-1} - 1$ 'extra' states in the first register, that are filtered in a post-processing step. Note, if $M < 2^{n-1}$, we have unneeded qubits in the first register.

5 More classes

Up to now we only considered a binary classification problem. However, many data sets have more than two possible labels, consider the Iris-data set for instance, already having three different classes. Minor modifications to the algorithm allow for extensions to more than two classes. Let C be the number of classes, the fourth register must then have $n_c = [\mathrm{dlog}_2 C]$ qubits.

First, in Section 5.1 we consider balanced data sets, i.e., data sets where each class has the same number of data points all equally likely, afterwards in Section 5.2 we consider unbalanced data sets.

5.1 Balanced classes

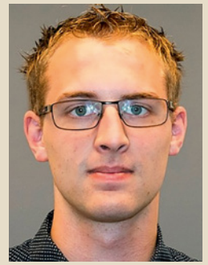
For normalized data points, the probability for each class i is given by

$$\mathbb{P}(\tilde{y} = i) = \frac{1}{4Mp_{acc}} \sum_{m|y_m=i} |\tilde{\boldsymbol{x}} + \boldsymbol{x}^m|^2, \tag{11}$$

with p_{acc} an acceptance probability and normalization constant. Note, Equation (11) is not limited to only two classes, but also holds for an arbitrary number of classes. For more classes, only the nc qubits in the fourth register need to be set to a state depending on the corresponding class. For balanced classes, this can be done using CNOToperations, with control in the first register.

The required number of gates can be optimized by a smart assignment of indices to data points. Suppose we have four index states |00⟩, |01⟩, |10⟩ and |11⟩ and two classes. Assigning |10⟩ and |11⟩ to class 1, requires only a single CNOT-operation,

Robert Wezeman
Robert Wezeman is a scientist at TNO. He works on near term applications of quantum computers. He studied mathematics and physics.



Niels Neumann
Niels Neumann is a scientist at TNO. He works on applications of quantum computers and -networks. He studied mathematics and physics.

Frank Phillipson
Frank Phillipson is senior scientist at TNO. He leads the project team within TNO that studies applications and algorithms for near term use on quantum computers and quantum simulators. He studied econometrics and mathematics and has a PhD in applied mathematics.

with the most significant qubit as control.

An efficient implementation for eight data points uses only two CNOT-operations with target in the first and control in the fourth register:

- Class 0: States $|000\rangle$ and $|001\rangle$ set by initialization;
- Class 1: States $|010\rangle$ and $|011\rangle$ set by CNOT with target on second qubit, control on least significant qubit;
- Class 2: States $|100\rangle$ and $|101\rangle$ set by CNOT with target on most significant qubit, control on most significant qubit;
- Class 3: States $|110\rangle$ and $|111\rangle$ set by the two CNOT-gates used for Label 1 and 2.

5.2 Unbalanced classes

In real data sets, classes are often not balanced. The algorithm however assumes both classes having the same number of data points. This assumption can be relaxed by assigning weights to data points.

A test point with distance d_0 to class zero with weight w_0 and distance d_1 to class one with weight w_1, should be given the first label if and only if $w_0 d_1 > w_1 d_0$. This can also be used to solve the problem of classes with different number of data points. Suppose we have a class with t data points, adding weights 1/(tC) to each data point, gives the class a total weight of 1/C. Doing so for each class creates equally likely classes. Weights can be introduced in the initial superposition $\sum_i w_i |i\rangle$, such that $\sum_i w_i^2 = 1$. The corresponding weighted version of Equation (11) is given by

$$\mathbb{P}(\tilde{y} = i) = \sum_{m|y_m=i} w_i^2 |\tilde{\boldsymbol{x}} + \boldsymbol{x}^m|^2. \qquad (12)$$

Creating the superposition $\sum_i \sqrt{w_i}\,|i\rangle$ in the first register for arbitrary wi is, however, exponentially hard. Instead, the algorithm can also be run as if each data point is equally likely, to obtain 'raw' probabilities for each data point, which are then multiplied by the weights in a classical postprocessing step. This gives the same result as encoding the weights in the superposition. For an unweighted superposition, the true probabilities of the data points are given by $\frac{1}{4Mp_{acc}}|\tilde{\boldsymbol{x}} + \boldsymbol{x}^m|^2$, which are precisely the unweighted arguments of Equation (12). Classically multiplying these probabilities with the weights is more efficient than creating a weighted initial superposition.

6 Dealing with concentric Data

Up to now we did not talk about assumptions on the data set, however, if classes are not separable based on angles, such as concentric circles, the proposed classifier does not work, as normalizing those data points gives two coinciding circles of radius one. With feature maps as proposed in [12], it is often not clear whether a data set consists of concentric circles or which features should be considered in a feature map. Furthermore, not all data sets are suitable for feature maps, as the number of available features is insufficient, or as other feature combinations also give concentric circles.

We propose a different approach, based on a combination of the norm of the data point before normalization and the angular features already used. Let p_x be the distance of a data point x to the origin and $p_{max} = \max_x p_x$ the maximum distance among all data points. Now map p_x to $(l_1,l_2) = (\cos(p_x\pi/p_{max})$,$\sin(p_x\pi/p_{max}))$ and use this result as an additional feature in the distance-based classifier. Note that this mapping gives a normalized result. Now construct the state

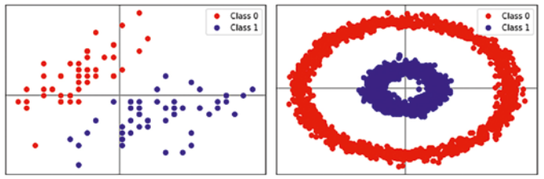
$$\sqrt{1-\omega}(\alpha_0 |0\rangle + \ldots + \alpha_N |N-1\rangle) + \sqrt{\omega}(\ell_1 |N\rangle + \ell_2 |N+1\rangle),$$

using n_d qubits and αi the amplitudes of the features. Here $n_d = [\log_2(N+2)]$ qubits are used and ω is a weighting between already used features and the new norm-based feature.

One would expect that this additional feature helps in better classifying data points. We again classically analyzed the classification function of Equation (1) with the extra norm based-feature for two different data sets, the Iris-data set and a data set consisting of concentric circles. Results are obtained by randomly sampling data points from the data sets and evaluating (1) 10,000 times. The performance is given in Table 2 in percentage of correctly classified points for different weight factors ω. Here for ω = 0 only the original features are considered, ω = 1 only the norm-features and for other ω-values, a weighted average of the two is considered. Visual representations of the data sets are shown in Figure 5.

Table 2: Classical evaluation of (1) for different number of training points and different weighting factors ω, shown in the columns. 10,000 random samples for both data sets are used and the results are the percentage of correctly predicted samples.

# of points	Iris data set			Concentric data		
	0	1	0.5	0	1	0.5
2	94.5	49.9	91.5	49.2	100	91.1
4	96.9	50.8	95.4	50.1	100	98.3
8	98.2	51.2	97.6	50.2	100	99.5



(a) The first two features (b) Example of concentric of the Iris-data set after data standardization.
Figure 5: The two example data sets considered

The table shows that on average ω = 0.5 performs best in terms of percentage of correctly predicted samples, even though it does not necessarily perform best on individual data sets. Moreover, in all cases it performs significantly better than random guessing, contrary to ω = 0 and ω = 1. In general, a combination of both angular data features and norm based features improves performance over using only one of the two. The best ω-value should be decided by means of optimization over different data sets.

7 Conclusion

Previous work on a quantum distance-based classifier is limited to data sets with binary labels and balanced, non-concentric data sets, with the number of data points a power of two. In this work we show how to modify the classifier to overcome these restrictions. We show how to extend the classifier to allow for classification between more than two classes. By applying different weighting factors to different classes we show that classification can still be done for unbalanced data sets. Furthermore, we extend previous work by giving the quantum circuits needed for the state preparation both for more data points and more features.

Another restriction of the classifier is that it only works for angular separable data. Other data sets could be dealt with using feature maps, however, this also has its limitations. Instead, we propose a different and more generic solution where we encode both the angular dependence as well as the radial dependence. Finding a good balance between the two should be determined by means of optimization over multiple data sets.

Although the demonstrated classifier is a small toy example and, due to the included state preparation, far from delivering a quantum advantage above classical computation, it clearly demonstrates the potential of quantum computing. If state preparation is not needed, for instance because the state is obtained from quantum RAM or a different quantum process, the algorithm shows significant speed-up, as a fixed number of only three operations is needed.

References: **[1]** C. M. Bishop. Pattern recognition and machine learning. Information science and statistics. Springer, 2006. **[2]** Y. Sun, B. Xue, M. Zhang, and G. G. Yen. Evolving Deep Convolutional Neural Networks for Image Classification. arXiv:1710.10741 [cs], Oct 2017. arXiv: 1710.10741. **[3]** A. L. Buczak and E. Guven. A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection. IEEE Communications Surveys Tutorials, 18(2):1153–1176, 2016. **[4]** D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot,L. Sifre, D. Kumaran, T. Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. Science, 362(6419):1140–1144, 2018. **[5]** D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, and D. Hassabis. Mastering the game of Go without human knowledge. Nature, 550, 2017. **[6]** S. Yoo, J. Bang, C. Lee, and J. Lee. A quantum speedup in machine learning: Finding a N-bit Boolean function for a classification. New Journal of Physics, 16(10), Oct 2014. arXiv: 1303.6055. **[7]** V. Dunjko, Y. K. Liu, X. Wu, and J. M. Taylor. Exponential improvements for quantum-accessible reinforcement learning. arXiv:1710.11160 [quant-ph], Oct 2017. arXiv: 1710.11160. **[8]** T. J. Lopez-Chagoya. Hybrid helmholtz machine: A gate-based quantum circuit implementation. Master's thesis, 2018. **[9]** N. M. P. Neumann, F. Phillipson, and R. Versluis. Machine learning in the quantum era. Digitale Welt, 2019. **[10]** M. Schuld, I. Sinayskiy, and F. Petruccione. An introduction to quantum machine learning. Contemporary Physics, 56(2):172–185, Apr 2015. arXiv: 1409.3097. **[11]** J. Preskill. Quantum Computing in the NISQ era and beyond. Quantum, 2:79, Aug 2018. arXiv: 1801.00862. **[12]** M. Schuld, M. Fingerhuth, and F. Petruccione. Implementing a distance-based classifier with a quantum interference circuit. EPL (Europhysics Letters), 119:60002, 09 2017. **[13]** QuTech. Quantum Inspire. https://www. quantum-inspire. com/. accessed: 2019-0630. **[14]** QuTech Quantum Inspire. A quantum distance-based classifier. https://github. com/QuTech-Delft/quantuminspire/tree/ dev/docs/classifier_example, 2019. **[15]** T. Hofmann, B. Schölkopf, and A. J. Smola. Kernel methods in machine learning. The Annals of Statistics, 36(3):1171–1220, Jun 2008. arXiv: math/0701907. **[16]** V. Giovannetti, S. Lloyd, and L. Maccone. Quantum random access memory. Physical Review Letters, 100(16):160501, Apr 2008. arXiv: 0708.1879. **[17]** R. A. Fisher. The use of multiple measurements in taxonomic problems. Annual Eugenics, 7:179–188, 1936. **[18]** G. L. Long and Y. Sun. Efficient Scheme for Initializing a Quantum Register with an Arbitrary Superposed State. Physical Review A, 64(1):014303, Jun 2001. arXiv: quantph/0104030. **[19]** D. Ventura and T. Martinez. Initializing the Amplitude Distribution of a Quantum State. arXiv:quant-ph/9807054, Jul 1998. arXiv: quant-ph/9807054. **[20]** A. N. Soklakov and R. Schack. Efficient state preparation for a register of quantum bits. arXiv:quant-ph/0408045, Aug 2004. arXiv: quant-ph/0408045. **[21]** P. Sadowski. Quantum distance-based classifier with distributed knowledge and state recycling. International Journal of Quantum Information, 16(8):1840013–687, Jan 2018. **[22]** F. Gray. Pulse code communication.

