Australian National University

ANU-UI RESEARCH INTERNSHIP IN COMPUTER SCIENCE 2019

1. BACKGROUND

The ANU-UI Research Internship in Computer Science is an initiative between the Research School of Computer Science (RSCS) at the Australian National University (ANU) and the Faculty of Computer Science (Fasilkom) at Universitas Indonesia (UI), that is part of a broader research collaboration between the two universities. Under this Research Internship program, RSCS will host students or staff members from Fasilkom UI to participate in short-term research projects supervised by academic staff at RSCS.

The aims of this program are

- to strengthen research collaborations between the universities by engaging staff and students in both universities in joint research projects, and
- to provide a pathway for UI undergraduate students, or UI staff who have not held doctoral degrees, to be considered for admission in the ANU PhD program. The PhD admission at RSCS requires the candidates to have an equivalent of ANU Bachelor's degrees with Honours. A successful completion of the summer research program may be considered as having completed the equivalent of a 12 credit unit research project, which is a minimum requirement for admission into ANU PhD program.

2. ELIGIBILITY

For the initial phase, we will consider hosting 5 (five) scholars from UI. These will comprise of at least 3 (three) undergraduate students enrolled in Fasilkom UI in their final year of study. Fasilkom UI staff will be eligible to participate in this program subject to this minimal quota for students, so at the maximum, we will consider admitting 2 (two) UI teaching/research staff who do not yet hold a PhD degree.

The number and the composition of the eligible scholars are subject to change in the future depending on availability of funding and supervisors.

The UI student applicants should be enrolled or plan to enroll in the final year project course at UI, since the intention of the program is to encourage students to undertake further research degrees.

Applicants should have GPA of at least 3.2 (out of 4), which is the equivalent of ANU First Class Honours.

3. FUNDING AND DURATION OF THE PROGRAM

ANU will cover local accommodation costs, local transports and incidental costs of the scholars. ANU will help the scholars in securing accommodation in Canberra.

The scholars are expected to find funding to cover their return airfares. In extenuating circumstances, we may consider covering airfares for exceptional students. The determination of who may be eligible will be decided on a case-by-case basis. The scholars are expected to arrange for their own travel insurance.

4. APPLICATION PROCEDURE

The application for the Research Internship program must be done through the online application website at:

http://tiny.cc/anu-ui-intern-2019

Application is open on June 18th, 2019 and closes on July 15th, 2019.

The list of supervisors and research projects available for this round of application is given in the Appendix. Each applicant is allowed to indicate, through the online application form, three (3) projects they are interested to apply to.

The applicants must also send the following supporting documents separately to studentexp.cecs@anu.edu.au:

- Curriculum vitae
- The most up-to-date transcript
- Two academic referee reports

The outcome of the selection process will be announced on July 23rd, 2019. Successful applicants will be provided with further information on the onboarding process.

5. ASSESSMENT

The scholars will be required to write a 2-page report and to give presentations at ANU at the conclusion of their projects, but there will be no official credits given or assessment done from the ANU side. However, undergraduate students who are taking a final year project course ('Tugas Akhir') at UI may consider writing up a research report on their work at ANU, to be evaluated by UI as per normal evaluation process at UI. ANU supervisors or coordinator will assist in the evaluation should it be deemed necessary.

6. VISA REQUIREMENT

The scholars are responsible for arranging the appropriate visas (if required) to visit Australia. The ANU contact person (see Contact) will assist in providing supporting documentations relating to the purpose of the visits.

7. IMPORTANT DATES

Application opens:	18 June 2019
Application closes:	15 July 2019
Results announced:	23 July 2019
Research visit starts:	23 September 2019
Research visit ends:	8 November 2019

8. CONTACT

ANU coordinator: Alwen Tiu (alwen.tiu@anu.edu.au)

UI coordinator: Ari Saptawijaya (saptawijaya@cs.ui.ac.id)

Australian National University

APPENDIX. SUPERVISOR AND PROJECT LIST

PROFESSOR STEVE BLACKBURN

I am a professor of computer science at the Australian National University. My primary research focus is on programming language implementation. I have published in venues that specialize in programming languages, memory management, operating systems, virtual machines, computer architecture, compilers, performance analysis and software engineering. I am a Fellow of the ACM.

Research
My research focus is on programming language implementation and performance analysis. My interests include garbage collection, just-in-time compilation, computer architecture, operating systems, and security. I am a strong believer in shared research infrastructure, leading the DaCapo benchmark project and the MMTk memory management framework.

