Wrapping practical problems into a machine learning framework: using water pipe failure prediction as a case study

Jianlong Zhou*

DATA61, CSIRO,
Level 5, 13 Garden Street,
Eveleigh, NSW 2015, Australia
Email: Jianlong.Zhou@data61.csiro.au
*Corresponding author

Jinjun Sun[1]

Red Planet of Qantas Loyalty,
10 Bourke Road,
Mascot, NSW 2020, Australia
Email: jsunster@gmail.com

Yang Wang and Fang Chen

DATA61, CSIRO,
Level 5, 13 Garden Street,
Eveleigh, NSW 2015, Australia
Email: Yang.Wang@data61.csiro.au
Email: Fang.Chen@data61.csiro.au

Abstract: Despite the recognised value of machine learning (ML) techniques and high expectation of applying ML techniques within various applications, users often find it difficult to effectively apply ML techniques in practice because of complicated interfaces between ML algorithms and users. This paper presents a work flow of wrapping practical problems into an ML framework. The water pipe failure prediction is used as a case study to show that the applying process can be divided into various steps: obtain domain data, interview with domain experts, clean/pre-process and preview original domain data, extract ML features, set up ML models, explain ML results and make decisions, as well as make feedback to the system based on decision making. In this process, domain experts and ML developers need to collaborate closely in order to make this workflow more effective.

Keywords: machine learning; practical problems; human-computer interaction; water pipe failure prediction.

Reference to this paper should be made as follows: Zhou, J., Sun, J., Wang, Y. and Chen, F. (2017) 'Wrapping practical problems into a machine learning framework: using water pipe failure prediction as a case study', *Int. J. Intelligent Systems Technologies and Applications*, Vol. 16, No. 3, pp.191–207.

Biographical notes: Jianlong Zhou is a Senior Research Scientist of Analytics Group in DATA61, CSIRO. He got a PhD in Computer Science from the University of Sydney, Australia. His research interests include transparent machine learning, human-computer interaction, cognitive computing, volume visualisation, spatial augmented reality and related applications.

Jinjun Sun is currently a Data Scientist in Red Planet of Qantas Loyalty, Australia. He was a Research Engineer of Machine Learning Research Group in NICTA before this role. He received his PhD in Physics from Macquarie University, Australia. His research interests include systematic architectures targeting at big data challenges.

Yang Wang is a Principal Research Scientist of Analytics Group in DATA61, CSIRO. He received his PhD in Computer Science from the National University of Singapore in 2004. His research interests include machine learning and information fusion techniques and their applications to intelligent infrastructure, cognitive and emotive computing.

Fang Chen is a Senior Principal Research Scientist of Analytics Group in DATA61, CSIRO. She holds a PhD in Signal and Information Processing, an MSc and BSc in Telecommunications and Electronic Systems respectively, and an MBA. Her research interests are behaviour analytics, machine learning, and pattern recognition in human and system performance prediction and evaluation. She has done extensive work on human-machine interaction and cognitive load modelling. She pioneered theoretical framework of measuring cognitive load through multimodal human behaviour, and provided much of empirical evidence on using human behaviour signals, and physiological responses to measure and monitor cognitive load.

1 Introduction

1.1 Problem description

With the rapid increasing of data from various fields such as biology, finance, medicine, and society, users are looking to integrate their 'Big Data' and advanced analytics into business operations in order to become more analytics-driven in their decision making. Such decisions can be integrated in various real world scenarios (e.g., remote collaboration in fixing problems (Huang and Alem, 2013). Much of machine learning (ML) research is inspired by such expectations. Various ML algorithms offer a large number of useful ways to approach those problems that otherwise require cumbersome manual solution. Despite the recognised value of ML techniques and high expectation of applying ML techniques within various applications, users often find it difficult to effectively apply ML techniques in practice because of complicated interfaces between ML algorithms and users, such as complex parameter settings and intermediate decisions. Because of these complexities, it is very hard to see ML as a general solution for widespread applications. As a result, ML is regarded as a large bag of tricks grasped by ML experts instead of a universal tool for non-experts. It is one of challenging tasks of wrapping practical problems into an ML framework for both domain experts and ML developers. Therefore, the investigation of workflow of applying ML to practical

problems benefits both domains and ML research fields and helps make ML transparent in practical applications.

Using water pipe failure prediction as an example, water supply networks constitute one of the most crucial and valuable urban assets. The combination of growing populations and aging pipe networks requires water utilities to develop advanced risk management strategies in order to maintain their distribution systems in a financially viable way (Li et al., 2014). Especially for critical water mains (generally >300 mm in diameter), defining based on the network location (for example, a single trunk line connecting distribution areas or under a major road) or size which infers impact potential, failure of them typically bring severe consequences due to service interruptions and negative economic and social impacts, such as flooding and traffic disruption (Li et al., 2014). The financial and social costs of reactive repairs in such scenarios amount to more than one billion dollars annually in Australia alone. For instance, over the past 10 years, Sydney Water has spent around $3.5 million each year on reactive critical water main repairs (Whiffin et al., 2013). Currently the critical main network in Sydney Water consists of 4700 km, with an average pipe age of 50 years over a geographical area of 12,700 km^2 (Whiffin et al., 2013). From an asset management perspective there are two goals for critical mains management (Whiffin et al., 2013):

- minimise unexpected critical main failure by prioritising timely renewals
- avoid replacing a pipe too early before the end of its economic life.