Nash embedding: a road map to realizing quantum hardware

Faisal Shah Khan

The non-Euclidean nature of the state-space of qubits (and qudits in general) gives rise to the problem of practically implementing quantum circuits in physical hardware which necessarily resides in the Euclidean space R^3. On the other hand, the Euclidean nature of bits (and dits in general) makes the implementation of reversible circuits in physical hardware relatively straight forward. I offer here a road-map to solving this problem in which the Nash embedding theorem isometrically maps qubits into bits and a quantum circuit into an equivalent reversible one, followed by the embedding of the resulting reversible circuit into R^3 as a hardware graph.

1 Introduction

Prototype quantum computing systems are currently available from several vendors, including DWave Systems and Rigetti Computing (and IBM, for that matter), yet there is still much controversy about whether these qualify as "quantum" computers and how accurately they presage eventual quantum computers that will deliver quantum supremacy [1]. A term that is used to describe these prototypes is Noisy IntermediateScale Quantum (NISQ) hardware. Instead of manipulating single qubits, NISQ devices manipulate the flow of a large collection of qubits, cooled down to near absolute-zero, so that quantum and classical noise is suppressed and quantum effects like superposition manifest within the ensemble of qubits. This is in contrast to the theoretically more robust notion of quantum hardware which allows the manipulation of individual qubits to produce superposition and entanglement. Noise, in particular classical noise coming in from the environment, is again a fundamental challenge since programming a quantum processor necessarily requires interaction with a programmer residing in the environment. Some levels of noise can be tolerated if the hardware is built with error-detection and correction codes [2], an idea motivated by how noise induced errors are dealt with in classical hardware. However, implementing these codes in quantum hardware turns out to be an expensive endeavor [3]. Designing and realizing quantum hardware continues to be an active area of research both in academic and industrial settings as its successful implementation (with respect to noise suppression, or ideally, cancellation) is understood to be the beginning of quantum supremacy over classical devices.

I propose here a mathematically formal approach as a step toward achieving quantum supremacy. Note (or recall) that the physics of qubits takes place in a complex projective space and then note that this space is also a compact Riemannian manifold. The physics of classical objects takes place in the Euclidean space, the stereotypical manifold of our every day experience. By John Nash's embedding theorem, we know that compact Riemannian manifolds can be embedded inside some high dimensional Euclidean space in a way that preserves length. We regard the embedded image of the compact manifold of qubits inside Euclidean space as representing exotic sectors in the Euclidean space that can be made to exhibit quantum properties by tracing the embedding back to its quantum origin.

Nash embedding has the following appealing properties: as a submanifold of the Euclidean space, its image "realizes" qubits. Being an isometry, that is, preserving length, it preserves spatial relationships between qubits when they are realized. This means that the action of *any* quantum logic gate (unitary matrix) on the qubits can be faithfully implemented by some reversible logic gate (orthogonal matrix) acting on the image of the qubits, albeit in a Euclidean space with dimension possibly higher than the original space of qubits. This property is the main focus here.

A noteworthy point here before proceeding to the details is that the classical environment is more properly pseudo-Riemannian than Euclidean, once relativistic considerations are taken into account. Pseudo-Riemannian (also known as semi-Riemannian) manifolds embed isometrically into the pseudo-Euclidean space (such as the Minkowski space) [4],[5], but the question of whether they isometrically embed into Riemannian manifolds appears to be unsettled. This is one explanation for why the reconciliation of quantum physics and relativity remains elusive. This issue will certainly come to the forefront when relativistic quantum circuits become ubiquitous, at which time new insights into reconciling the two forms of physics might become apparent.

2 Quantum state space and Nash embedding

In a quantum computing context, the relevant compact Riemannian manifold is the twodimensional complex projective space of a single qubit, CP^1. This set consists of equivalence classes of "complex lines", that is, all the non-zero vectors $v_1, v_2 \in C^2$ declared equivalent if $v_1 = \lambda v_2$ for non-zero complex numbers λ. We can visualize this space by shrinking all the complex lines down to unit length so as to generate the unit sphere S^3 in $R^4 (= C^2)$, and then identifying the antipodal points on S^3. This means that

$$\mathbb{C}P^1 = \mathbb{S}^3/\mathbb{S}^1 = \mathbb{S}^2 \subset \mathbb{R}^3. \qquad (1)$$

In quantum computing literature, S^2 is utilized as the Bloch sphere, a *representation* of a qubit to assist with calculations.

Next, consider the Cartesian product $CP^1 \times CP^1$ as the model for joint system of two qubits. This set can be given a four-dimensional Hilbert space structure using the direct-sum construction. However, the direct-sum is incompatible with projectivity. Instead, the tensor product can be utilized, which while not well-defined for projective spaces, is well-defined for Hilbert spaces. The model for the joint space of two qubits is therefore the projected four-dimensional Hilbert space $C^2 \otimes C^2 = C^4$, denoted as CP^3. Using the Segre embedding [6], a copy of $CP^1 \times CP^1$ can be found inside CP^3 as a submanifold. Hence, the image of the Segre embedding describes two qubit product states and the remaining CP^3 describes two qubit entangled states. The joint complex projective space of $n > 2$ qubits is produced by iterating this construction to get

$$\mathbb{C}P^{2^n-1} = \mathbb{S}^{(2^{n+1}-1)}/\mathbb{S}^1 \qquad (2)$$

where $S^{(2_{n+1}-1)}$ is the unit sphere in $R^{2n} = C^n$. A copy of the set

$$\mathbb{C}P^1 \times \mathbb{C}P^1 \cdots \times \mathbb{C}P^1$$

resides as a submanifold of CP^{2_n-1} and contains the product states of n qubits. This submanifold is sometimes written as

$$\otimes_{i=1}^{n} (\mathbb{C}P^1)_i \qquad (3)$$

which is an abuse of notation obviously but has the advantage of being a clear reference to the product.

In [7], Nobel Laureate John Nash established the following result:

Nash embedding theorem: *For every compact Riemannian manifold M, there exists an isometric embedding of M into* R^m *for a suitably large m.*

An embedding is a differentiable *homeomorphism*, that is, a bi-continuous one-to-one and onto function from the manifold onto a submanifold of R^m. According to Gunther [8],

$$m = \max\left(\frac{k(k+5)}{2}, \frac{k(k+3)}{2} + 5\right) \qquad (4)$$

where k is the dimension of the M. Nash's result has been developed further over the years and a more up to date exposition can be found in [9]. In the next section, Nash embedding is applied to the n qubit register CP^{2_n-1}.

2.1 Quantum logic gates as faithful reversible ones

In [10], Bennett showed that it is possible to make any logically irreversible circuit (or its corresponding Turing machine), logically reversible; in other words, the function computed by the circuit can be made invertible. This further implied physical or thermodynamic reversibility, meaning that in principle, digital computers can be built that dissipate an arbitrary small amount of heat. Bennett's (and related) works motivated further studies in reversible computing, leading Ingarden to formulate the (Shannon) theory of quantum information [11] and leading Feynman to propose the construction of quantum computers [12] to simulate complicated quantum systems with simpler one's. As I show here, the ability of Nash embedding to realize a quantum circuit with a reversible one, and vice versa, elegantly completes the one-to-one relationship between the processing of quantum and classical information.

To initialize the n qubit register CP^{2_n-1}, a unitary transformation, and therefore an isometry with respect to the norm induced by the FubiniStudy metric, is enacted on it as a quantum logic gate

$$Q : \mathbb{C}P^{2^n-1} \longrightarrow \mathbb{C}P^{2^n-1} \qquad (5)$$

to configure its state to the desired one. Under a Nash embedding, the n qubit register maps to a submanifold S of a Euclidean space R^m,

$$e : \mathbb{C}P^{2^n-1} \hookrightarrow S, \qquad (6)$$

with $k = 2^n$ and $m = 2^{2n-1} + 3 \cdot 2^{n-1} + 5$ for $n = 1,2$ and $m = 2^{2n-1} + 5 \cdot 2^{n-1}$ otherwise. Because e is also an isometry, Q can *always* be enacted with respect to the realized register S via a length-preserving (with respect to the Euclidean norm) orthogonal transformation

$$R : S \longrightarrow S.$$

That is, for $p \in CP^{2_n-1}$

$$R(e(p)) = e(Q(p)) \qquad (7)$$

as depicted in the commutative diagram of Figure 1 and from which it follows that

$$Q = e^{-1}(R(e(p))). \qquad (8)$$

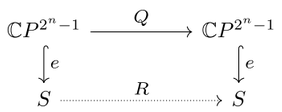


Figure 1: Quantum logic gate Q transforming a quantum state to another. If e is a Nash embedding into $S \subset R^k$ for a suitable k, then the isometry of both means that there always exists an orthogonal transformation R from S to itself that realizes Q.

For the simplest instance of Nash embedding, $CP^1 \to R^{10}$ and a 2×2 unitary matrix $Q \in U(2)$ is realized by a 10×10 orthogonal matrix $R \in O(10)$. A two qubit register will Nash embed into R^{19}! Even more dramatically, $CP^7 \to R^{52}$ and $CP^{15} \to R^{168}$. Nash embedding clearly requires a large amount of Euclidean space by virtue of being an isometry; compare with the non-isometric Whitney embedding [13] for which $CP^1 \to R^5$.