Garbage Collection
Garbage collection is a major focus of my research. I am particularly interested in novel garbage collection algorithms, the detailed performance analysis of garbage collection, the design of novel high performance garbage collection mechanisms, and the implementation and software engineering of high performance garbage collectors. I am also interested in architectural support for garage collection. I lead the development of the MMTk memory management framework.

Language Implementation
The context for most of my research has been language implementation. I am interested in the software engineering of high performance language runtimes, and also the social dimension of successfully creating large software artefacts within the research community.

Performance Analysis
I have a major interest in performance analysis. I am interested in performance analysis techniques, benchmark suites, and sound methodology. I lead the development of the DaCapo benchmark suite.

Email: steve.blackburn@anu.edu.au
Website: http://users.cecs.anu.edu.au/~steveb/

AVAILABLE PROJECTS WITH PROFESSOR STEVE BLACKBURN

[SB1] RESEARCH PROJECTS IN PROGRAMMING LANGUAGE IMPLEMENTATION

Please contact Prof Blackburn to discuss details of available projects.

Australian National University

DR SID CHAU

Sid Chi-Kin Chau is with the Research School of Computer Science at the Australian National University. He was an Associate Professor with the Department of Computer Science at Masdar Institute, which was created in collaboration with MIT, and is a part of Khalifa University.

His research interests are related to the computing systems and applications for sustainable smart cities by applying Internet-of-things, computational intelligence, advanced algorithms and big data analytics to develop sustainable solutions for smart cities (including smart grid, smart buildings, intelligent vehicles and transportation). He also researches in broad areas of algorithms, optimization, and Internet-of-things.

Previously, he was a visiting professor with MIT, a senior research fellow with A*STAR in Singapore, a Croucher Foundation research fellow with University College London with a research fellowship awarded by the Croucher Foundation Hong Kong, a visiting researcher with IBM Watson Research Center and BBN Technologies, and a post-doctoral research associate with University of Cambridge. He received the Ph.D. from University of Cambridge with a scholarship by the Croucher Foundation Hong Kong, and B.Eng. (First-class Honours) from the Chinese University of Hong Kong.

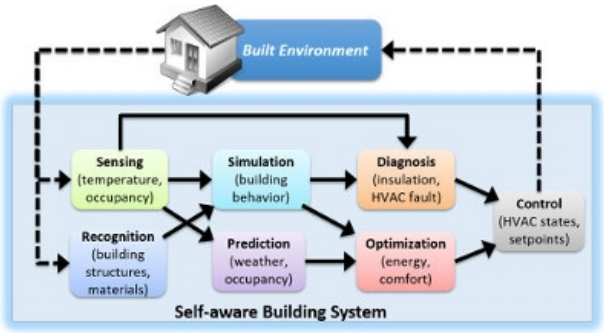
He is on the TPC of several top conferences in smart energy systems and smart cities, such as ACM e-Energy, ACM BuildSys, ACM IoTDI. He is a TPC co-chair of ACM e-Energy 2018, and guest editor for IEEE Journal on Selected Areas in Communications Special Issue on Design and Analysis of Communication Interfaces for Industry 4.0, IEEE Journal of Internet-of-Things Special Issue on Internet-of-Things for Smart Energy Systems, and IEEE Transactions on Sustainable Computing Special Issue on Intersection of Computing and Communication Technologies with Energy Systems. He is an associate editor of IET Smart Grid. He was a co-founder of a start-up specializing in intelligent systems and big data analytics for smart building management.

Email: sid.chau@anu.edu.au
Website: https://users.cecs.anu.edu.au/~sid.chau/

Australian National University

AVAILABLE PROJECTS WITH DR SID CHAU

[SC1] ADVANCED SMART BUILDINGS WITH INTERNET-OF-THINGS AND PRIVACY-PRESERVING OPTIMIZATION



Buildings are among the largest consumers of energy, topping 40% of total energy usage in many countries. A significant portion of energy use in buildings is attributed to the heating, ventilation, and air conditioning (HVAC), which accounts for up to 50% of the total energy consumption in buildings.

In this project, we design and develop intelligent building management system for the next-generation smart buildings, which can sense occupants' preferences and status in a privacy-preseriving manner and subsequently optimize building usage energy-efficiently. We will utilize Internet-of-Things systems, such as wireless sensors, mobile devices, and smart appliances.