If high-risk pipes can be identified before a failure occurs, it is likely that repairs can be completed with minimal service interruption, water loss and negative reputational and community impacts. Identification of an accurate predictor measure that indicates imminent failure will allow utility companies to take actions to mitigate the failure for a lower cost than repairing a full-scale failure. This will contribute to extending the service life of pipes that are still in good condition and allow running the mains to an acceptable defined risk limit (Whiffin et al., 2013). As the average age of the network increases, pipes are easily failed with the decrease of pipe strength. It will become more important to accurately predict the risks of pipe failure and provide the right level of pipe maintenance and renewal at the right time, according to risks associated with each pipe. Such pipe management benefits utility authorities in following ways:

- increase customer satisfaction by reducing critical main failures and service disruption
- improve the way of pipe management by:
    - doing preventative maintenance rather than reactive repair
    - providing better understanding of the risk factors in each area
    - performing target monitoring programs to collect the right data
    - finding more poor condition pipes with the same level of assessment activity
- avoiding replacing pipes that still have remaining life.

1.2 Information available for problems

The management authorities of water pipes collect various data on water mains, which include geographical information of pipes, failure history, and attributes of pipes. Specifically, these data mainly include:

- geographical and environmental factors: location of pipes, soil types, weather conditions and transportation conditions around the pipe area
- failure history: failure type, failure date, failure times
- pipe attributes: laid year, size (diameter), length, materials (typically include cement mortar lined cast iron, ductile iron or steel, asbestos cement, or plastic), coating types
- internal pressure.

Other information such as wall thickness of pipes and cement lining thickness, depth of main, photos of pipe sample are also collected.

1.3 Goals of water pipe failure prediction

ML is becoming a viable technique to quantify probabilities in practical applications including water pipe failure prediction. ML techniques are expected to improve accuracy of risk analysis (e.g., pipe failure prediction) and reduce maintenance cost of pipes (e.g., better prioritisation criteria). The ultimate goals of water pipe failure prediction utilising ML techniques include: 1) provide an assessment of current condition; and 2) predict likelihood of pipe failure for given time period. The challenges for water pipe failure prediction using ML techniques include two aspects:

- data assimilation problem:
  - How to assemble relevant information into a composite understanding of pipe condition
  - incomplete/missing data.
- prediction problem:
  - How to translate the understanding into a prediction of pipe failure
  - when and where will a pipe fail under uncertainties.

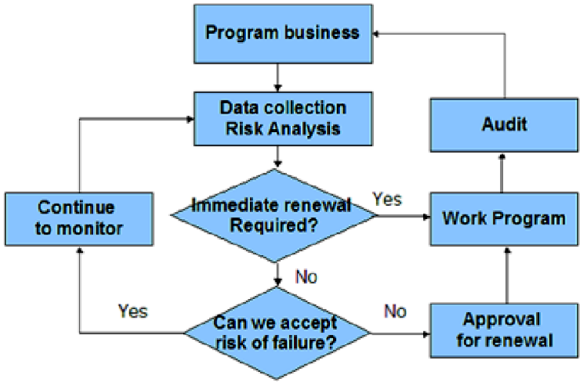
2 Conventional workflow on water pipe failure management

Current tools for water pipe failure management are risk-based approaches. For example, Sydney Water sets up a risk matrix based on the quantitative calculations of likelihood of failure and consequence (economic) of failure (Kane et al., 2014). The risk categorisation is based on best available quantitative information from actual field condition assessment and cost data or best quantitative estimates by other means unless field data is unavailable. Figure 1 illustrates the typical critical water main decision framework utilised (Kane et al., 2014). This framework is a formal decision process that identifies, priorities, and recommends critical water mains for condition assessment and/or renewal

based on a quantified risk level of the assets. The process includes an initial risk assessment based on available information, prioritisation, and progressive refinement of the risk assessment through condition assessments and analysis of failure history (Kane et al., 2014).

From this decision framework, it is obvious that risk matrix plays central roles in the water pipe failure management. The risk matrix highly depends on likelihood of failures of pipes, which is based on different factors such as pipe age, pipe material, past failure history, and engineering judgement.

Figure 1 Dynamic decision support in water pipe failure management (see online version for colours)



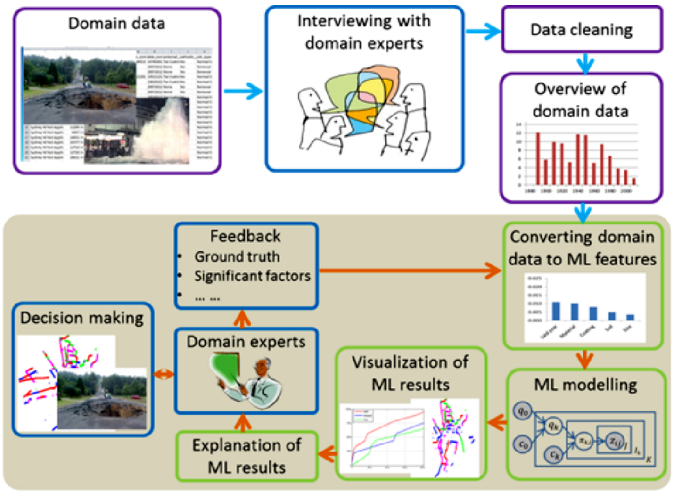
Source: Kane et al. (2014)