For an $r \times r$ unitary with $2r^2$ real parameters and an $s \times s$ orthogonal matrix with s^2 real parameters, the orthogonal matrix will be less costly, with respect to number of real parameters that define it, as long as $s^2 < 2r^2$. But for an $n \geq 3$ qubit register, $r = 2^n$ and $s = \left(2^{2n-1} + 5 \cdot 2^{n-1} \times 2^{2n-1} + 5 \cdot 2^{n-1}\right)$, and for these values $s^2 > 2r^2$. Hence, for any computationally meaningful number of qubits, the reversible gate will always be costlier than the unitary gate.

3 Noise and quantum hardware implementation

Since programming the qubit register via Q is equivalent under the Nash embedding to programming its real image in Euclidean space via R, classical noise arising from the programming effort can now be modeled and studied as having affected a real register. This real register may further be reduced to a binary register and classical errordetecting and correcting codes can be built into R. Thus, the well established theory and applications of classical error detecting and correcting codes will suffice for developing a robust understanding of error-free quantum computing.

The main focus here however is the physical implementation of quantum hardware in R^3. Reversible gates and circuits, when represented as orthogonal matrices, define hardware graphs in Euclidean space and can therefore be embedded into R^3 using graph embedding techniques that have been utilized in the past for implementing highdimensional hardware architectures such as VLSI. More precisely, once Nash embedding has faithfully realized Q as the orthogonal matrix R, this matrix can be represented as a weighted adjacency matrix for a graph [14] G_R in R^m. The graph G_R represents the hardware architecture that implements R in R^m. Finally, G_R can in turn be embedded inside R^3 since *any* graph can be embedded in this space [15]. To further manage this embedding of the hardware graph G_R into R^3, techniques like *book embeddings* can be utilized where the graph in R^3 is laid out as stack of sheets (planes) which connects together along a common back-bone of the "book" [16]. The theoretical plan of action would be

$$Q \longrightarrow R \longrightarrow (G_R \subset \mathbb{R}^m) \longrightarrow \mathbb{R}^3 \longrightarrow (\text{Book Graph} \subset \text{stacked } \mathbb{R}^2)$$

before fabrication of quantum hardware can begin.

4 Conclusion

Nash's embedding result is non-algorithmic in nature. In the context of imaging and learning algorithms, efforts in trying to algorithmize Nash's embedding have been made, for instance, in [17], [18], and [19]. However, similar efforts in algorithmic implementations that take into account features of quantum information data, such as quantum measurement, appear not to have been explored. While a direct, forward construction of a Nash embedding from CP^{2^n-1} to R^m requires topological and differential geometric study, it may also be possible to use quantum state tomography and purification techniques to produce an inverse construction. Quantum state tomography and parameter estimation techniques attempt to recreate a quantum state from the date collected from repeated measurements (sometimes in different basis) of several copies of the state [20], [21], [22].

Formally, this is the function

$$t : \Delta_l \longrightarrow \Delta(\mathbb{C}P^{2^n-1}) \longrightarrow \mathbb{C}P^{2^n-1}$$

where t goes from the simplex of probability distributions in R^l to the space of density matrices, and then, after purification, to the state space of pure quantum states. For instance, t can be the linear inversion function which uses a conditional probabilistic version of Born's rule to estimate a pure quantum state from measurement data. Tomographical methods may not immediately produce an inverse isometric embedding and will likely require mathematical tuning, but they will likely serve as good first approximations.

References **[1]** J. Preskill, Quantum computing and the entanglement frontier, Rapporteur talk at the 25th Solvay Conference on Physics ("The Theory of the Quantum World"), 19-22 October 2011. **[2]** S. Devitt, W. J. Munro, K. Nemoto, Quantum error correction for beginners, Reports on Progress in Physics, Volume 76, Number 7, 2013. **[3]** J. Preskill, Quantum computing in the NISQ era and beyond, Quantum 2, 79 (2018). **[4]** , C.J.S. Clarke, On the global isometric embedding of pseudo-Riemannian manifolds **[5]** , R. Greene, Isometric embeddings of Riemannian and psuedo-Riemannian manifolds, American Mathematical Society, 1970. **[6]** I. Bengtsson, K. Zyczkowski, Geometry of quantum states: an introduction to quantum entanglement, Cambridge University Press; 1 edition , anuary 14, 2007. **[7]** J. Nash, The imbedding problem for Riemannian manifolds, Annals of Mathematics, 63 (1): 2063, 1956. **[8]** M. Gunther, Isometric embeddings of Riemannian manifolds, Proceedings of the International Congress of Mathematicians, Vol. I, II (Kyoto, 1990), 11371143, Math. Soc. Japan, Tokyo, 1991. **[9]** Q. Han and J. Hong, Isometric embedding of Riemannian manifolds in Euclidean spaces, Mathematical Surveys and Monographs, Volume: 130, American Mathematical Society (October 15, 2006). **[10]** C. Bennett, Logical Reversibility of Computation, IBM Journal of Research and Development, vol. 17, no. 6, pp. 525-532, 1973. **[11]** R. Ingarden, Quantum information theory, Reports on Mathematical Physics, vol. 10, no. 1, pp. 4372, 1976. **[12]** R. Feynman, Simulating physics with computers, International Journal of Theoretical Physics, vol. 21, pp. 467488, 1982. **[13]** H. Whitney, The collected papers of Hassler Whitney, Birkhauser, Boston 1992. **[14]** N. Biggs, Algebraic graph theory, Cambridge University Press; 2 edition (February 25, 1994). **[15]** Cohen, R.F., Eades, P., Lin, T. et al., Three-dimensional graph drawing, Algorithmica (1997) 17: 199. https://doi.org/10.1007/BF02522826 **[16]** Fan R. K. Chung, Frank Thomson Leighton, and Arnold L. Rosenberg, Embedding graphs in books: A layout problem with applications to VLSI design, SIAM. J. on Algebraic and Discrete Methods, 8(1), 3358. (26 pages). **[17]** J. McQueen, M. Melia, D. Joncas, Nearly Isometric Embedding by Relaxation, Proceedings of Neural Information Processing Systems 2016, https://papers.nips.cc/paper/6535nearly-isometric-embedding-by-relaxation. **[18]** N. Verma, Towards an algorithmic realization of Nash's embedding theorem, available at https://pdfs.semanticscholar.org/ c79b/67fa46b80d900c62cd7f09278687cf20642d .pdf. **[19]** Z. Zhong et al., Computing a highdimensional euclidean embedding from an arbitrary smooth riemannian metric, ACM Transactions on Graphics, volume 37 issue 4, 2018. **[20]** J. Altepeter, D. James and P. Kwiat, Quantum state tomography, available at http://research.physics.illinois.edu/QI/ Photonics/ tomographyfiles/tomochapter2004.pdf **[21]** M. Paris, Quantum estimation for quantum technology, Int. J. Quant. Inf. 7, 125 (2009). **[22]** S. Olivares and M. G. A. Paris, Bayesian estimation in homodyne interferometry, J. Phys. B: At. Mol. Opt. Phys. 42, 055506 (2009).

Faisal Shah Khan
Faisal Shah Khan is an Assistant Professor of Mathematics and a Principal Investigator with the Center on Cyber-Physical Systems at Khalifa University, Abu Dhabi. He also serves as an advisor to the quantum-inspired technology company Quantum Computing, Inc. based in Virginia, USA.

Implementation of a Variational Quantum Circuit for **Machine Learning with Compact Data Representation**

Nicholas Meinhardt, Bastiaan Dekker, Niels M.P. Neumann, Frank Phillipson

Machine learning is already used to solve numerous problems and is growing in importance every day. Quantum computers may be able to improve the performance of machine learning algorithms by exploiting the power of quantum mechanics. In this work, we present and implement a variational quantum circuit, which is able to classify small artificially generated data sets with binary labels. Moreover, we consider a more efficient representation of data, which is important for implementations on real near-term quantum devices. The implementation is done using simulations, and can easily be applied to real quantum devices with only minor adaptions.

1 Introduction

With quantum mechanics limiting further minimization of transistors at some point, Moore's law is facing its end [1]. However, the same rules of quantum mechanics offer a new path for information processing, where information is stored in quantum states and computations can be performed in parallel [2]. Although several algorithms are considered to outperform classical computing, such as Shor's factoring algorithm [3], large and fault-tolerant quantum computers will probably not be realized within near future due to technological challenges. Instead, small quantum computers with few and imperfect qubits will dominate on near-term, so called noisy intermediate-scale quantum (NISQ) devices [4], some of which are already commercially available. One of the promising candidates to show a quantum advantage on such devices is believed to be quantum machine learning (QML) [5, 6]. Machine learning (ML) has become an important tool to process data and extract information from them in a great variety of applications. Facing a steadily increasing amount of data and limiting processor capacity, developing more efficient algorithms seems crucial and using quantum computers seems to be a promising approach.

In this work we build on the variational quantum circuit proposed by Farhi and Neven [7], that has a similar structure as a feed-forward neural network. We present the explicit circuit representation and apply the circuit to the canonical bars-and-stripes dataset. The circuits are simulated on a cloudbased quantum simulator. Next, we propose a more compact presentation of input data than in the original proposal. The limited availability of qubits will, at least in the near future, ask for an efficient way to store input data. The remainder of this work is as follows. In Sec. 2 we give an introduction to classical machine learning to introduce the concept and some terminology. In Sec. 3 our contribution will be elaborated. Next, results of the learning and the dense representation are presented in Sec. 4 and discussed in Sec. 5. We conclude with Sec. 6.

2 Machine learning

ML algorithms are traditionally divided into three classes: supervised, unsupervised and reinforcement learning. We will focus on supervised learning, where a trainingset containing N feature vectors x_p and their correct classifications c_p is presented to the algorithm. The algorithm then optimizes the overlap of its outputs y_p with the desired c_p by varying

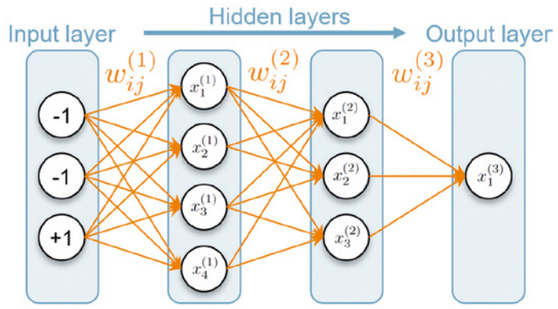


Figure 1: A schematic overview of a layered feedforward neural network with three input neurons, two hidden layers and a single output neuron.

internal parameters. After training, new inputs are classified without further optimization. This type of learning is mainly used for pattern recognition.