See more related projects at: http://smartsustainability.org

[SC2] BLOCKCHAIN-BASED DECENTRALIZED AUTONOMOUS ORGANIZATIONS

This project designs and develops effective algorithms and systems to support blockchain-based decentralized autonomous organizations, with applications to transactive energy, IoT, and smart cities.

See more related projects at: http://smartsustainability.org

[SC3] PEER-TO-PEER SHARING ECONOMY FOR ENERGY AND TRANSPORTATION IN SMART CITIES

Peer-to-Peer Ride Sharing

Peer-to-Peer Energy Sharing

Peer-to-peer sharing economy is becoming a new social paradigm empowered by intelligent and networked systems. We explore novel algorithmic and system designs to enable future peer-to-peer sharing economy in energy and transportation services.

In this project, we design and develop decision support tools and algorithms for peer-to-peer sharing economy. We will implement our ideas in smartphone apps to solve real-world sharing problems of energy and transportation services.

See more related projects at: http://smartsustainability.org

DR. PATRIK HASLUM

Patrik is a lecturer and researcher in the AI Planning and Optmisation group at ANU. Most of his work in AI planning is about generating optimal plans, i.e., plans that have minimum cost or execution time. Mostly through the use of heuristic search. But he is interested in other stuff too, for example, analysing/understanding classes of problems ("domains") used as benchmarks in evaluation of planning algorithms, and tractable subclasses of planning problems.

Email: patrik.haslum@anu.edu.au

Website: http://users.rsise.anu.edu.au/~patrik/

AVAILABLE PROJECTS WITH DR HASLUM

[PH1] OPTIMISATION WITH IMPRECISE BOUNDARIES

Cutting problems (https://en.wikipedia.org/wiki/Cutting_stock_problem), in which we seek to maximise the utilisation of a piece of raw material, are commonplace in operations research. Most research has studied problems in one or two dimensions. The goal of this project is to solve a cutting optimisation problem in 3D. Moreover, the problem is complicated by the fact that the boundaries into which the target shape must fit are not sharp, as the quality of the raw material is not even. We must trade off between making the most use of the raw material and the quality of what that goes into the product.

[PH2] AUTOMATED PLANNING FOR CYBER-SECURITY RED-TEAMING

Cyber security evaluation is often carried out through security professionals mounting a "simulated" attack on the target network in order to find its weaknesses. The goal of this project is to automate some of this process, using AI planning techniques. From an attacker's point of view, the planning problem features uncertainty (the attacker has a limited view of what is on a network or host) and multiple criteria to optimise (speed, stealth, effort, etc). Challenges in making this work are many: How to derive planning models from the information about security vulnerabilities that is available, and how to obtain realistic estimates of the information that is not? How to exploit the structure of the problem to achieve both scalable planning (making plans for networks with hundreds or thousands of hosts) while making realistic assumptions. Finally, many types of cyber attacks are not only technical but target people's and organisation's vulnerabilities. How to incorporate those in a planning model is one more open research question.

Australian National University

PROFESSOR TONY HOSKING

I am a professor of computer science, with academic appointments at Purdue University (on leave from August 2015) and the Australian National University, contributing also as a researcher with Data61, Australia. I also consult regularly in intellectual property litigation as a technical expert. I studied computer science at the University of Adelaide, the University of Waikato, and the University of Massachusetts at Amherst, receiving BSc, MSc, and PhD degrees, respectively. My research interests lie in the area of programming language implementation, and I work on problems arising in object persistence, object databases, distribution, memory management (*garbage collection*), managed language runtimes, language virtual machines, optimizing compilers, and architectural support for programming languages and applications.

I am a Member of the IEEE and Life Member of the Association for Computing Machinery, and was named a Distinguished Scientist of the ACM in 2012.

I am co-author of the primary reference work for automatic memory management *The Garbage Collection Handbook*, now translated to Japanese and Chinese!

Email: antony.hosking@anu.edu.au
Website: http://hosking.github.io

AVAILABLE PROJECTS WITH PROF HOSKING

[TH1] RESEARCH PROJECTS IN PROGRAMMING LANGUAGE SYSTEMS.

Please contact Prof Hosking to enquire about the available projects.