3 Workflow of applying ML to water pipe failure prediction

As mentioned, the likelihood of failures of pipes plays significant roles in water pipe failure management. Furthermore, ML techniques are powerful in learning probabilities based on historical data. However, because of complexities of both water pipe failure problems and ML techniques, it is challenging to phrase the water pipe failures as an ML framework. Figure 2 illustrates the workflow of phrasing water pipe failures as an ML problem in our practice. In this workflow, original domain data are firstly collected from customers. Then the interviewing with domain experts is arranged to learn details on water pipes, such as what factors affect pipe failures from the domain expert's view, how domain experts predict pipe failures in their routine work. After this stage, because of missing information or other reasons, the original domain data are cleaned in order to be processed easily by future stages, such as removing records with missing information or inputting default values in records with missing information. After cleaning the data, we, as ML technique developers, try to get an overview of domain data and learn some patterns in the data. Based on the overview of the cleaned data, various data features are derived and ML models are developed. To allow users easily understand ML results, visualisation of ML results are then presented. The results need to be explained to users using domain knowledge. According to the explanation of ML results, decisions are made to practice domain actions such as digging out and replacing high risk pipes. From the practice actions, significant information can be got such as whether pipes predicted as high risk ones are confirmed or violated from actual digging. The information can be

used as feedback to the pipeline to improve effectiveness of ML analysis, such as feature definition and ML modelling. The following sections present details on each stage of this workflow in our practice.

Figure 2 Workflow of phrasing water pipe failures as an ML framework (see online version for colours)



4 Domain data and interviewing with domain experts

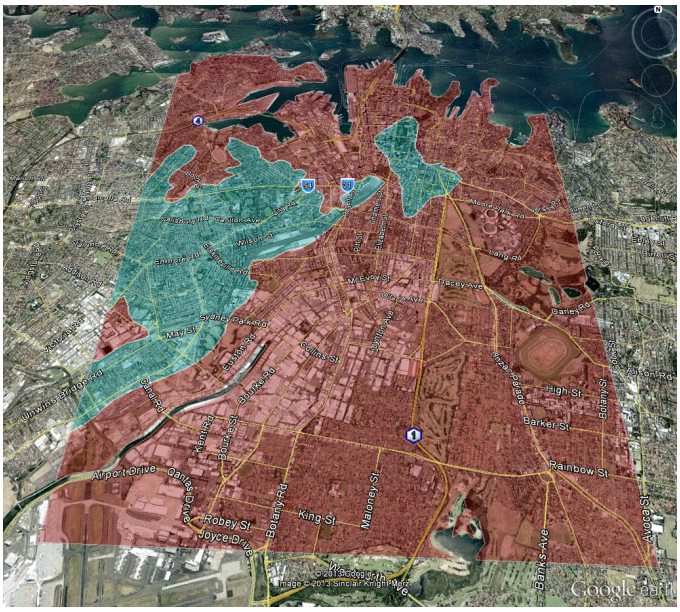
The original domain data got from customers were usually various spread sheets recording geographical information, failure history and physical attributes of pipes. Other data (e.g., images) were also provided to further understand pipe failures, such as types of failures and consequences caused by pipe failures.

To have better understanding of domain datasets and exact goals that customers want to get from data, interviewing with domain experts was actively performed. Domain experts explained their considerations of factors that affect pipe failures. For example, different soil types (see an example in Figure 3) may affect pipe life expectancies. Pipes in casted iron are easily got corrosions. Pipes within busy traffic areas may have high failure rates than in other areas. The views from domain experts help to define and evaluate data features which are fed to ML models.

Regarding the water pipe failure prediction, domain experts have various questions such as (Kane et al., 2014):

- How, when, and where will pipes fail within the entire network?
- How do we assess the condition of the pipe cost effectively?
- How do we calculate pipe deterioration rates accurately with respect to the pipe environment?
- What is the time-dependent probability of the pipe failure along the pipeline?
- How do we transfer the new knowledge to the industry for optimal pipe management?

Figure 3 Different soil types encoded by colours in a region (see online version for colours)



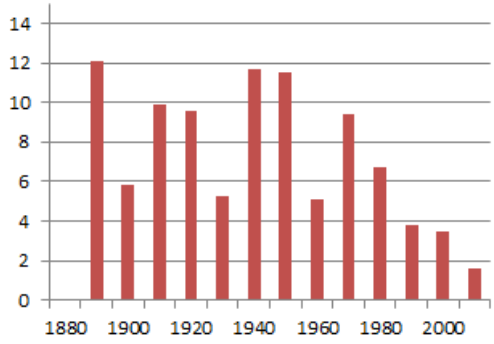
5 Data cleaning and overview of domain data

The domain data collected from customers are usually not able to be used directly in computational analyses programs because of incompleteness, mismatching, noises or other reasons. Therefore, pre-processing operations are conducted in order to make the data format ready for convenient operations in later stages. For instance, the water pipe data usually include main and failure records which indicate the laid pipes and failure pipes respectively. To make the data consistently between main and failure records, data matching is one of significant operations conducted during pre-processing step. It is usually conducted based on pipe ID or location of pipes.