Inspired by neurons in nature, artificial neural networks (ANN) are popular models in machine learning. Such networks consist of nodes, called neurons, and edges connecting the neurons. To create a network, neurons and edges are connected in various configurations. ANNs are often considered as layered, which means that only nodes of adjacent layers are connected, whereas there are no synapses within one layer. The input and output neurons can be directly observed from the outside, therefore they are regarded as visible. Neurons in intermediate layers are referred to as hidden and do not interact with the outside. An example of a layered ANN with three input neurons, two hidden layers of four and three neurons and a single output neuron is shown in Fig. 1. The state of the *i*-th neuron in the *k*-th layer is described by $x_i^{(k)}$. The edges connecting layers $k-1$ and k are associated with weights 1], corresponding to the synaptic strength of two nodes. Here we only consider feed-forward ANNs with directed weights, which are only nonzero in forward direction, such that signals are heading in one direction only. The weights between layers *kk* can be summarized in a weighting matrix . The evolution of a network is determined by the input states and its weights. For the activation of a single neuron, the pre-synaptic states of all connected nodes are added up by a weighted sum. An activation function *f* is then applied to this sum to determine the state of $x_i^{(k)}$.

To train an ANN in a supervised fashion on a trainingset, its weights are optimized to minimize a loss function *loss*(*y*,*c*) averaged on all samples. The loss is a measure of the distance of the predicted result y_p and the actual label c_p, i.e., a measure of the performance of the ANN. The goal is to find a function $f_{\theta\min}$ parametrized by parameters θ that minimizes the empirical risk

$$E_N(f_{\boldsymbol{\theta}}) = \frac{1}{N}\sum_{p=0}^{N-1} loss(f_{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{c}). \qquad (1)$$

The parameters θ correspond to the weights introduced before. One algorithm for this optimization is gradient descent, meaning that the parameters are updated by a term linear in the gradient of Eq. (1) with respect to θ. For a single update, the loss function must be evaluated for the whole training set. Stochastic gradient descent, however, requires only the estimation of the gradient of *loss*(*y*,*c*) for a single sample for one update [8], which proves beneficial in the quantum setting, where simulations can be very time consuming. A sample x_p and its label c_p are chosen uniformly at random from the training set and the parameters are updated according to

$$\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i - r\boldsymbol{\nabla}_\theta \quad loss(\boldsymbol{x}_p, \boldsymbol{c}_p). \qquad (2)$$

Here, the learning rate *r* is introduced, which controls the step size of updates and plays an important role in most ML algorithms as hyperparameter. If *r* is too large, the steps in parameter space may be too large and result in a chaotic behavior. Contrary, a too small *r* may drastically increase the number of iterations to reach a minimum, and it becomes more likely to be caught in a local minimum rather than reaching the global one. A proposed method to find a good starting *r* is to start with a small learning rate and increase it exponentially or linearly during a test run of training [9]. The sweet spot for an initial *r* is where the loss decreases most. This should be taken as a rule of thumb to find the order of *r*, meaning that a precise analysis of the test run is not required.

3 A variational quantum circuit approach to neural networks

A recently very popular approach to find and implement new hybrid QML algorithms are so called variational quantum circuits (VQC), which consist of a number of quantum gates with parameters that are optimized. These quantum circuits can be used to evaluate some cost function. To optimize the cost function, a variety of classical strategies can be used, which in turn may again employ a quantum circuit. Typically, the gate composition of a VQC is based on trial and error or a physically motivated ansatz [10], which shows that a theoretical framework is hardly established yet. Recent examples of VQCs aiming for near-term viability are proposed for instance in Refs. [7, 10–12].

In Ref. [7], Farhi and Neven propose a VQC with a sequence of unitary gate operations depending on continuous variables, used for binary classification. We present their framework in more detail in Sec. 3.1. Overall, their proposed model of a supervised quantum machine learning algorithm seems promising for implementation on actual quantum devices in the near future.

In our work, we follow the framework of Ref. [7] and aim for a more compact representation of input data. Unlike in the original work, we present an explicit decomposition of the formalism into singlequbit rotations and CNOTs, which form a universal gate set. Also, we use a different scheme for gradient estimation without multi-controlled gates. Our circuits for training are presented in Sec. 3.2 and can directly be applied to actual quantum devices. Finally, we will present a way to construct a dense presentation of the data in Section 3.3.

3.1 Theoretical framework

As applied in Ref. [7], a classical input bit string of length *n* represented by an integer $z \in \{0,\ldots,2^{n-1}\}$ is translated to the initial state $|z,1\rangle = |z_1,z_2,\ldots,z_n,1\rangle$ of a quantum register with $n+1$ qubits, which is called qubit encoding. We will regard the first *n* qubits as input register and the last qubit as readout qubit to estimate the classification of a sample eventually. A set of *L* parameter-dependent unitaries $\{U_k(\theta_k)\}_k$ acts on all qubits, where each unitary has the form $U_k(\theta_k) = e^{i\theta_k\Sigma_k}$. Here, the Σ_k are generalized Pauli operators and $\theta_k \in [0,2\pi]$ are the parameters to be adjusted during training of the network. We call this the classification circuit. To simplify notation, we summarize the applied unitaries as

$$U(\boldsymbol{\theta}) = U_L(\theta_L)\cdots U_1(\theta_1). \qquad (3)$$

The unitaries act on the register sequentially as shown in Fig. 2 for $L = 3$, which appears similar to a neural network with *L* layers, each with the same number of neurons. In this picture, the unitary operations play a similar role as weights in feed-forward neural networks. The state of the quantum register between two adjacent unitaries then corresponds to the states of neurons in the classical sense. Note, however, that the analogy to a feed-forward neural network is not perfect. In the quantum case there is no dissipation, since all operations are unitary. In the classical case, this may be interpreted as linear activation functions and neurons that take continuous values. Contrary, classical neural networks with non-linear activation functions are dissipative. Moreover, quantum mechanics allows for superpositions of computational basis states, whereas classical neurons are deterministic. This means we usually have to run our quantum classification circuit multiple times in order to accurately measure the final classification result.

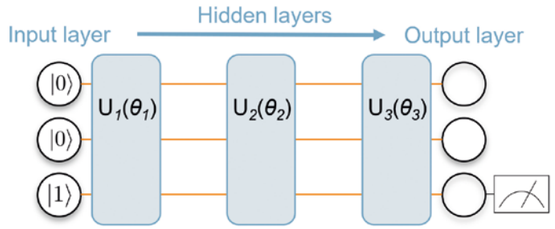


Figure 2: A quantum neural network consists of a layered structure with parameter-dependent unitary operations. The lines correspond to qubits, with the lowest one being the readout qubit and the other ones being the data qubits. In this small circuit the classical input (0,0) is trivially mapped to the computational basis state |00⟩, but also other encoding schemes are possible.

The expectation value of the final measurement operator Y_{n+1} is given by $\langle z,1|U^\dagger(\theta)Y_{n+1}U(\theta)|z,1\rangle$. From this, a cost or loss function *loss* can be introduced as

$$loss(\boldsymbol{\theta}, z) = 1 - l(z)\,\langle z,1|U^\dagger(\boldsymbol{\theta})Y_{n+1}U(\boldsymbol{\theta})|z,1\rangle\,, \qquad (4)$$

where $l(z) \in \{\pm 1\}$ is the correct label of an ut *z*. The loss can take values between 0 and 2. Obviously, if the expectation value yields the same value as the label, the loss vanishes, and takes its maximum if always the opposite label is measured. A loss of 1 corresponds to randomly guessing the classification with equal probability of outcomes.

To minimize the loss the θ_k are updated according to

$$\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i - r\left(\frac{loss(\boldsymbol{\theta}_i, z)}{\boldsymbol{g}^2}\right)\boldsymbol{g}, \qquad (5)$$

which is corresponds to the usual update rule of stochastic gradient descent in Eq. (2) up to the factor in brackets. This factor offers the advantage of adaptivity: gradients with a small norm are rendered more important than in Eq. (2) and the influence of gradients with larger norms is decreased. In this way the updating process is more stable than the non-adaptive variant.

As all unitaries are of the form (3), the partial derivative of the loss with respect to a parameter θ_k can be written as

$$\frac{\partial loss(\boldsymbol{\theta}, z)}{\partial\theta_k} = 2\,l(z)\,\mathrm{Im}\,\langle z,1|U_1^\dagger\cdots U_L^\dagger Y_{n+1}U_L\cdots\Sigma_k U_k\cdots U_1|z,1\rangle \qquad (6)$$

For clarity, we skipped the dependence on gate parameters θ_k here. To estimate the imaginary part in Eq. (6), we follow the strategy presented by Romero et al. in Ref. [11], which is schematically displayed in Fig. 3. Here, the classification circuit with an input state $|z,1\rangle$ is extended with an ancilla qubit and two controlled operations. The ancilla qubit is initialized to $|0\rangle$
and a Hadamard gate maps it to $|+\mathrm{i}$. After the k-th unitary, Σ_k is applied to the register controlled by the ancilla. Rather than performing a Y -measurement on the n + 1-th qubit as before, a controlled Y gate is applied instead. At the end of the circuit, the ancilla qubit is measured in the Y -basis. It can be shown that the probability to measure $|+\mathrm{ii}$ can be related to the imaginary part in Eq. (6) and the partial derivative of loss with respect to θ_k can be written as $2(2\mathrm{Pr}_Y[+\mathrm{i}]-1)$. A more detailed discussion is provided in the Supplementary Information (SI) [13]. Note that all applied operators in Eq. (6) are unitary, so the norm of the expectation value is bounded by 1. Thus, the norm of each partial derivative is bounded by 2, which means the norm of the whole gradient is bounded by 2L, yielding the advantage that the gradient cannot blow up as in other ML algorithms.

We now have all tools to train our network. Initially, all parameters are initialized uniformly at random. In each iteration, a sample is drawn from the training set and the quantum register of $n+1$ qubits is initialized accordingly

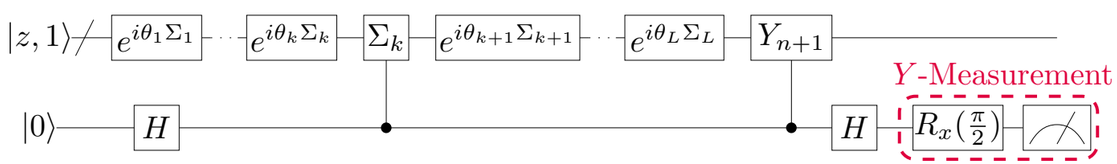


Figure 3: Schematic circuit used to estimate $\frac{\partial loss}{\partial\theta_k}$ for an input sample $|z,1\rangle$.

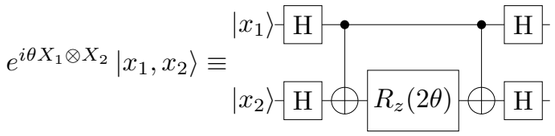


Figure 4: Implementation of the unitary $e^{i\theta X \otimes X}$ as quantum circuit. The other unitary $e^{i\theta X \otimes X}$ used in this work can be realized by omitting the Hadamard gates acting on the first qubit.