Australian National University

DR HANNA KURNIAWATI

Hanna Kurniawati is a Senior Lecturer with ANU and CS Futures Fellowship at the Research School of Computer Science, ANU. She is also a Research Affiliate at the Dept. of Mechanical Engineering, Massachusetts Institute of Technology (MIT). She earned a Ph.D. in Computer Science for work in robot motion planning from National University of Singapore and is an alumnus of Fasilkom UI (NPM: 1298000424 :)).

Hanna's research interest spans artificial intelligence and robotics. She has been working on autonomous decision making under uncertainty —developing algorithms to enable (semi)-autonomous agents to operate robustly, despite imperfect sensing, despite not knowing the exact effect of actions, and despite a lack of information and understanding about the environment. Recently, she started to work on the combination of planning and learning for robust decision making, with applications on robot manipulation and the assurance of autonomous systems.

Email: hanna.kurniawati@anu.edu.au
Website: http://users.cecs.anu.edu.au/~hannakur/

AVAILABLE PROJECTS WITH DR KURNIAWATI

[HK1] AUTOMATIC TESTING FOR SOFTWARE THAT CONTROLS DRIVERLESS CARS

Unexpected response to rarely seen sensor inputs have caused catastrophic (driverless) car crash. Could we design an automatic testing mechanism to reduce the chances of such potentially disastrous behaviours from occurring? To start answering this question, we will explore a simple planning and learning approach to automatically and efficiently generate simulated sensor inputs that are likely to trigger undesirable robot behaviour.

This project can be expanded into a PhD topic.

Expected background:

- Fluent in C++/Python (higher preference for C++) programming.
- Have a good understanding of abstract data structures.
- Have a good understanding of basic probability.
- Experience with robotics simulator (e.g., Gazebo or VREP) is a plus.
- Good results in Artificial Intelligence course would be a plus.

Australian National University

[HK2] USING TOOLS FOR ROBOT MANIPULATION

The ability to use tools to manipulate objects is believed to require a sophisticated level of cognition. Despite the tremendous advances in robotics over the past decade, such a level of cognition is rarely explored and remains an open problem in the robotics domain. This project will explore methods to enable the use of tools in robot manipulation.

Of course, as a 10-week project, we will simplify the problem significantly. For instance, by limiting the type of manipulation and tools available to the robot. We will adopt one of the decision-making algorithms we have developed in-house to solve this simplified problem.

This project can be expanded into a PhD topic.

Expected background:

- Fluent in C++/Python (higher preference: C++) programming.
- Have a good understanding of abstract data structures.
- Have a good understanding of basic probability.
- Experience with robot and robotics software (e.g., Robotics Operating System) is a plus, though not necessary.

Good results in Artificial Intelligence course would be a plus.

Australian National University

ASSOCIATE PROFESSOR DIRK PATTINSON

Bio: Dirk Pattinson holds a MSc in Mathematics and a PhD in Computer Science from the University of Munich, Germany. He was teaching at Imperial College London before moving to ANU in 2012. His research interests span computational logic in a broad sense, electronic voting and the foundations of computation.

Email: dirk.pattinson@anu.edu.au
Website: http://users.cecs.anu.edu.au/~dpattinson/

AVAILABLE PROJECTS WITH PROF PATTINSON:

[DP1] CERTIFIED DECISION PROCEDURES FOR THE SEQUENT CALCULUS.

There's a lot of work that we're doing on the formalisation of sequent calculi

https://en.wikipedia.org/wiki/Sequent_calculus#The_system_LK

where the idea is that the formalisation gives a provably correct decision procedure of whether things are provable or not. This doesn't work for first order logic, but for a large range of systems of modal logic.

One problem is that verified implementations are correct, but usually slow. One way around that that we are exploring right now is to have a fast implementation that generates a proof or a counter proof / counter model. So to guarantee correctness, one only needs to check the (counter) proof -- which should be done in a verifiable way. That is, proof generation and proof checking are developed in tandem.

[DP2] FAIRNESS OF VOTING PROTOCOLS.

There is a paper by Rivest and Shen

https://www.stat.uchicago.edu/~lekheng/meetings/mathofranking/ref/rivest.pdf

that defines an arguably reasonable measure on the quality / fairness of a single winner voting protocol. Generalising this to multi winner election schemes like the ones used in Australia would allow a systematic comparative study of voting schemes in use. The Australian Electoral Commission publishes all ballots for recent elections, so that one could (i) define the measure, (ii) argue that it is "good" by mathematically proving some desirable properties (e.g. electing a smith set is always better than not to), and (iii) use the measure on real world data to compare the different systems.