When finishing cleaning data, data summary and overview need to be conducted in order to better understand failure patterns. The data summary includes record summaries and visualisations on summaries. For instance, for the main records, the summary includes total number and total length of laid pipes as well as percentage of laid pipes that are still working. For the failure records, the summary includes total number and total length of failure pipes as well as the number of failure pipes that can be matched back to the main pipe records.

The summary may also be visualised in various forms and on maps if geographic location information is available. For example, an overview of failure rate by installed year as shown in Figure 4 is helpful to understand patterns of pipe failures, where the vertical axis is the failure rate measured by failure numbers over 10 years per 100 km, the horizontal axis is the installation year. Similarly, the overview of failure rate by other pipe attributes (e.g., size, materials) also help understand how those attributes affect pipe failures.

Figure 4 Overview of failure rate by installed year in region A (see online version for colours)



6 Converting domain data to ML features

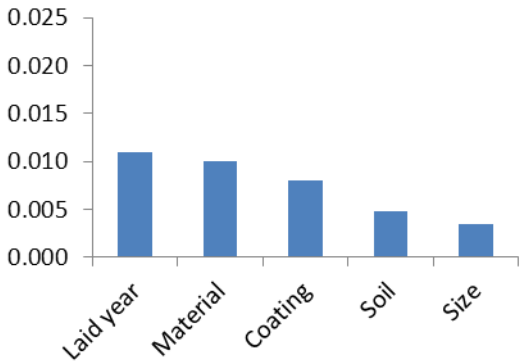
6.1 Factor analysis

As mentioned, there are lots of attributes/factors that may affect water pipe failures. However, not all factors have same contributions to pipe failures. Therefore, the estimation of importance of factors is helpful in the selection of data features for ML modelling. Given a factor x (e.g., material), the information gain (IG) on pipe failure y ($y = 1$ for a failed pipe, otherwise $y = 0$) can be determined by:

$$IG(x, y) = H(x) - H(x \mid y), \tag{1}$$

where $H()$ and $H(|)$ represent information entropy and conditional entropy respectively. For example, as illustrated in Figure 5, the vertical axis represents the information gain of various factors. This figure shows that some factors such as laid year, material and coating significantly affect pipe failures, while other factors such as size have smaller contributions to pipe failures. Based on this evaluation, factors that significantly contribute to pipe failures are selected for further analysis in the coming stages.

Figure 5 Factor analysis in water pipe failure prediction in region A (see online version for colours)



6.2 Feature definitions

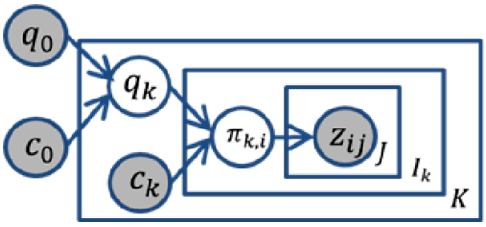
Feature definition is an important step for ML analysis. It directly affects the effectiveness of ML analysis. In water pipe failure prediction, data features used for ML analysis are defined based on previous project experience and domain experts' feedback. The main features/attributes used in water pipe failure prediction include installed years, pipe diameter, pipe material, coating surface, soil and traffic in the area of pipes.

7 ML modelling

The probability of a pipe failure itself is a random variable dependent on a set of pipe physical attributes (e.g., age) as well as related environmental conditions (e.g., soil type). Various parametric or semi-parametric models such as the Cox model, Markov model, and Weibull model (Ibrahim et al., 2005) have been developed for water pipe failure analysis. However, parametric models are usually limited by their fixed model structure based on a priori assumptions on the data behaviour and their inability to adaptively adjust the model to the complexity of the problem (Li et al., 2014). To address these limitations, the use of Bayesian nonparametric learning is proposed to predict water pipe condition (Li et al., 2014). Historical water pipe data can be incorporated and the model can grow to accommodate future data as necessary. Our work particularly investigated hierarchical beta process (HBP) (Li et al., 2014) for sparse incident data to develop an efficient approximate inference algorithm. The method can be used to predict the failure rate of each individual pipe more accurately by capturing specific failure patterns of different water-pipe groups. Compared to existing statistical prediction methods, HBP offers a more flexible model structure to accommodate the volume and diversity of historical data and are less sensitive to the effects of various noisy factors. It is also possible to incorporate spatial relationships among neighbouring pipes to better predict infrequent failures (Whiffin et al., 2013).

In the HBP model as shown in Figure 6, pipes are divided into K groups based on laid years and modelled as a HBP. In the top level, hyper parameters, which control across all groups of pipes by a beta distribution, are set manually according to domain experts' experience. Then, the mean failure rate (q_k) in each group can be generated from the distribution. In the middle level, the mean failure rate ($\pi_{k,i}$) of each pipe asset is generated through another beta distribution with q_k as parameter. In the bottom level, the actual failures $z_{i,j}$ are generated from a Bernoulli process year by year using $\pi_{k,i}$.