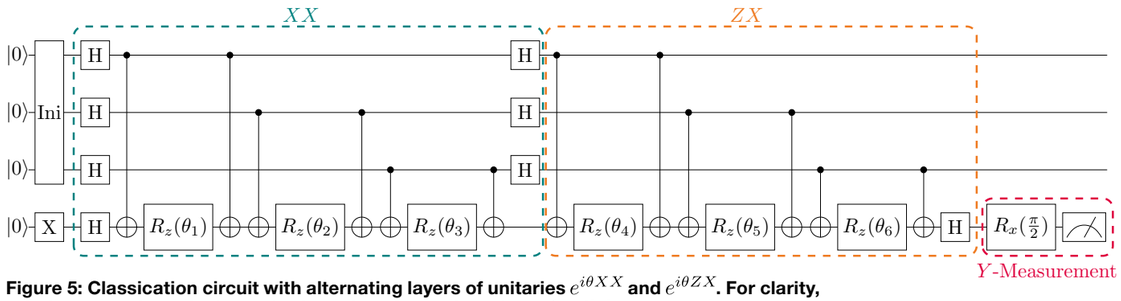


Figure 5: Classication circuit with alternating layers of unitaries $e^{i\theta XX}$ and $e^{i\theta ZX}$. For clarity, only three input qubits, the output qubit and one alternation of both layers are shown.

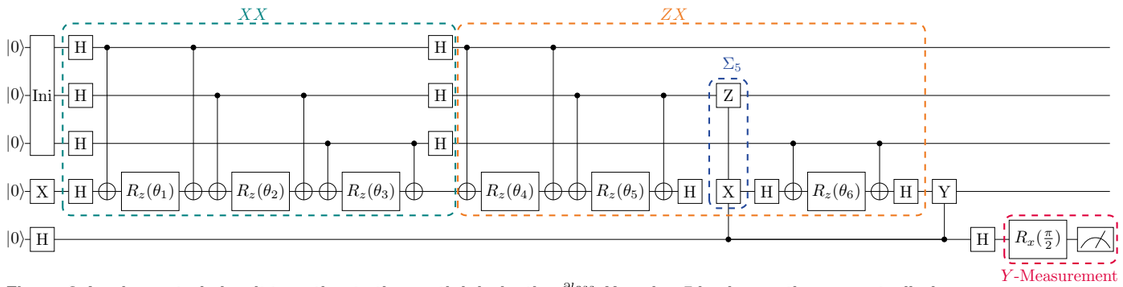


Figure 6: Implemented circuit to estimate the partial derivative $\frac{\partial loss}{\partial \theta_k}$. Here k = 5 is shown, thus a controlled Z-gate is inserted. In comparison to the classication circuit, an additional ancilla is required that controls the Σ- and Y -gates and is measured at the end.

using qubit encoding. By applying the classification circuit with L unitaries, the loss can be estimated by a Y measurement via Eq. (4). Using another ancilla qubit, the gradient is estimated with a circuit of $L + 2$ unitaries and the parameters are updated according to Eq. (5).

After testing sequences of randomly chosen Σ_k, the authors in Ref. [7] find that sequences of Paulioperators of the type $X_k \otimes X_{n+1}$ and $Z_k \otimes X_{n+1}$ performed best. The last operator of each always acts on the read-out qubit, the first on any of the others. This choice can be physically motivated: Depending on the state of a qubit of the input register, the readout qubit is rotated in different directions by a certain amount. Gates of the form $Z_k \otimes X_{n+1}$ thus rotate the last qubit by θ_k in mathematically positive direction if the corresponding input qubit is in state $|0\rangle$ and in the opposite direction if $|1\rangle$ is realized. Since quantum mechanics allows for superpositions of basis states, in general both rotations are performed with a certain probability. In this sense, the choice of gates seems reasonable, as $Z_k \otimes X_{n+1}$ considers the two computational basis states and $X_k \otimes X_{n+1}$ the relative phase between both. However, it is worth mentioning that this is an explanation why such a combination of gates seems reasonable, but it is not necessarily the best arrangement of gates one could find.

The proposed structure is to first apply the $X_k \otimes X_{n+1}$ unitary to each input and the output qubit and then the same again for the $Z_k \otimes X_{n+1}$ unitary. We regard this sequence as one alteration of first a XX-layer and then a ZX-layer. The number of alternations can be chosen arbitrarily, depending on the complexity of the dataset.

3.2 Explicit Quantum Circuits

Besides the simulation of the discussed quantum circuits we also aim for a framework that could directly be implemented on a real quantum device. Thus, we decompose each unitary into single-qubit rotations and two CNOTs, which together form a universal gate set. The decompositions are presented in Fig. 4.

Using this implementation, the classification circuit can be realized as shown in Fig 5. For reasons of clarity, only one alternation of gate layers is applied and $n = 3$. At the end, the readout qubit is measured in the Y -basis, which corresponds to a $\frac{\pi}{2}$-rotation around the X-axis followed by a measurement in the computational basis. For an input bit string of length n this implementation requires n+1 qubits and $4 \cdot n \cdot N_A$ CNOT-gates, where N_A is the number of alternations. The circuit used to estimate the partial derivatives $\frac{\partial loss}{\partial \theta_k}$ is presented in Fig. 6. For clarity, again a version for only three input qubits is shown. In the given illustration $k = 5$, which corresponds to a unitary in a ZX-block, therefore a controlled Z-gate is applied. Otherwise, if the unitary belongs to a XX-block, a controlled X-gate is applied.

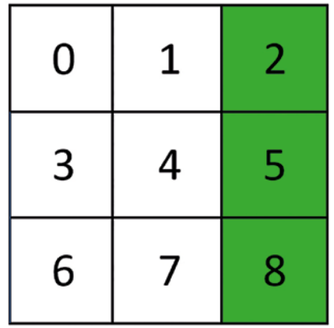


Figure 7: Example of an input bit string (0,0,1,0,0,1,0,0,1) of length 9 visualized as binary 3x3 pixel image. The pixels are numbered and bits with value 1 are colored in green.

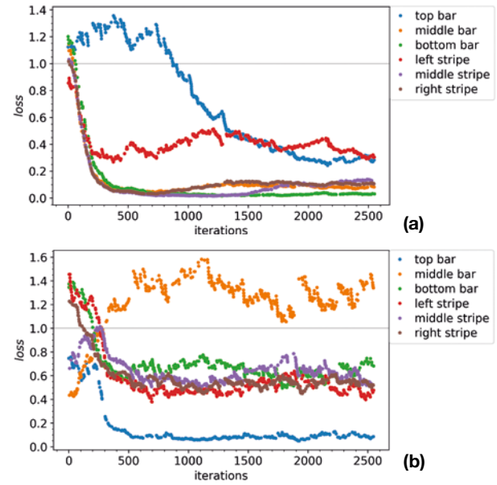


Figure 8: (a) Loss estimated during training with a constant learning rate r_2 = 0.04 and one XX- and ZX-block for trivial and (b) dense data encoding.

3.3 Dense presentation of data

In the original work by Farhi and Neven, classical input data are translated to a quantum register using qubit encoding. Now we discuss a more compact presentation of data, assuming that the inputs are binary bit strings of length N. Say $\sqrt{N}$ is an integer, then we can visualize the bit strings as square pixel images with binary values. Note that this is used rather for visualization and is no restriction. An example image is shown in Fig. 7, where the white elds are assigned with 0 and the greens with 1. The bits are numbered from 0 to N – 1. Using amplitude encoding, which means that a bit string is mapped to a superposition of computational basis states of a register with $n = \lceil \log_2 N \rceil$ qubits, the idea of our dense representation of data is to initialize the positions of green pixels in superposition. Choosing the white ones works just as ne. In the case of our example this is 1 p3 ($|2\rangle + |5\rangle + |8\rangle$). Our method assumes that there are no entirely white samples in the dataset. If such samples are present a dierent encoding scheme would be required. The number of qubits required to represent the data is logarithmic in the input size, oering a big advantage over qubit encoding. Especially on NISQ devices with limited number of qubits, such a scheme is preferable.

Initializing a general superposition is not as trivial as it may seem. Several approaches have been proposed to perform such an encoding of data into amplitudes of a quantum register efficiently, where often ancilla qubits are explicitly used. Thus, we follow a scheme proposed by Long and Sun in Ref. [14], which initializes a register in a normalized superposition of basis states

$$|\psi\rangle = \sum_{i=0}^{N-1} a_i |i\rangle, \qquad a_i \in \mathbb{C}, \tag{7}$$

without need for ancillary qubits. Their algorithm consists of $N-1$ single-qubit rotations, which may be controlled by k other qubits. A discussion of this algorithm including an exemplary quantum circuit to initialize the state $|\psi\rangle = \frac{1}{\sqrt{3}}(|2\rangle + |5\rangle + |8\rangle)$ is provided in the Supplementary Information [13].

4 Results

To test the performance of the presented algorithm with both schemes of data representation, we use a randomly generated dataset of bars and stripes. The samples consist of 3 × 3 pixel figures that contain either a bar or a stripe in one of the rows or columns, respectively. The simulations are implemented in Python, using the ProjectQ package [15] to set up the circuits. The actual computation is performed with the QX-Simulator back-end of Quantum Inspire [16]. The simulator back-end can easily be switched with a hardware back-end, which allows an implementation of our algorithm on a real quantum device.

4.1 Learning the model

First, a suitable learning rate should be estimated for this dataset by exponentially increasing r after each training iteration. Based on this test run, we perform the actual training of our VQC with three learning rates $r_1 = 0.01$, $r_2 = 0.04$ and $r_3 = 0.16$. The results of the test run are discussed in more detail in the Supplementary Information [13].

The resulting losses of the actual training for r_2 are plotted in Fig. 8a. The colors represent the different input samples presented during training. With increasing r, the loss fluctuates more, such that r_1 has smoother and r_3 more fluctuating branches, respectively, compared to Fig. 8a. Both

for r_2 and r_3 the loss of all samples drops below 1 during training, meaning that all samples are classified correctly, whereas for r_1 the loss of one sample is decreasing, but remains above 1 and thus this sample is classified incorrectly.

The evolution of the classification errors as well as the empirical risk E_N are compared in Fig. 9a and 9b for all three training runs. As only six different samples are considered, the classification error decreases step-wise for each additional sample that can be classified correctly. The empirical risk is a smoother measure of the learning performance and decreases more continuously. Note that with the definitions in Eq. (1) and (4), $E_N = 1$ corresponds to random guessing and $E_N = 0$ to perfect knowledge. Initially, E_N is slightly larger than 1 and decreases quickly for r_2 and r_3, whereas it decreases significantly slower for r_1. For r_3, E_N seems to settle around 0.21, for r_2 and r_1 it reaches 0.19 and 0.3, respectively, and seem to further decrease.