[DP3] CERTIFIED VOTING

The idea is to take a voting protocol, e.g. Kemeny Young

https://en.wikipedia.org/wiki/Kemeny–Young_method

and develop a "certificate" for it. That is, a run of the method would produce winner(s) together with a proof of the fact that these are *really* the winners. This allows us to have a fast implementation (similar to the proof generation / proof checking tandem) together with a verified checker. This is called "universal verifiability" in the voting literature: any member of the public can check the proof and convince themselves that the correct winner(s) have been chosen.

Kemeny Young is interesting as the runtime is exponential, i.e. it is not necessarily feasible to re-compute, but one can develop a small (polynomial size) certificate that proves correctness of the count.

[DP4] STABILITY OF VOTING PROTOCOLS

We know that there is some margin of error when it comes to feeding ballot papers into electronic counts. Ballot papers are e.g. processed by OCR, and there's the usual error that confuses the number "1" and the number "7" for example. This motivates to analyse voting protocols as to their stability. Intuitively, a protocol is more table than another one, if I can change a larger number of votes without changing the result. The goal would be to develop a notion of stability (this would be through monte carlo simulations as with most protocols it's nearly impossible to calculate this exactly), and then run some experiments, to again quantitatively analyse different protocols. There is no literature on this that I know of.

[DP5] AUDITS

The challenge here is to confirm that the results of an election are correct with high probability. The result that one is after here is that to confirm with a likelihood of say 99%, one needs to re-visit x% of the ballot papers (we hope that x would be small), and re-compute the result. To do this, one needs to compute the margin of winning, that is, the least number of ballot papers that need to be changed to change the result. This is very hard for schemes used in Australia, but doable for a system that we are analysing right now, the so-called Schulze Method:

https://en.wikipedia.org/wiki/Schulze_method

See here:

https://www.stat.berkeley.edu/~stark/Preprints/gentle12.pdf

for some background on risk limiting audits. The idea would be to develop this for Schulze.

Australian National University

DR ALWEN TIU

I am a Senior Lecturer at the Research School of Computer Science, The Australian National University. I obtained his BSc in Comp Sci from Universitas Indonesia, Masters in Computational Logic from TU Dresden and PhD in Comp Sci from Penn State University. My main research interests span theoretical as well as practical aspects of computer science; these include formal methods, computational logic, automated theorem proving and computer security. More specifically, I am interested in modelling aspects of computational systems (such as parts of operating systems, communication protocols, simple authentication devices, etc) as mathematical theories, and developing tools and techniques to prove their correctness or to find potential flaws.

Email: alwen.tiu@anu.edu.au
Website: http://users.cecs.anu.edu.au/~tiu

AVAILABLE PROJECTS WITH DR TIU

[AT1] TYPE INFERENCE FOR A PERMISSION-DEPENDENT SECURE INFORMATION FLOW TYPE SYSTEM

Information flow analysis of a program can be done statically by reasoning about the potential propagation of information within a program if a formal semantics of the programming language is available. In general, determining whether a sensitive information can be potentially leaked by a program is an undecidable problem, so in practice various approximation methods are used, e.g., via type systems. In this project we will look at using a type system to check whether a program is secure in the sense that it does not leak any information of data that a user marks as sensitive. The programming language used in this study is a simplified and idealised version of Java, that assumes additionally a permission-based access control mechanism similar to that used in Android operating system. A type system that information flow for this language has been proposed in a prior work of Chen et. al., where a permission-dependent type system is developed and proved sound, i.e., if a program is typeable in this type system then it is guaranteed to secure, under certain assumptions. This project will look at the type inference problem – that is, given a program and a desired type (encoding a desired secure information flow policy), decides whether the program can be typed using the given type.

Reference:
Hongxu Chen, Alwen Tiu, Zhiwu Xu, Yang Liu: *A Permission-Dependent Type System for Secure Information Flow Analysis.* In Proceedings of the 31st IEEE Computer Security Foundations Symposium (CSF) 2018, pages 218-232, IEEE Computer Society, 2018.