Figure 6 The diagram of HBP model (see online version for colours)

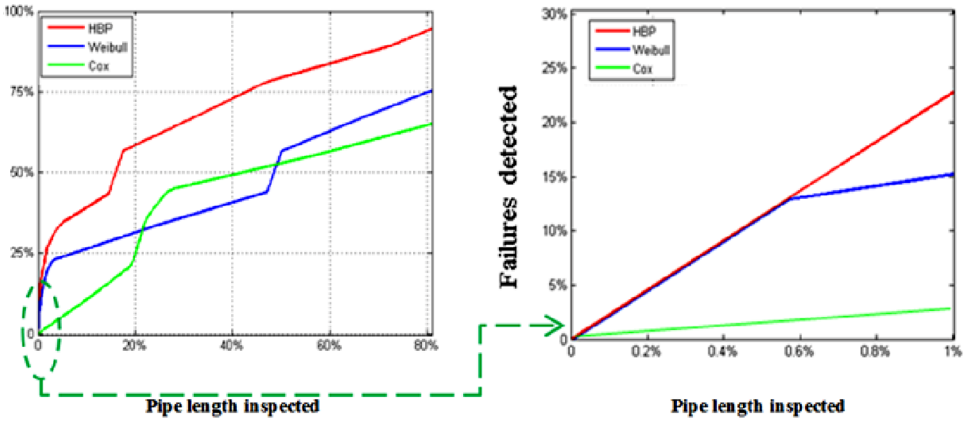


8 Visualisation and explanation of ML results

8.1 Visualisation of ML results

The analysis results from ML modelling are usually in abstract forms such as probabilities. To transform the abstract results into easily understandable representations, various visualisations of ML results are conducted at this stage. For the water pipe failure prediction, because pipes are associated with geographic locations besides failure risks learning from HBP model, pipes are visualised on maps with colour encoded failure risks. This visualisation is called as risk map. From the risk map, users can easily get answers on pipe failures such as where are the most risky pipes located from the visualisation.

Figure 7 Results of pipe failure prediction using different models (see online version for colours)



8.2 Comparison of ML performance

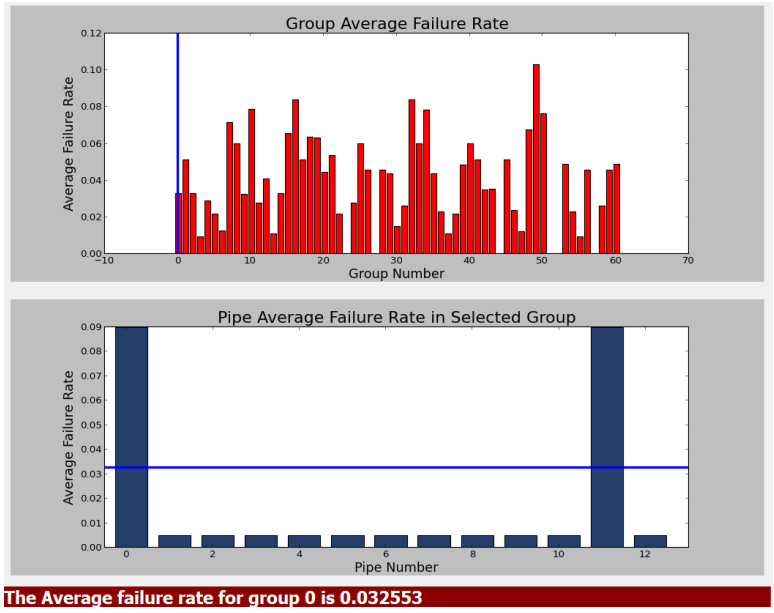
Besides the visualisation of ML results, ML results also need to be explained to users based on domain knowledge in order to let end users trust ML results and utilise ML results for include: why the proposed approach is better than other approaches, why the ML results are believable, etc. For example, as illustrated in Figure 7 (Li et al., 2014), the x-axis represents the length of condition assessed water pipes starting at the top of the list, and the y-axis represents the percentage of actual failures detected from those inspections. The right figure is the enlarged version of the circled region in the left diagram for the first 1% of all critical water mains. This is because that the budget and resources allocable for pipe condition assessment are usually limited, each year only a small fraction of the critical water mains can be physically inspected, typically around 1% of the whole network length. The comparison shows that nonparametric approach (HBP) outperformed traditional model of Weibull (Ibrahim et al., 2005) and Cox.

9 Explanation of ML process

For a domain expert who may not have expertise in ML or programming, an ML algorithm acts as a 'black-box', where the user defines parameters and input data for the 'black-box' and gets output from its execution. This 'black-box' approach has obvious drawbacks: it is difficult for the user to understand the complicated ML models, such as what is going on inside the ML models and how to accomplish the learning problem. As a result, the user is uncertain about the usefulness of ML results and this affects the effectiveness of ML methods. Therefore, ML process needs to be explained in an appropriate way in order to make it easily understandable (Zhou and Chen, 2015).

Research found that providing support for explaining 'run-time' behaviour had a significantly positive impact on both end users' effectiveness of debugging and their attitude toward the system (Kulesza et al., 2010). Providing explanations has been shown to be effective in other domains such as decision making (Dzindolet et al., 2003) and recommender systems (Herlocker et al., 2000) where providing explanations led to increased trust and acceptance. Commercial applications, such as Amazon's product recommender system or Pandora's music recommender now integrate explanations into their interfaces (Lim et al., 2009). It was also found that why and why not explanations lead to improved user understanding, trust, perception, and performance more than having no explanations (Stumpf et al., 2007). ML models are also explained by learning an interpretable model locally around the prediction, and presenting representative individual predictions and their explanations in a non-redundant way (Ribeiro et al., 2016). Krause et al. (2016) employed interactive visual analytics to help users understand how features affect the prediction overall by providing interactive partial dependence diagnostics.