4.2 Dense presentation of data

The loss estimated during training with r_2 and a single alternation of an *XX*- and *ZX*-layer, for the bars and stripes data presented using amplitude encoding, is shown in Fig. 8b. Again the colors represent the input samples for which the loss is estimated. As in Sec. 4.1, the branches become smaller for smaller r, however, the losses fluctuate more for the amplitude encoding scheme and with r_3 no visible branches can be found anymore. The evolution of the empirical risks E_N for the three r are displayed in Fig. 9c. For r_1 and r_2, E_N settles at 0.61, the empirical risk of r_3 remains larger and fluctuates approximately around 0.71.

We further investigated whether more gate parameters help to reduce the stronger fluctuations by adding another alteration of an *XX*- and *ZX*-layer. However, we found no significant improvement.

4.3 Generalization

For both encoding schemes, the generalization performance is measured by omitting a pair of samples and training our model with the remaining training samples. Since training with r_1 shows the most stable behavior of the three learning rates discussed before, we choose this rate here. Afterwards, the left out samples are presented. We repeat this training for all 16 combinations of pairs and each 40 randomly chosen initial parameters. The averaged generalization errors are 0.51±0.36 and 0.50±0.29 for trivial and dense data encoding, respectively, meaning that statistically one of the two left out samples cannot be classified correctly. The averaged losses of the left out samples are 1.018±0.759 and 0.966 ± 0.450, respectively. With dense data representation the losses are slightly lower than with trivial data representation, however not significantly. As an example, Fig. 10 shows the evolution of loss of the middle bar and middle stripe, which were omitted during training, for both representation schemes.

5 Discussion

Whereas branches could be seen for all chosen learning rates with normal presentation of data, in the case of more compact presentation of data branches can not be found for the largest learning rate r_3, but are recovered for the smaller r_1 and r_2. The large fluctuations of the loss and also the empirical risk for r_3 can be explained by a too large step size in the parameter space, which leads to instability of training. From this we infer that the learning rate for the dense representation should be chosen smaller than for the trivial presentation of data.

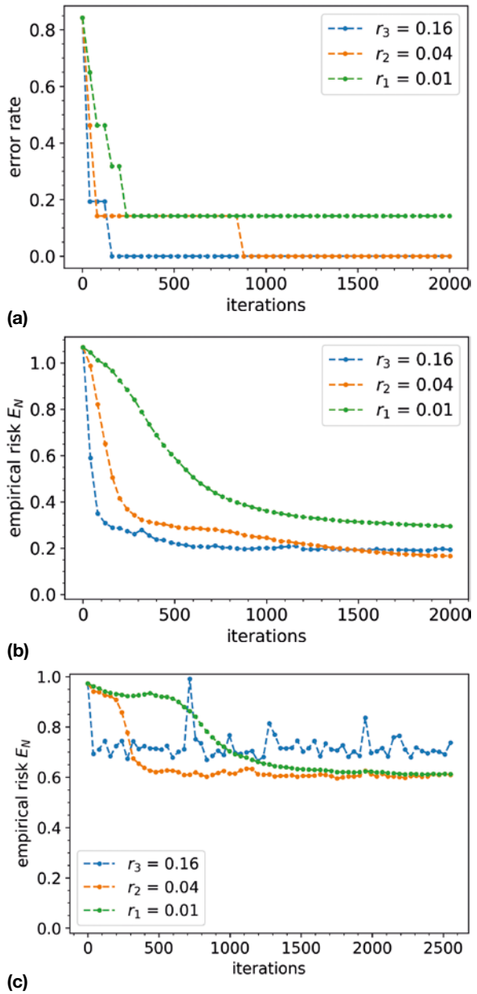


Figure 9: (a) Evolution of classification error and (b) empirical risk E_N during training with three different learning rates r and trivial data representation. (c) E_N for all r when presenting samples of the training set as superposition of basis states.

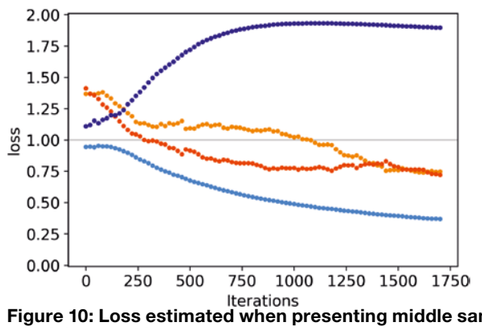


Figure 10: Loss estimated when presenting middle samples, which have not been presented during training, for trivial (blue) and dense (orange) data representation. The lighter color tones are the middle bars, darker ones the middle stripes.

For a clear comparison between the model with one and two alternations, further analysis is required. We expect that the overall performance does not improve significantly and we thus do not further investigate such a comparison.

As the estimated losses for all learning rates are mostly smaller than 1 for amplitude encoding, the classification errors are comparable to the trivial presentation of data. From this, we conclude that the generalization ability remains unchanged with our dense representation scheme, while the number of qubits is significantly reduced. However, the empirical risk of r_1 and r_2 settles at 0.61, which is significantly larger than what we achieve with a trivial data representation. This means that samples are classified correctly with a smaller probability than with a trivial presentation of data. Thus, in this respect, the model performs worse with the denser presentation. Possibly this is because the relevant information contained in the input samples is partially erased using the dense encoding scheme. To further analyze this potential trade-off between accuracy and compactness a more complex dataset should be considered.

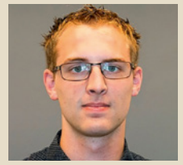
6 Conclusion

In this work we have implemented a variational quantum circuit being able to classify small artificial datasets with binary labels, based on the work in Ref. [7]. However, we have used a different scheme for gradient estimation and presented an explicit decomposition into single-qubit rotations and CNOTs. Moreover, our code could be applied to real quantum devices with only minor adaptions. We have considered the generic bars and stripes dataset and the model has learned to classify the inputs correctly. The loss of most samples has decreased and a small classification error has been reached. We have further introduced a different scheme to present data, which focuses on the positions of the bits being 1. This approach requires a quantum register that is logarithmic in the input size instead of polynomial, which can be important on near-term quantum devices with a restricted number of qubits. Using this presentation, our model has learned to correctly classify all bars and stripes with a comparable degree of generalization. However, in terms of the empirical risk the performance has been worse than for a trivial presentation of data. Therefore, the probability to classify data correctly has been smaller for our new scheme. We consider a more efficient representation of data as important for implementations on near-term quantum devices and more work on good encoding schemes is required, our scheme should be seen as small step towards this direction.

References **[1]** M. M. Waldrop. The chips are down for moores law. Nature News, 530(7589):144, 2016. **[2]** R. P. Feynman. Quantum mechanical computers. Optics news, 11(2):11–20, 1985. **[3]** P. W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In Proceedings 35th annual symposium on foundations of computer science, pages 124–134. IEEE, 1994. **[4]** J. Preskill. Quantum computing in the nisq era and beyond. Quantum, 2:79, 2018. **[5]** A. Perdomo-Ortiz, M. Benedetti, J. Realpe- G´omez, and R. Biswas. Opportunities and challenges for quantum-assisted machine learning in near-term quantum computers. Quantum Science and Technology, 3(3):030502, 2018. **[6]** M. Schuld, I. Sinayskiy, and F. Petruccione. An introduction to quantum machine learning. Contemporary Physics, 56(2):172–185, 2015. **[7]** E. Farhi and H. Neven. Classification with quantum neural networks on near term processors. arXiv preprint arXiv:1802.06002, 2018. **[8]** L. Bottou. Stochastic gradient descent tricks. Lecture Notes in Computer Science. Springer, 2012. **[9]** P. Surmenok. Estimating an optimal learning rate for a deep neural network. **[10]** M. Schuld, A. Bocharov, K. Svore, and N. Wiebe. Circuit-centric quantum classifiers. arXiv preprint arXiv:1804.00633, 2018. **[11]** J. Romero, R. Babbush, J. R. McClean, C. Hempel, P. J. Love, and A. Aspuru-Guzik. Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz. Quantum Science and Technology, 4(1):014008, 2018. **[12]** M. Schuld, V. Bergholm, C. Gogolin, J. Izaac, and N. Killoran. Evaluating analytic gradients on quantum hardware. Physical Review A, 99(3):032331, 2019. **[13]** See Supplementary Information available on https://tinyurl.com/y2myyc6d. **[14]** G.-L. Long and Y. Sun. Efficient scheme for initializing a quantum register with an arbitrary superposed state. Physical Review A, 64(1):014303, 2001. **[15]** D. S. Steiger, T. H¨aner, and M. Troyer. Projectq: an open source software framework for quantum computing. Quantum, 2:49, 2018. **[16]** QuTech. Quantum Inspire Home, 2018. https://www.quantum-inspire.com/ [Accessed: 29 July 2019].

Nicholas Meinhardt
Nicholas Meinhardt studies physics at ETH Zurich with focus on quantum optics and quantum information. As an intern at TNO he works on quantum algorithms for machine learning and recently started his graduation project in this field.

Bastiaan Dekker
Bastiaan Dekker is a scientist at TNO working on innovative signal processing methods such as machine learning. He studied applied physics.



Niels Neumann
Niels Neumann is a scientist at TNO. He works on applications of quantum computers and -networks. He studied mathematics and physics.

Frank Phillipson
Frank Phillipson is senior scientist at TNO. He leads the project team within TNO that studies applications and algorithms for near future use on quantum computers and quantum simulators. He studied econometrics and mathematics and has a PhD in applied mathematics.

Quantum Humanities:
A First Use Case for Quantum-ML in Media Science

Johanna Barzen, Frank Leymann

Quantum Humanities, the vison of combining quantum computing and digital humanities, is a promising new research field that aims at supporting digital humanities by using the advantages provided by the upcoming technology of the quantum computer for addressing existing as well as completely new questions in the humanities. To foster the vision of quantum humanities we want to outline a beneficial use case from the field of media science using machine learning algorithms implemented on quantum computers to solve issues from the humanities.

1 Introduction

The establishment of digital humanities as a research field has shown that the use of computers and techniques from computer science can contribute enormously to research done in the humanities [1]. Since quantum computers are getting real, it is promising to use the advantages of this upcoming technology for addressing existing as well as completely new questions in the humanities, as outlined in the vision for quantum humanities [2]. There are multiple benefits the quantum computer provides in specific areas compared to a classical computer, like being significantly faster [3], processing a vast amount of data in a single step [4] being more precise [5], solving problem classes that were previously considered practically unsolvable [4] or solving problems that can only be solved on a quantum computer [6]. Some of these benefits can in particular contribute to problems stated in the humanities, as shown below in a use case from our digital humanities project MUSE [7,8].