[AT2] EFFICIENT REPRESENTATIONS OF PERMISSION-DEPENDENT TYPES FOR INFORMATION SECURITY ANALYSIS

Information flow analysis of a program can be done statically by reasoning about the potential propagation of information within a program if a formal semantics of the programming language is available. In general, determining whether a sensitive information can be potentially leaked by a program is an undecidable problem, so in practice various approximation methods are used, e.g., via type systems. In this project we will look at using a type system to check whether a program is secure in the sense that it does not leak any information of data that a user marks as sensitive. The programming language used in this study is a simplified and idealised version of Java, that assumes additionally a permission-based access control mechanism similar to that used in Android operating system. A user can specify the sensitivity level of a piece of data (variables, parameters, etc). At the most basic level, this sensitivity level is a boolean value ('true' or 'high', and 'false' or 'low'), but this can be generalise to arbitrary lattice structures. A type in this case can be naively represented as a mapping from permission sets to sensitivity levels. Since in Android one can have a large number of permissions (over 200 of them), which means we have over 2^200 possible permission sets, a naïve representation of a type in this case is not feasible, both in terms of space and time complexity (for performing operations over types). In this project we will investigate a compact representation of permission-dependent types, and how type related operations can be efficiently performed on them. This is feasible due to the fact that applications typically requires only a small number of permissions (typically less than 10).

Reference:

Hongxu Chen, Alwen Tiu, Zhiwu Xu, Yang Liu: *A Permission-Dependent Type System for Secure Information Flow Analysis.* In Proceedings of the 31st IEEE Computer Security Foundations Symposium (CSF) 2018, pages 218-232, IEEE Computer Society, 2018.

[AT3] AUTOMATED REASONING FOR SECURITY PROTOCOL ANALYSIS

A security protocol is a sequence of message exchanges between parties in a network system, usually used to establish certain security targets such as authentication, exchanges of session keys, etc. Security protocols are often described informally, with many underlying assumptions about a possible attacker or intruder left implicit. Often such a description looks deceptively simple (typically a few lines of texts) and it is easy to overlook some implicit assumptions in its analysis. This has been the case for a number of well-known protocols in the literature, where flaws were discovered only when the intruder model was made explicit and precise.

A common, but coarse, abstraction of the capability of an intruder used in security protocol models is the so-called Dolev-Yao model, in which encryption is treated as a black box. In reality, the implementation of encryption functions is often done with operators that satisfy some algebraic properties, e.g., exclusive-Or (which satisfies the properties of an abelian group), homomorphic encryptions, or more advanced techniques based on elliptic curves.

In this project, we are investigating various formal models of intruders, based on formal logic and algebra, and how automated reasoning techniques can be developed for the analysis of security protocols under those models. This in turn will be used to develop a rigorous framework in which the

corretness claims, or the existence of a flaw, of a security protocol can be stated precisely and proven mechanically.

Some potential research topics in this area:

1. Exploring uses of state-of-the-art theorem provers for protocol verification: Some security properties, such as secrecy and authentication, can be encoded as statements in first-order logic. The encoding is such that attacks on a protocol correspond to proofs in first-order logic. Automated reasoning for first-order logic is well-developed and there are now a number of highly effective and mature theorem provers for first-order logic. We will explore various encoding techniques and experiments with using these first-order solvers to help proving security properties of protocols.
2. Techniques for analysing algebraic properties in intruder models: Although encoding protocols in first-order logic may help solve security protocol analysis problems, since first-order logic is semi-decidable, first-order theorem provers may not always produce answers. This subproject will look at designing specific decision procedures for specific intruder models.
3. Scaling up protocol verification: automated protocol verification is essentially a search problem for an infinite state system. Most practical protocol verifiers will use some symbolic representation of states in their search. A natural question is whether this search procedure can be parallelised. This subproject will experiment with a number of state-of-the-art protocol verifiers, to evaluate their inherent support for parallelisation. A more advanced topic would be to modify existing verification algorithms to be more suitable for parallelisation. A collection of benchmarks, specific to protocol verification problems, will be produced.
4. Modelling real-world protocols: This subproject involves surveying security protocols used in many different applied settings. Of particular interests are protocols for low-power devices, eg, IoT devices, implantable medical devices, and protocols in new standards, eg, 5G network protocols. We will model these protocols formally and verify their security properties.

DR QING WANG

Dr. Qing Wang joined the Research School of Computer Science in April 2012. She leads the Database Group at the Research School of Computer Science, ANU. Prior to this, She was an Information Systems Analyst at the PBRF team, Deputy Vice Chancellor's Office (Research), University of Otago, New Zealand. She has over a decade of industry experience in China and New Zealand in the areas of databases, data management and analysis.