Figure 8 The real-time status update of ML process is presented to users with interactive graphs and animations (see online version for colours)



This study uses an approach of revealing internal ML states to make end users more convincing on ML results from HBP. As shown in Figure 8, the top chart presents the status update of q_k and the bottom chart presents the status update of $\pi_{k,i}$. During ML process, the charts are dynamically changed to reveal the internal real-time status update. To interact, users can point to any (interact detail) q_k in the top chart and the corresponding $\pi_{k,i}$ is presented accordingly in the bottom chart. Compared with directly presenting the final prediction of failure rate $\pi_{k,i}$, the presentation of q_k and $\pi_{k,i}$ allows users learn how the prediction of failure rate of each pipe is approached. As a result, users' convincingness on predictions is increased. With the help of a user study, we found that revealing of the internal states of ML process can help to improve easiness of understanding the data analysis process, make real-time status update more meaningful, and make ML results more convincing.

10 Decision making and feedback

After ML processes are explained to domain experts, decisions are made by domain experts to take actual actions. For example, in water pipe failure management, domain experts make decisions to dig out the most risky pipes and make condition assessment. This is the step where ML technologies have actual impact on real-world.

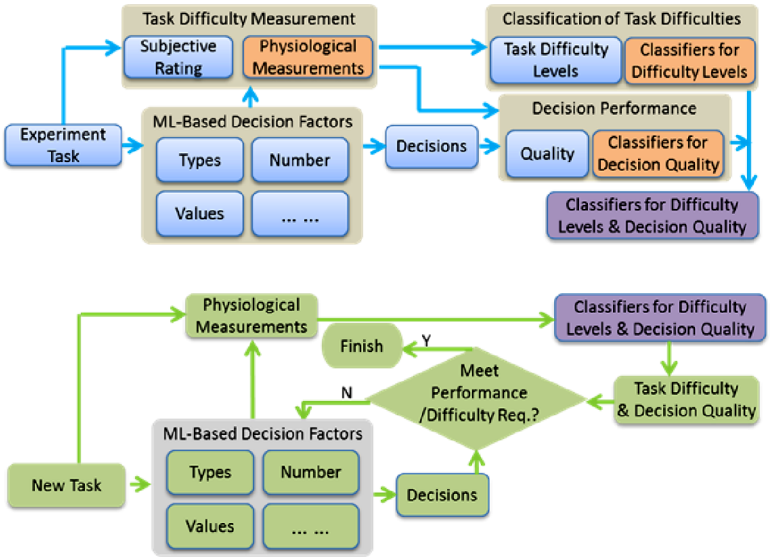
Because ML results are usually abstract and complicated, how to exploit ML results effectively in decision making is challenging. To learn what factors of ML results affect decision making and how to measure decision performance with different ML results, we proposed that decision making can be measured in order to let users perceive decision qualities and decision difficulty levels in real-time (Zhou et al., 2015, 2016). In the proposed framework of adaptive measurable decision-making (see Figure 9), when an experiment task with certain decision factors (e.g., ML-based decision factors) is exposed to users for decision making, task difficulty is measured at the same time with subjective ratings and physiological measurements (e.g., GSR, eye-tracker). After the user makes decisions, the decision performance is evaluated with the user's choice and physiological measurements. The measured information is then analysed and classifiers for decision quality and task difficulty are derived.

When a new task is coming, users' workload during decision making is recorded using physiological measurements in real-time. The measurements are sent to classifiers for difficulty levels and decision quality learned in the experiment task stage. The task difficulty level and decision quality from classifiers are exported to users. If users are satisfied with the decision performance and decision itself, then the decision making process is finished. Otherwise, decision factors are refined (e.g., increase/decrease number of certain decision factors) based on the analysis results at the experiment task stage to continue a new decision-making session. For example, if the decision difficulty level derived from measurements is low, more certain decision factors may be included in order to get higher quality decisions. This process is iteratively performed until decisions meet performance and difficulty requirements from users. Such framework allows users refine decisions adaptively and interact with system more efficiently in human-computer interaction (HCI) systems. Zhou et al.'s (2015) study found that various aspects

of ML results (e.g., type, numbers, and values) affected decision difficulty levels and decision qualities. For example, the number of ML-based decision factors significantly affected easiness of decision-making process, and more number of ML-based decision factors made the decision-making process more difficult.

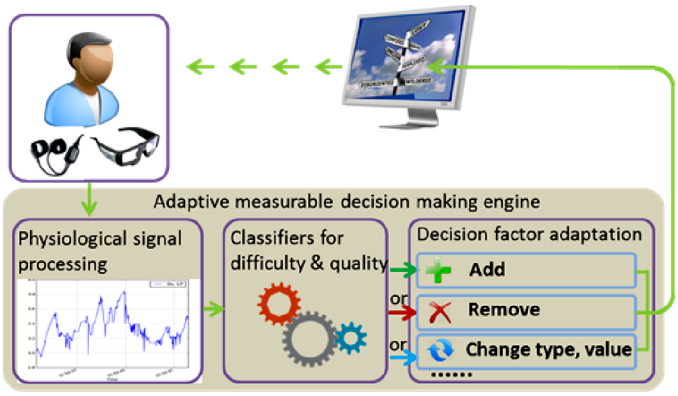
Figure 10 illustrates the loop of using adaptive measurable decision-making. In this loop, the adaptive measurable decision-making engine is mainly composed of the physiological signal processing component, classifiers for decision difficulty and decision quality, as well as decision factor adaptation. Raw physiological signals from the user are input into the adaptive measurable decision-making engine. The decision difficulty levels and decision qualities are derived from signals. If the user is not satisfied with the decision difficulty levels and decision qualities, decision factors are refined (e.g., add/remove decision factors, change types/values of decision factors) and a new decision process is performed based on the updated decision factors until the user is satisfied with the decision performance.