2 Use Case

MUSE includes a clothes- and costume repository supporting to capture, store and analyse clothes and costumes, focusing on costumes in films. The concrete costumes occurring in films are captured in their relevant details and are analysed to identify abstract costume patterns. At the current state there are more than 25.000 costume items with all their detailed attributes like colour, material, design, condition and way of waring [9] – to only name a few – stored in the MUSE database that need to be further analysed [10]. As Figure 1 depicts, we envision to address this with a combination of different techniques from machine learning containing an offline application part performing feature extraction via principle component analysis followed by clustering to identify new costume patterns and an online classification part, where new costumes are mapped to the already identified costume patterns.

This is where the use of the quantum computers can contribute: Implementations already exist for several steps outlined, that promise to be faster or more precise than on classical computers. These implementations promise to be advantageous even on nowadays quantum computers, so called NISQ machines [10]. To improve the rather time-consuming task of finding the right features building a multi-dimensional feature space, feature extraction can be supported by techniques that help to reduce the dimensions like quantum principal component analysis (PCA) [12]. PCA requires to determine eigenvalues and eigenvectors, thus, algorithms like Variational Quantum Eigensolver (VQE) [13], Phase Estimation [14] as well as HHL [15] can speed up this tremendously. Once the principal component from the costume parameters are identified, several implementations of clustering algorithms implemented on quantum computers, like Restricted Boltzmann Machines running on D-Wave [16] or Weighted Maximum Cut running on Rigetti [17], support the task to identify the clusters that hint to possible costume patterns. Whenever new costumes are added to the database, they need to be classified. To map the new clothes to the suitable cluster, classification algorithms realized on a quantum computer like Support Vector Machines implemented on the IBM quantum computer [18] promise a more precise classification due to a specific kernel function [5].

3 Conclusion and Outlook

This outlined vision for a quantum humanities use case from the media science wants to stress the potential benefits of using quantum computing technology for the humanities as well. Currently, we are in the process of implementing the stated tasks and plan to compare the achieved results with those achieved in a classical environment. With this use case we aim at providing first knowledge on how quantum applications in the digital humanities may be used in a beneficial way and to provide first, possibly reusable components for further use cases from different domains of the humanities.

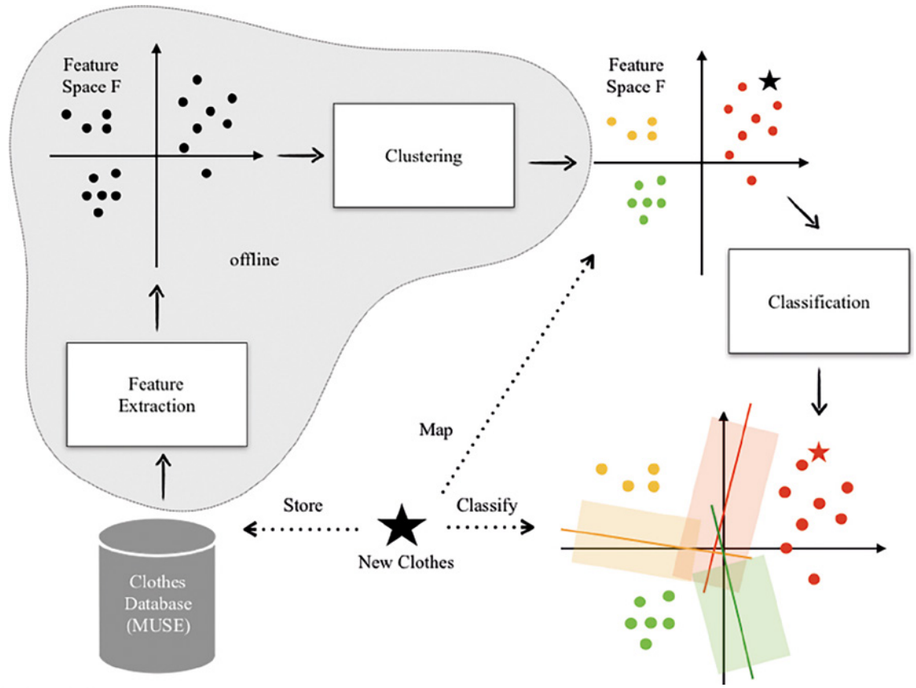


Figure 1: Analysing clothes.

References **[1]** D. M.Berry (ed.), Understanding Digital Humanities Palgrave Macmillan, 2012. **[2]** J. Barzen, F. Leymann, “Quantum humanities: a vision for quantum computing in digital humanities“ SICS Software-Intensive Cyber-Physical Systems, pp 1–6, 2019, DOI: https://doi.org/10.1007/s00450-019-00419-4. **[3]** T. F. Rønnow et al., “Defi ning and detecting quantum speedup“ Science Vol. 345(6195) July 2014. **[4]** M. A. Nielsen, I. L. Chuang, Quantum Computation and Quantum Information. Cambridge University Press 2010. **[5]** V. Havlicek et al, “Supervised learning with quantum enhanced feature spaces“ arXiv:1804.11326v2, 2018. **[6]** R. Raz, A. Tal “Oracle Separation of BQP and PH“ Electronic Colloquium on Computational Complexity Report No. 107, 2018. **[7]** MUSE. https://www.iaas.uni-stuttgart.de/forschung/projekte/muse/. **[8]** J. Barzen et al. “Wenn Kostüme sprechen könnten: MUSE - Ein musterbasierter Ansatz an die vestimentäre Kommunikation im Film“ In: P. Bockwinkel, B. Nickel, G. Viehhauser (ed.) Digital Humanities. Perspektiven der Praxis Frank & Timme, 2018, pp 223-241. **[9]** J. Barzen “Taxonomies of costume relevant parameters: approximating an ontology of the domain of film costumes“ Technical Report No. 2013/04, University Stuttgart, 2013, https://www.iaas.unistuttgart.de/publications/TR-2013-04_Taxonomien-_kostuemrelevanter_Parameter.pdf“. **[10]** J. Barzen, “Wenn Kostüme sprechen – Musterforschung in den Digital Humanities am Beispiel vestimentärer Kommunikation im Film“ Dissertation, Univerity of Cologne, 2018, https://kups.ub.uni-koeln.de/9134/. **[11]** L. Preskill, “Quantum Computing in the NISQ era and beyond“ Quantum 2, 79 (2018). **[12]** S. Lloyd, M. Mohseni, P. Rebentrost, “Quantum principal component analysis“ arXiv:1307.0401v2, 2013. **[13]** A. Peruzzo et al. “A variational eigenvalue solver on a quantum processor“ arXiv: 1304.3061, 2013. **[14]** D. Dervovic et al. “Quantum linear systems algorithms: a primer“ arXiv:1802.08227v1, 2018. **[15]** A. W. Harrow, A. Hassidim, S. Lloyd “Quantum algorithm for solving linear systems of equations“ arXiv: 0811.3171, 2008. **[16]** M. H. Amin et al. “Quantum Boltzmann Machine“ arXiv:1601.02036v1, 2016. **[17]** G. E. Crooks “Performance of the Quantum Approximate Optimization Algorithm on the Maximum Cut Problem “ arXiv:1811.08419v1, 2018. **[18]** IBM Q Experience. https://quantum-computing.ibm.com/. All links have been followed on September 11, 2019.

Johanna Barzen
Johanna Barzen studied media science, musicology and phonetics at the University of Cologne. Next to this she studied costume design at the ifs (international film school Cologne) and worked in several film productions in the costume department in different roles. Currently she is Postdoc and research staff member at the Institute of Architecture of Application Systems (IAAS) at the University Stuttgart performing research on Digital Humanities, Pattern Languages and Quantum Computing.

Frank Leymann
Frank Leymann is a full professor of computer science at University of Stuttgart, Germany. His research interests include service-oriented architectures and associated middleware, workflow- and business process management, cloud computing, patterns, and quantum computing. Frank is co-author of more than 450 peer-reviewed papers, about 70 patents, and several industry standards. He is elected member of the Academy of Europe.

Towards Understanding Approximation Complexity on a **Quantum Annealer**

Irmi Sax, Sebastian Feld, Sebastian Zielinski, Thomas Gabor, Claudia Linnhoff-Popien, Wolfgang Mauerer

Extended Abstract

Many industrially relevant problems can be deterministically solved by computers in principle, but are intractable in practice, as the seminal P/NP dichotomy of complexity theory and Cobham's thesis testify. For the many NP-complete problems, industry needs to resort to using heuristics or approximation algorithms. For approximation algorithms, there is a more refined classification in complexity classes that goes beyond the simple P/NP dichotomy. As it is well known, approximation classes form a hierarchy, that is, FPTAS ⊆ PTAS ⊆ APX ⊆ NPO. This classification gives a more realistic notion of complexity but—unless unexpected breakthroughs happen for fundamental problems like P = NP or related questions— there is no known efficient algorithm that can solve such problems exactly on a realistic computer. Therefore, new ways of computations are sought.

Recently, considerable hope was placed on the possible computational powers of quantum computers and quantum annealing (QA) in particular. However, the precise benefits of such a drastic shift in hardware are still unchartered territory to a good extent. Firstly, the exact relations between classical and quantum complexity classes pose many open questions, and secondly, technical details of formulating and implementing quantum algorithms play a crucial role in real-world applications.

Guided by the hierarchy of classical optimisation complexity classes, we discuss how to map problems of each class to a quantum annealer. Those problems are the Minimum Multiprocessor Scheduling (MMS) problem, the Minimum Vertex Cover (MVC) problem and the Maximum Independent Set (MIS) problem. We experimentally investigate if and how the degree of approximability influences implementation and run-time performance.

Our experiments indicate a discrepancy between classical approximation complexity and QA behaviour: Problems MIS and MVC, members of APX respectively PTAS, exhibit better solution quality on a QA than MMS, which is in FPTAS, even despite the use of preprocessing the for latter. This leads to the hypothesis that traditional classifications do not immediately extend to the quantum annealing domain, at least when the properties of real-world devices are taken into account. A structural reason, why FPTAS problems do not show good solution quality, might be the use of an inequlity in the problem description of the FPTAS problems. Formulating those inequalities on a quantum hardware (mostly done by formulating a Quadratic Unconstrained Binary optimisation (QUBO) problem in form of a matrix) requires a lot of hardware space which makes finding an optimal solution more difficult.

Reducing the density of a QUBO is possible by appropriately pruning QUBO matrices. For the problems considered in our evaluation, we find that the achievable solution quality on a real-world machine is unexpectedly robust against pruning, often up to ratios as high as 50% or more. Since quantum annealers are probabilistic machines by design, the loss in solution quality is only of subordinate relevance, especially considering that the pruning of QUBO matrices allows for solving larger problem instances on hardware of a given capacity. We quantitatively discuss the interplay between these factors.