Education

Dr. Qing Wang received her Ph.D. (Dr.rer.nat.) in Computer Science (Summa Cum Laude) from Christian-Albrechts-University Kiel, Germany, in 2010. Before that, she received a Master of Information Systems (First Class Honours) from Massey University, New Zealand, a Master of Economics from Jinan University and a Bachelor of Engineering from South China University of Technology, China.

Fellowships, grants & awards

Dr. Qing Wang is a Fellow of the Higher Education Academy (HEA) since December 2015 and obtained ANU Dean's Award for Teaching Excellence in 2015. She received the Best Paper Award at the 33rd International Conference on Conceptual Modeling in 2014, and the Best Paper Award at the IEEE Asia-Pacific Services Computing Conference in 2009. She is a chief investigator in the ARC Discovery Project "Creating the social genome: Advanced techniques for linking dynamic data" together with Prof. Peter Christen and Prof. Erhard Rahm, 2016-2018, and a co-investigator in the project "Advancing data integration: Privacy and semantics for record linkage" funded under the 2015-2016 Australia-Germany Joint Research Cooperation Scheme. She has won Excellence Grant by the Governor of Upper-Austria, Austria, 2013, an Endeavour Research Fellowship Award, Australia, 2012 (declined), Research Grant for Doctoral Candidates and Young Scientists and Academics, German Academic Exchange Service (DAAD), Germany, 2010, Bright Futures Top Achiever Doctoral Scholarship, Tertiary Education Commission, New Zealand, 2006-2009, Todd Foundation Award for Excellence, New Zealand, 2005.

Email: qing.wang@anu.edu.au
Website: http://users.cecs.anu.edu.au/~u5170295/

AVAILABLE PROJECTS WITH DR WANG

[QW1] DYNAMIC KNOWLEDGE TRACING

Knowledge tracing refers to the problem of modelling knowledge of students over time as they interact with a learning system, which have been studied extensively in computer supported education. However, the knowledge tracing problem is inherently difficult as human learning is grounded in the complexity of both the human brain and human knowledge. Recently, deep neural networks have been successfully used to build several knowledge tracing models, including the Deep Knowledge Tracing (DKT) model developed by the researchers from Stanford University (Piech et al., Deep knowledge tracing, NIPS 2015) and Dynamic Key-Value Memory Network (DKVMN) model (Zhang et al., Dynamic key-value memory

networks for knowledge tracing, WWW 2017), which have been shown to be able to predict student performance better than the previous knowledge tracing models.

Benefit for the Student:
Gain a solid understanding of deep learning models, and learn how to implement and apply these techniques in a real-world application setting.

Requirements:
Strong skills in programming (Python) are required. Solid knowledge background in data mining and machine learning are desired.

Background Literature:

- Piech et al., Deep knowledge tracing. In Advances in Neural Information Processing Systems, 2015
- Xiong et al., Going deeper with deep knowledge tracing. In Educational Data Mining, 2016
- Zhang et al., Dynamic key-value memory networks for knowledge tracing. In Proceedings of the 26th International Conference on World Wide Web, 2017

[QW2] DISCOVERING INCONSISTENT DATA IN A DYNAMIC WORLD

Inconsistent data exists everywhere but not always evident. However, using inconsistent data may lead to poor decision making, expensive mistakes, communication chaos, etc. There is an increasing industry-driven demand for tools that can efficiently identify and reduce inconsistent data. One important aspect in developing such tools is to find efficient and effective ways of capturing the structure and semantics of data which will enable the detection of inconsistent data.

The project aims to analyse and mine patterns of inconsistencies occurring among data, and then develop efficient algorithms to capture and resolve inconsistent data based on such patterns.

Benefit for the Student:
Gain a solid understanding of data mining and database technologies, and learn how to use these techniques to analyse data and improve data quality.

Requirements:
Strong skills in software development (Java or Python) are required. Solid knowledge background in data mining and relational database are desired.

Australian National University

DR. ZHENCHANG XING

Dr. Zhenchang Xing is a Senior Lecturer in the Research School of Computer Science, Australian National University. Previously, he was an Assistant Professor in the School of Computer Science and Engineering, Nanyang Technological University, Singapore, from 2012-2016. Before joining NTU, Dr. Xing was a Lee Kuan Yew Research Fellow in the School of Computing, National University of Singapore from 2009-2012. Dr. Xing obtained his Master of Engineering from Nankai University, Tianjin, China in 2001, and his PhD degree from the University of Alberta, Canada in 2008.

