Knowledge Extraction in Requirement Engineering with Machine Learning Perspective

Geet Sandhu
Assistant Professor
Amity School Of Engineering & Technology
Amity University, Haryana

Shally Pal
Btech CSE
Amity School Of Engineering & Technology
Amity University, Haryana

Atish
Btech CSE
Amity School Of Engineering & Technology
Amity University, Haryana

Pratap Pal
Btech CSE
Amity School Of Engineering & Technology
Amity University, Haryana

ABSTRACT
Requirement Engineering predicts the intended behaviour and constraints of the software solution well in advance of the software development process; hence it is a very crucial activity in the entire development process. Since requirements are specified in natural language, it becomes necessary to extract relevant knowledge from the given information. This paper reviews existing knowledge extraction techniques and gives an overview on how machine learning can help optimize knowledge extraction process.

General Terms
Requirement Engineering, Knowledge extraction, Machine learning

Keywords
Requirement Engineering-RE, Reinforcement Learning-RL, Machine Learning-ML

1. INTRODUCTION
Software engineering anticipates a systematic and quantifiable approach for development of a software solution[1]. Software development process starts from translating user needs into software requirements. Hence the prerequisite for developing any software solution has to go through the requirement engineering phase. Requirement engineering is a process of understanding needs and constraints specified by different stakeholders for a software solution[2]. Requirement Engineering (RE is a four step procedure of Requirement elicitation, requirement analysis, requirement validation and requirement documentation[2]. Requirement elicitation determines, comprehends, reports and realizes stakeholder needs and constraints. Requirement analyses focuses on in depth investigation of gathered user requirements. Requirement documentation involves specification of software requirements while Requirement validation focuses on whether the software solution is absolute enough to meet user satisfaction. RE holds the ability to affect the entire system as conflicts such as inconsistencies, ambiguities, incompleteness, imprecision, instability etc, if remained undetected initially would penetrate deep into the system, thus effecting subsequent phases of development. So before discussing RE process in much detail, it becomes important to realize what types of software requirements a requirement engineer would deal with. Software requirements describe user concerns and expectations. A good requirement would be parameterized on the basis of clarity, accuracy, clarity, traceability, completeness, correctness, and consistency and implementation independence. All these features filter a good requirement with others[3]. Software requirements broadly fall into two categories-functional requirements which describe functionality or services provided by the software solution and its related components while non functional requirements describes the constraints imposed on the design software solution[4]. Both types of requirements need to be properly modeled and quantified.

Since these requirements are specified in Natural language it becomes necessary to represent and reckon these requirements in a formal manner. For this relevant knowledge extraction is an important step so that they can serve as precise input to the Software Requirement Specification or SRS document which is the final outcome of Requirement Engineering activity. According to standard recommendation of IEEE on SRS, a good SRS should provide basis for instituting agreement or conformity between customer and supplier, reducing development effort, estimating cost and schedule, foundation for validation and verification enhancements [5]. Realizing the importance of a fine SRS, it becomes indispensable to optimize the knowledge Extraction process. One such emerging concept in this regard is through learning algorithm which is what Machine learning all about. This paper discusses about requirement engineering challenges in 2nd section. 3rd section discusses few approaches of Machine learning that helps optimize knowledge extraction process and presents research work of few authors. Section 4 discusses knowledge extraction metrics. Section 5 discusses a case study discussed by a researcher. Section 6 presents an observation on the discussed work of various authors. 7th section concludes the research and talks about the future scope in this area. Section 8 acknowledges the contribution of other research scholar in this work. Section 9 mentions all the references used in presenting the work in this paper.

2. REQUIREMENT ENGINEERING CHALLENGES IN KNOWLEDGE EXTRACTION
The main aim of user requirement is to meet stakeholder satisfaction. Inability to distinguish between nouns and verbs, to exclude irrelevant data and to integrate the contents are the

key issues which should be dealt with while extracting knowledge from natural language specified requirements.

3. KNOWLEDGE EXTRACTION & MACHINE LEARNING

Machine learning optimizes performance criteria using examples and past experiences in order to program computers which is necessary for knowledge extraction. Various researchers have proposed approaches to define different types of machine learning. Some of them are discussed in this section.

3.1 Supervised Learning

The aim of supervised learning is to provide a correct mapping from input to output[6].

3.1.1 Naïve base classifier

Wang.et.al proposes a Naive Bayesian classifer based ML approach for determining whether a phrase occurring in a document is a key phrase. This proposed approach considers term frequency in the document. Besides this, it takes into account the location of occurance of the key phrase I.e. If it is in the title, abstract, heading or subheading. It also includes the frequency of occurance in a paragraph of the document. Standard information retrieval metrics-precision. & recall are used to evaluate & validate the proposed technique. Experimental results yield 80% recall percentage. & the precision value is also high[7].