Irmi Sax

Irmi Sax wrote her bachelor's thesis on the topic of solving optimisation problems with Quantum Annealing. Since 2018 she is studying the Master of applied Research at the OTH Regensburg work- ing on industrial applications of adiabatic Quan- tum Computing.

Gerrymandering as a Combinatorial Optimization Problem

Sebastian Feld, Maximilian Bley

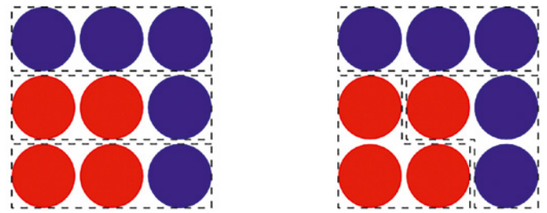


Figure 1: Example of Gerrymandering. Left: With regular district boundaries blue wins 1 district. Right: Irregular boundaries lead to blue winning 2 districts.

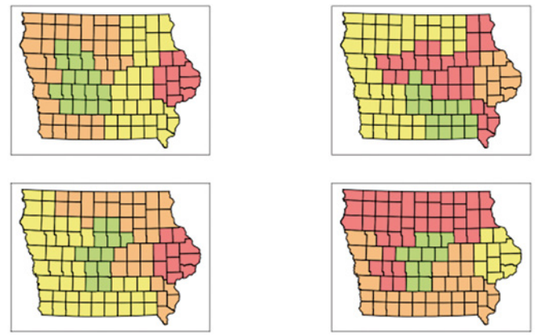


Figure 2: Real-world example of US state Iowa. The algorithm finds several electoral district allocations that lead to an advantage for a certain party.

1 Extended Abstract

This talk presents an approach that deals with electoral shifts, i.e. Gerrymandering, on a quantum annealer developed by D-Wave-Systems.

Gerrymandering is the manipulation of electoral district boundaries to give a certain party an advantage. The phenomenon occurs on the whole extent only in countries with a majority voting system. An electoral district victory means a seat in parliament, i.e. the candidate needs more than 50% of the votes. For an example, see Figure 1.

An algorithm is presented which examines combinations of possible electoral district allocations with the aim of maximizing the electoral district victories of a party while taking into account the classic criteria of the political districting problem. The problem is modeled as the NP-complete problem of exact coverage and formulated as a QUBO model. The algorithm addresses a real political districting issue in the US state of Iowa, see Figure 2.

Dr. Sebastian Feld

Dr. Sebastian Feld is head of the Quantum Applications and Research Lab (QARLab) at the Mobile and Distributed Systems Group of the LMU Munich. Currently, he pursues the goal of habilitation with a main focus being on optimization problems and the application of quantum technology. He joined LMU in 2013 and earned his doctorate in 2018 working on planning of alternative routes, time series analysis and geospatial trajectories.

A crystal-clear story for a meaningful future

In his famous Moon Speech, US President John F. Kennedy demonstrated how electrifying a vision of the future could be. He spoke to his nation through the metaphor of a hopeful, great story that culminated in 1969. America was to land on the moon before the end of the decade. Not because it was easy, but because it was hard. It was because of this story that the US invested billions in developments that were unthinkable before.

Kennedy‘s speech is a lesson for applied storytelling. In a motivating story a protagonist who fights as an active problem solver is the first thing you need.

KI pioneer Jürgen Schmidhuber formed his future story at the age of 15. He would build intelligence that would make him unnecessary as a human being. Today, billions of his neural networks are in use every day. He realizes the active neural networks that address the world‘s major problems.

Xi Yinping is starting with an equally great vision. With a crystal-clear story, he wants to make China the leading AI nation by 2030.

It is obvious: In the age of the beginning of artificial intelligence, the role of humans in competition and economy changes fundamentally. Those who now act without an image of the future are lost. But how can we, as entrepreneurs, develop such a narrative?

1. First of all, our pain must be significant enough. Those who work for a long time without a future story travel from one crisis to the next in visual flight. Europe, the people‘s parties, churches, banks, or schools - they all make themselves vulnerable through their lack of vision. Instead of acting and directing, they have to put up with the losses of citizens, voters, encouragement, members, and turnover.
2. We must be aware that a mature, clear story needs adequate development time.
3. To develop this vision, we need to engage in dialogue with all stakeholders. With our colleagues, employees, customers, competitors, and experts.
4. Now the „wisdom of the many“ comes to light. Our task is to moderate them sensitively until the story is understandable and round.
5. We must open ourselves to unbridled learning and be prepared to overcome our limited perspective in favor of profitable insight. We also listen to our critics.
6. We set up a process - at best with an expert.
7. To back up our story, we compare ourselves with all our major competitors - and outperform them.
8. If our future story stands, we can entirely derive all the measures from it.

Believe uninhibitedly in your future. Win and find the employees, form teams that inspire and deploy you. With your future story, you will have a lucid, pure, and detailed visual and narrative imagination that you can supplement with critical figures and agile methods. This story will be your new world map to navigate safely into the future.

Uwe Walter is a storytelling and change expert for media and industrial enterprises. He advises clients as diverse as YouTube stars, start-ups, bloggers, publishers, radio and television stations and film productions. His expertise: How do I generate reach through future-proof storytelling?

Foto: Privat

IMPRESSUM

VERLAG
Vogel Communications Group GmbH & Co. KG,
Max-Planck-Str. 7/9, 97064 Würzburg, www.vogel.de
Geschäftsführer
Matthias Bauer, Florian Fischer, Günter Schürger

REDAKTION
Chefredaktion Claudia Linnhoff-Popien (V. i. S. d. P.)
Chef vom Dienst Robert Müller
Fachbeirat Patric Fedlmeier, Norbert Gaus, Sandro Gaycken, Michaela Harlander, Markus Heyn, Martin Hofmann, Manfred Klaus, Andrea Martin, Niko Mohr, Christian Plenge, Frank Rosenberger, Ralf Schneider, Stephan Schneider, Marc Schröder, Uwe Walter, Michael Zaddach
Redaktion Hannes Mittermaier, Florentina Hofbauer
Blog Steffen Illium, Tanja Zecca, Tamara Tomasevic
Redaktionsassistenz Katja Grenner
Mitarbeiter dieser Ausgabe Thomy Phan, Kyrill Schmid
Schlussredaktion Barbara Haber

ANFRAGEN AN DIE REDAKTION
redaktion@digitaleweltmagazin.de

GRAFIK
Layout Stefan Stockinger, www.stefanstockinger.com

ANZEIGEN
Ansprechpartner Tanja Zecca, Tel. +49 89 2180-9171,
E-Mail: anzeigen@digitaleweltmagazin.de
Es gilt die gültige Preisliste, Informationen hierzu unter www.digitaleweltmagazin.de/mediadaten

HERSTELLUNG
ColorDruck Solutions GmbH,
Gutenbergstraße 4, 69181 Leimen

ABO-SERVICE
DataM-Services GmbH, Aboservice Digitale Welt,
Franz-Horn-Str. 2, 97082 Würzburg,
Tel. +49 931 4170-435
E-Mail: abodigitalewelt@vogel.de
Digitale Welt erscheint einmal pro Quartal

ABONNEMENT-PREISE
Jahres-Abo inklusive Versandkosten: Inland 78,00 €, Ausland 87,60 €; ermäßigtes Abo für Schüler, Studenten, Auszubildende: Inland 39,00 €
Der Bezug der Zeitschrift Digitale Welt ist im Mitglieds-Beitrag des Verbandes VOICE - Bundesverband der IT-Anwender e.V., Digitale Stadt München e.V. und Hannover IT e.V. enthalten.

HERAUSGEBER
Prof. Dr. Claudia Linnhoff-Popien, Institut für Informatik, Ludwig-Maximilians-Universität München, Oettingenstr. 67, 80538 München,
Tel. +49 89 2180-9153, www.digitaleweltmagazin.de

CALL FOR CONTRIBUTION
for DIGITALE-WELT-Blog

The next DIGITALE WELT will be released on 06.03.2020

Become part of our top-class authorship and place your digital topics of tomorrow on today's platform with **850.000*** clicks so far.

OUR TOPICS ARE:

- ✓ **Machine Learning**
- ✓ **Quantum Computing**
- ✓ **Internet of Things**
- ✓ **Blockchain**
- ✓ **Cyber Security**
- ✓ **Human Resource**

GAINED YOUR INTEREST?
Then do not hesitate to contact the DIGITALE WELT -Office via Mail: **blog@digitaleweltmagazin.de**

***Our articles were published online at www.digitaleweltmagazin.de/blog and achieved the above-mentioned number of clicks in the period 01. August 2017 – 29. October 2019.**

Guide to the publication of technical papers

PLEASE NOTE THE FOLLOWING POINTS FOR YOUR SUBMISSION:

1. Your paper meets the following requirements:
   - Contentwise orientation at the topics of the DIGITALE WELT
   - Title (max. 60 characters incl. spaces)
   - Blog post length: 7,000-15,000 characters including spaces
   - Full article length: 15.000 - 40.000 characters including spaces
   - Written exclusively for DIGITALE WELT
   - All graphics and pictures are free of rights
   - Does not contain any advertising
2. CV and picture of the author:
   To introduce you as an author we need:
   - Your full name
   - Any academic titles
   - Position in the company (max. 40 characters)
   - Name of your company (max. 25 characters)
   - Portrait image with min. 300 DPI resolution
   - CV with max. 300 characters incl. spaces
3. Consent to Publish:
   For publication in print & online media, we require the fully completed and signed declaration of consent. Please find it at digitaleweltmagazin.de/consent

YOU WILL BENEFIT FROM THE FOLLOWING SERVICES:

- Your high quality contribution will be published in our Online Blog of the DIGITALE WELT Magazine
- The best contributions are to be published in our print magazine, additionally
- High range by distribution via Social Media
- This service is of course free of charge for you

Send us your complete documents by use of our online form at **digitaleweltmagazin.de/submit**

A listing of our current and past „Call-For-Contributions" can be found at digitaleweltmagazin.de/calls

We are looking forward to your technical contribution with your expert knowledge.

Your DIGITALE WELT Team

Im Lauf der Geschichte
gab es schon immer Probleme.
Niemand mag sie.
Aber Probleme treiben uns an,
Dinge besser zu machen.

Smart loves problems.

Wissenslücken, Stromausfall, Verkehrschaos
oder Plastik in den Weltmeeren.
Sie haben uns sogar zum Mond und zurück gebracht.
Mit einer Tankfüllung.
Tatsächlich bringen Probleme die Welt weiter voran.
Und die Menschen, die diese Probleme angehen,
arbeiten bei jedem Schritt eng mit uns zusammen.

ibm.com/smart/de

Let's put smart to work.™

IBM