Dr. Xing's main research area is software engineering, applied data analytics, and human-computer interaction. He has published expertise in traditional areas of software engineering such as software differencing, clone analysis and feature location. In his most recent work, Dr. Xing has focused on designing domain-specific data-mining techniques and recommendation systems for the timely and serendipitous discovery of software engineering information on the internet.

Dr. Xing has over 80 publications in refereed journals and conference proceedings. His research work has been published in top software engineering venues such as ICSE, FSE, ASE, ICSME, SANER, MSR, TSE, EMSE. Four of his research papers have won Best Paper awards in ICSM2006, ICSM2011, SANER2016, ASE2016 ToolDemo. Three others have won nominations for Best Paper awards in ASE2005, ICSM2014, and SANER2015. Dr. Xing regularly referees papers and serves on the program committees of the top software engineering conferences (e.g., ICSE, ASE, ICSME). He has won AUD$1,000,000 in industry and government funding and has industrial collaborations with Rolls-Royce, Insigma Hengtian, and Bangsun Technology.

Dr. Xing has graduated six phd students, two of which are now faculty members in universities, two are postdoctural fellow, one in MSRA Suzhou, and one in Tencent AI lab. He currently supervise 3 PhD students, 7 Master students and several undergraduate Honours projects. Dr. Xing and his research team has been actively working on the following research projects:

- Software Engineering: Semantic clone analysis, Software graph embedding and reasoning, Code search and generation
- Human-Computer Interaction: Policy assurance bot, UI design critics and generation, Domain-specific dialogue systems
- Knowledge Graph: Entity and relation extraction in software text, Software weaknesses and vulnerabilities representation learning
- Deep Learning in SE: Question answering and chatbot, Software development knowledge and behavior tracing, Multi-modal (e.g, code, text, image) embedding and retrieval

Email: Zhenchang.Xing@anu.edu.au

Website: https://cecs.anu.edu.au/people/zhenchang-xing

Australian National University

AVAILABLE PROJECTS WITH DR ZHENCHANG XING

[ZX1] SOFTWARE-SPECIFIC KNOWLEDGE GRAPH CONSTRUCTION AND APPLICATIONS

Knowledge graph is a structured knowledge of domain-specific entities and relations in an application domain. It is the foundation for next-generation of knowledge search system, as opposed to the current information retrieval systems. It is also important for building knowledge-aware AI systems. In this project, we will devise domain-specific natural language processing techniques to extract from software text software-specific entities (e.g., APIs, API caveats, programming tasks, software weaknesses) and the rich relations among these software entities, and hence construct software-specific knowledge graph. Based on the constructed knowledge graph, we will investigate knowledge-aware applications such as task-centric knowledge search, software vulnerability analysis, zero-shot API misuse detection.

[ZX2] DATA-DRIVEN RAPID CREATION, ASSESSMENT AND PROTOTYPING OF UI DESIGNS

A software application's Graphical User Interface (GUI) should not only provide a working interface for users to interact with the technology, but also create an intuitive, secure and trustworthy experience of using the technology. A well-designed User Interface (UI) is crucial for the acceptance of a technology by ordinary users and for preventing human error. In this project, we will develop computer vision and natural language processing methods to distil the crowd-scale knowledge of UI designs and implementations from large numbers of existing applications, and exploits the distilled knowledge as an "expert" who knows a vast variety of UI designs and implementations to assist designers and developers in searching for design inspirations, criticizing design quality and alternatives, and expediting the cycle of UI prototyping.

[ZX3] RECOGNIZING ARTIFICIAL ACTIONS IN KNOWLEDGE WORK

When a knowledge work (e.g., software developer, UI designer) interact with a computer in his/her work, the computer reacts to human instructions by visualizing a sequence of artificial actions on the computer screen, such as move cursor or mouse pointer, display entered text, highlight selected text, or switch application windows. These artificial actions can be recorded as a time series of screenshots (i.e., screencasts) for demonstrating the process of completing a task in computing environment. Recognizing artificial actions in such tasks is the foundation for indexing, searching and navigating the task-specific knowledge in knowledge work. In this project, we will construct large-scale datasets of knowledge-work actions and investigate the problem and solution space recognizing artificial actions in screencasts by computer vision techniques. This work will build the foundation for developing AI pair programmers.