3.1.2 Classification theory

Classification is another technique of supervised learning in which class code is 0 or 1 on the basis of discriminant. The application of discriminant is prediction which finally classifies the result in 0 or 1 class. For example categorizing low risk and high risk customers for crediting loan to a bank based on the training data and classification rule[6].

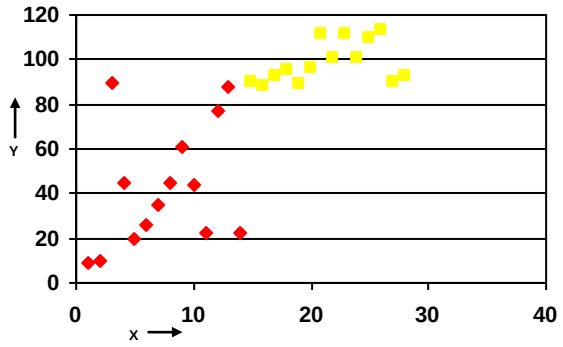


Figure 1- Classification theory by assuming training data sets on X axis representing Income and Y-axis representing Savings.

Red dots represents High Risk class while yellow dots represents Low risk class of customers for crediting loan.

3.1.3 Regression

Regression and classification are supervised techniques. Regression is applied where a problem solution is in terms of numbers. For example, in order to buy a car we consider one of the attribute as milage besides brand,year,engine capacity[6].

It is a statistical measure that attempts to determine the strenghth of the relationship between dependent variable (usually denoted by Y) and a series of other changing variables(known as independent variable).

It is also known as function approximation.

Regression is of two types- Linear regression and multiple regression.

Linear regression exercises one independent variable i.e X to predict or calculate Y.

Multiple regression exercises more than one independent variable to predict or calculate Y.

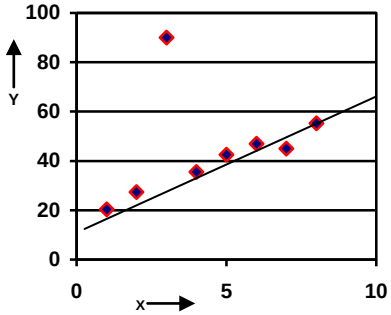


Figure 2- Graph for Regression theory where x-axis represents milage units and y-axis as price in thousand dollars.

3.2 Unsupervised Learning

In this type of learning only input data is given and there is no supervisor as in the case of supervised learning. Unsupervised learning is based on searching regularities in the input space. The behaviour shown by those regularities is assessed. On the basis of that ouput is predicted. Staistically this is same as density estimation[6].

3.2.1 Density based clustering

Clustering is a technique based on density estimation in which entities showing similar attributes are grouped together in clusters. For example demographic information of customers can be made an attribute for carrying out clustering for retrieving customer information[6].

3.2.2 Genetic Algorithm

This is another approach that exploits the concepts of learning in unsupervised manner. Genetic algorithm mimics the principle of natural selection to construct search and optimization procedure.

In this approach the design space gets converted to genetic space.

It works with coding of variables, hence the genetic space is discretized even if the function is continuous. It uses a population of points instead of a single point, hence can process multiple designs at same point. GA uses a population instead of single point, hence can process multiple designs at same point.

3.3 Reinforcement learning

Sultanov.et.al uses reinforcement learning to predict requirement traceability in software requirements. Reinforcement learning includes interaction of agents in the

environment to predict state transition based on agent's action. The author[8] starts by building a states space search which serves as the environment in Reinforcement learning. The agent navigates through the state space on the environment depending on the state in which it is currently present. The agents includes high level document, term in high level document, low level document, term in low level document and synonym. Learning is through navigation of agents in the above mentioned states.

Figure 3- Agent-State Transition diagram [8]

4. KNOWLEDGE EXTRACTION METRICS

4.1 Recall

Recall metrics is defined by the ratio of number of data retrieved to the number of relevant data in the collection.

4.2 Precision

Precision is described by the ratio of number of relevant data retrieved to the total number of collection.

5. CASE STUDY

Dimitriadis et.al in his work applies machine learning technique generate decision rules/trees to extract knowledge. The data sets chosen were pre classified. Domain expert classified the examples through decision rules. Classification algorithm is used where the the classifies is based on set of decision rules. The a forementraried approach by the author [9]is iterative & is a semi-automatic model of knowledge extraction as it uses collaborates with the application domain expert aswell as data mining expert. The final sets which are generated now,serves as input to machine learning algorithm. Using machine learning also, the post processing phase classifies the output into useful knowledge & common knowledge. Overall success rate & false positive are the two metrics used to evaluate performance.