Figure 9 Framework of adaptive measurable decision-making (see online version for colours)



Based on this decision-making study (Zhou et al., 2015), water pipe failure management can utilise various factors (e.g., likelihood of failure from ML analysis, economic factors) adaptively to refine and get high quality decisions. As mentioned in Section 2, decision making in conventional water pipe failure management conducts progressive refinement of risk assessment. In ML-based water pipe failure prediction, this action provides domain experts the opportunity to check whether the ML prediction is confirmed to be true or not. As a result, it provides a ground truth for the prediction. Furthermore, domain experts may also analyse what factors more significantly affect ML performance from the confirmation of ML predictions. Such information can be used as feedback for the ML pipeline to improve ML models. For example, using feedback to choose more meaningful features and modulate parameters for ML models.

Figure 10 Diagram of the use of adaptive measurable decision-making in an application (see online version for colours)



11 Discussions

We presented a workflow of how a practical problem was wrapped as an ML framework. We used water pipe failure prediction as a successful example to show how to make ML techniques useable in solving practical problems.

Regarding our proposed workflow, the interaction with the ML system can be modelled as an 'explanatory debugging' perspective as illustrated in Figure 11. From this perspective, in order to make ML transparent and useable to end users, ML results need to be explained to end users based on both domain knowledge and ML theories to let end users understand and trust ML results. End users then make decisions based on the explanation. From checking of decision-making results, end users give feedback to the interaction loop to control the further analysis process such as refining data features or changing ML parameters. Explanation and Feedback play significant roles in this loop to make ML analysis more effective.

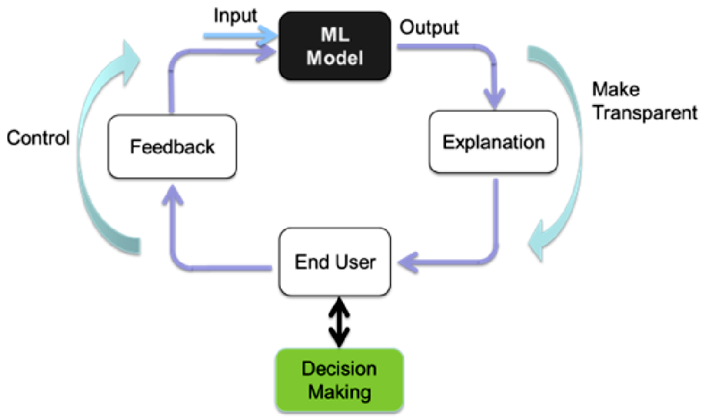
Types of corrective feedback end users would like to give to ML systems include (Stumpf et al., 2007): reweighting features, creating new features (such as by combining features or creating features based on relational information), changes to algorithms. Domain knowledge can also be encoded into the feedback in order to improve ML models. Experiments found that feature reweighting was difficult for end users because of its insensitivity to changes. User co-training framework is another feedback approach which treats user's feedback as if it were a second classifier (Stumpf et al., 2009). Feature rather than instance labelling shows promising results for user feedback in locally-weighted logistic regression (Wong et al., 2011). To incorporate domain knowledge into the feedback pipeline, a module of converting domain knowledge into features usable by ML models is necessary.

In our case study, we made explanations of ML system by revealing internal states interactively to users in order to improve easiness of understanding the data analysis process, make real-time status update more meaningful, and make ML results more convincing. However, further research is expected to explain ML results meaningfully in order to help domain experts make decisions confidently. Furthermore, feedback in the interaction loop needs to be investigated by considering application background. For

example, it is still challenging that what kind of information is useful as feedback from both end users and ML systems for improving effectiveness of ML processes. Therefore, domain experts and ML developers need to collaborate closely to analyse feedback for effective controlling of ML systems.

In summary, from the case study, it was concluded that in order to phrase a practical problem into an ML framework, developers of ML techniques need to start with the cooperation with domain experts closely to understand the problems to be investigated. This affects ML experts' decisions such as what data features to be extracted and what kind of ML models to be used in data analyses. ML results also need to be explained meaningfully before decision making. The feedback based on decision action results is also significant for improving effectiveness of ML models. This case study also demonstrated that an applicable ML analysis process is not a completely automatic process but an interactive pipeline in which domain knowledge and feedback from end users make the ML analysis more understandable and controllable.

Figure 11 An 'explanatory debugging' perspective of end user interaction with an ML system (see online version for colours)



12 Conclusions and future work

This paper proposed a workflow of phrasing practical problems as an ML framework. We used water pipe failure prediction as a case study to show the steps of wrapping practical problems into various stages of an ML pipeline. The workflow showed that applying ML to a practical problem such as water pipe failure prediction can be divided into various steps: obtain domain data, interview with domain experts, clean/pre-process and preview original domain data, extract ML features, set up ML models, explain ML results and make decisions, as well as make feedback to the system based on decision making. Domain experts and ML developers need to cooperate closely in order to make this workflow more effective.