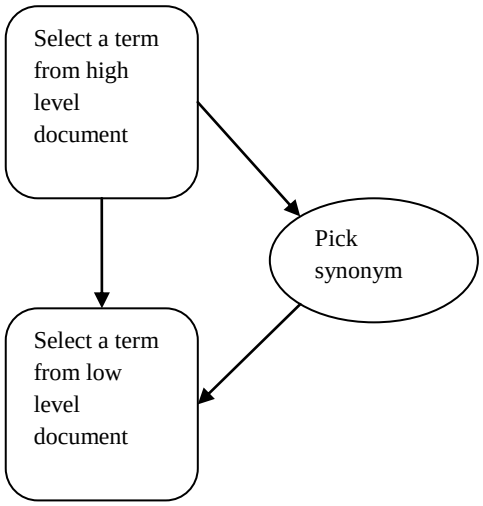
6. DISCUSSION & OBSERVATIONS

Naïve Bayesian classifier approach is a promising approach as it is able to achieve a high percentage (82.7%) recall even in cases where the key phrase extraction is from large no. of data set. Key phrase extraction procedure proposed also achieves good results when it is applied to text from varied domain & is not specific to a particular domain[7].

Reinforcement learning[8] as discussed has an ability to identify common text segments and their related linkages to other terms. Even synonyms are considered which is rarely found in other techniques. It yields a high recall-Precision value which shows that the approach is promising.

The proposed approach on classification theory[9] as discussed in the previous section gave good results when applied for plant science management in the field of agriculture to predict health & unhealthy status of plants. Naive bayes correctly classifies the status to a satisfactory result of 88%. The proposed approach by the author 9] is comprehensive & understandable to the user as it simplifies the process by letting domain expert intervene. The experimental results clearly validate the content system.

Moreover one significant achievement achieved by the author through his work that low information value attributes need not be eliminated but it can be easily updated later, once c



learly identified.

Table 1- Recall, Precision and accuracy values for various approaches

Metric	Recall	Precision	Accuracy
Naïve base Classifier	47.48%	82.7%	92%
Classification	-	-	83.3%
Reinforcement learning	90%	90%	-

7. CONCLUSION & FUTURE SCOPE

Naïve based classifier approach as discussed although achieve high 82.7% recall factor but is only able to achieve 47.48% precision percentage. Hence it has a wide research gap in terms of precision factor. Experimental results are based only on 10 different data sets. Hence it would be interesting to see whether it exhibits the same behavior if the data sets >10. This concept can be extended on widely available machine learning algorithm such as supervised, learning, unsupervised learning, reinforcement learning etc.

Significant experiments have been done for the proposed classification based iterative approach but future work is needed to verify the estimated inference rules. It would be interesting to see that whether the classification based proposed approach works well with multivariate data sets as well. The author has classified the status of plant in two extreme states i.e. Healthy & Not healthy. But work can be done to predict fuzziness in a situation where the status of the plant is partially healthy or partially unhealthy. Other machine algorithm concepts such as genetic algorithm neural n/w's, are also widely used, hence a comparative analysis would play a crucial role for future work in this regard. The proposed approach can be applied to other domains like engineering applications, healthcare etc.

8. ACKNOWLEDGMENTS

My sincere thanks to student rearch scholars-,Atish, Pratap and Shally who have contributed significantly towards the development of the research work.

9. REFERENCES

[1] IEEE Std 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology, 1990..

[2] R.Saranya, "Survey on Security Measures of Software Requirement Engineering", International journal of computer applications, vol 90, no 17, 2014.

[3] Hagal M.A, Alshareef, "A systematic approach to generate and clarify consistent requirements", IT convergence and security conference, IEEE, INSPEC accession no. 14047633, December 2013.

[4] Khatter et.al, "Impact of Non-functional Requirements on requirement evolution", 6th ICETET 2013, IEEE,2013.

[5] IEEE Std 830-1998, IEEE Recommended Glossary of Software Requirement Specification", 1998.

[6] Alpahydin, "Introduction to machine learning", PHI Learning Private Limited, 2008 edition .

[7] Wang et.al, 2003. Machine learning for keyphrase extraction based on Naïve Bayesian Classifier.

[8] Sultanov et.al , "Application of Reinforcement learning to Requirement Engineering-Requirement Tracing", IEEE,2013

[9] Dimitriadis et.al , "Applying Machine Learning to extract new knowledge in precision agriculture applications", Panhellenic Conference on Informatics, IEEE, 2008.