Our future work will focus more on the explanation of ML results and setting up feedback to the ML analysis process. The challenges for the explanation of ML results lie in the selection of objects to be explained and methods to be used for the explanation. The explanation of ML results has close relations with various fields, such as human-computer interaction, domain knowledge, and ML theories. We will focus on the analysis

of how these different aspects contribute to the explanation of ML results in order to make ML process understandable. As a result, end users trust ML results and make domain decisions based on ML results. Furthermore, in practice, ML analysis is an interactive process. Therefore, feedback in the analysis pipeline can help to improve effectiveness of ML analysis. However, it is challenging to set up an effective feedback mechanism to refine the ML analysis process. The main questions lie in what data are used as feedback and how to combine feedback effectively into the interaction loop.

References

Dzindolet, M.T., Peterson, S.A., Pomranky, R.A., Pierce, L.G. and Beck, H.P. (2003) 'The role of trust in automation reliance', *Int. J. Hum.-Comput. Stud.*, Vol. 58, No. 6, June, pp.697–718.

Herlocker, J.L., Konstan, J.A. and Riedl, J. (2000) 'Explaining collaborative filtering recommendations', *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work, CSCW'00*, ACM, New York, NY, USA, pp.241–250.

Huang, W. and Alem, L. (2013) 'Handsinair: a wearable system for remote collaboration on physical tasks', *Proceedings of the 2013 Conference on Computer Supported Cooperative Work Companion (CSCW 2013)*, San Antonio, Texas, USA, pp.153–156.

Ibrahim, J.G., Chen, M-H. and Sinha, D. (2005) 'Bayesian survival analysis', *Encyclopedia of Biostatistics*, John Wiley & Sons, Ltd., New Jersey, USA.

Kane, G., Zhang, D., Lynch, D. and Bendeli, M. (2014) 'Sydney Water's critical water main strategy and implementation - a quantitative, triple-bottom line approach to risk based asset management', *Water Asset Management International*, Vol. 10, No. 1, pp.19–24.

Krause, J., Perer, A. and Ng, K. (2016) 'Interacting with predictions: visual inspection of black-box machine learning models', *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, CHI '16*, San Jose, CA, USA, pp.5686–5697.

Kulesza, T., Stumpf, S., Burnett, M., Wong, W-K., Riche, Y., Moore, T., Oberst, I., Shinsel, A. and McIntosh, K. (2010) 'Explanatory debugging: supporting end-user debugging of machine learned programs', *2010 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, Leganés-Madrid, Spain, pp.41–48.

Li, Z., Zhang, B., Wang, Y., Chen, F., Taib, R., Whiffin, V. and Wang, Y. (2014) 'Water pipe condition assessment: a hierarchical beta process approach for sparse incident data', *Machine Learning*, Vol. 95, No. 1, pp.11–26.

Lim, B.Y., Dey, A.K. and Avrahami, D. (2009) 'Why and why not explanations improve the intelligibility of context-aware intelligent systems', *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '09*, ACM, New York, NY, USA, pp.2119–2128.

Ribeiro, M.T., Singh, S. and Guestrin, C. (2016) ''Why should I trust you?': explaining the predictions of any classifier', *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD2016)*, San Francisco, CA, USA, pp.1135–1144.

Stumpf, S., Rajaram, V., Li, L., Burnett, M., Dietterich, T., Sullivan, E., Drummond, R. and Herlocker, J. (2007) 'Toward harnessing user feedback for machine learning', *Proceedings of the 12th International Conference on Intelligent User Interfaces, IUI '07*, ACM, New York, NY, USA, pp.82–91.

Stumpf, S., Rajaram, V., Li, L., Wong, W-K., Burnett, M., Dietterich, T., Sullivan, E. and Herlocker, J. (2009) 'Interacting meaningfully with machine learning systems: three experiments', *International Journal of Human- Computer Studies*, Vol. 67, No. 8, August, pp.639–662.

Whiffin, V.S., Crawley, C., Wang, Y., Li, Z. and Chen, F. (2013) 'Evaluation of machine learning for predicting critical main failure', *Water Asset Management International*, Vol. 9, No. 4, pp.17–20.

Wong, W-K., Oberst, I., Das, S., Moore, T., Stumpf, S., McIntosh, K. and Burnett, M. (2011) 'End-user feature labeling: a locally weighted regression approach', *Proceedings of the 16th International Conference on Intelligent User Interfaces, IUI '11*, ACM, New York, NY, USA, pp.115–124.

Zhou, J. and Chen, F. (2015) 'Making machine learning useable', *International Journal of Intelligent Systems Technologies and Applications*, Vol. 14, No. 2, pp.91–109.

Zhou, J., Khawaja, M.A., Li, Z., Sun, J., Wang, Y. and Chen, F. (2016) 'Making machine learning useable by revealing internal states update – a transparent approach', *International Journal of Computational Science and Engineering*, Vol. 13, No. 4, pp.378–389.

Zhou, J., Sun, J., Fang, C., Wang, Y., Taib, R., Khawaji, A. and Li, A. (2015) 'Measurable decision making with galvanic skin response and pupillary analysis', *ACM Transactions on Computer-Human Interaction*, Vol. 21, No. 6, pp.33:1–33:23.

Note

[1]This work was conducted when Jinjun Sun acted as a research engineer in NICTA.