2019 IEEE 27th International Requirements Engineering Conference Workshops (REW)

On the Use of Word Embeddings for Identifying Domain Specific Ambiguities in Requirements

Siba Mishra and Arpit Sharma
EECS Department, Indian Institute of Science Education and Research Bhopal, India
{sibam,arpit}@iiserb.ac.in

***Abstract*—Software requirements are usually written in common natural language. An important quality criterion for each documented requirement is unambiguity. This simply means that all readers of the requirement must arrive at the same understanding of the requirement. Due to differences in the domain expertise of requirements engineer and other stakeholders of the project, it is possible that requirements contain several words that allow alternative interpretations. Our objective is to identify and detect domain specific ambiguous words in natural language text. This paper applies an NLP technique based on word embeddings to detect such ambiguous words. More specifically, we measure the ambiguity potential of most frequently used computer science (CS) words when they are used in other application areas or subdomains of engineering, e.g., aerospace, civil, petroleum, biomedical and environmental etc. Our extensive and detailed experiments with several different subdomains show that word embedding based techniques are very effective in identifying domain specific ambiguities. Our findings also demonstrate that this technique can be applied to documents of varying sizes. Finally, we provide pointers for future research.**

***Index Terms*—Requirements, quality, ambiguity, domain, word embedding, natural language.**

I. Introduction

Requirements are the basis for every project, defining what the stakeholders in a potential new system need from it, and also what the system must do in order to satisfy that need [1], [2]. All subsequent steps in software development are influenced by the requirements. Hence, improving the quality of requirements means improving the quality of the product. Requirements are usually written in natural language. Ambiguity is one of the major cause of poor quality requirements document [2], [3]. Ambiguity means that a single reader can interpret the requirement in more than one way or that multiple readers come to different interpretations. The latter could be present because of situations which involve people with different technical backgrounds and domain expertise. Requirements engineer usually has a background in computer science and may use typical computer science words in the requirements document which may be interpreted differently by other stakeholders of the project. This is because these stakeholders may not necessarily have a background in computer science. For example, the word "Windows" means an operating system when used in the context of computer science. On the other hand, for a building engineer, window means a space in the wall of a building. Similarly, for a computer engineer, the word "platform" denotes a complete software programming development environment, whereas for a chemical or petroleum engineer, it means a raised level surface on which people can stand. These ambiguities can lead to time and cost overrun in a software project. In the worst case scenario, these issues can lead to failure of the whole project. Therefore, it is of utmost importance that these issues are detected and resolved in the early phase of software development.

This paper focuses on detecting domain or context specific ambiguities in natural language text. To this end, we use word embeddings for identifying ambiguous words in a document. Word embedding is capable of capturing the context of a word and compute its semantic similarity relation with other words used in a document. It represents individual words as real-valued vectors in a predefined vector space. Each word is mapped to one vector and the vector values are learned based on the usage of words. Words that are used in similar ways tend to have similar representations. This means the distance between two words that are semantically very similar is going to be smaller. More formally, the cosine of the angle between such vectors should be close to 1.

We prepare a list L of most frequently used words in a computer science corpus created by crawling the computer science category on Wikipedia. Similarly, for each subdomain or application area (subcategory) of engineering domain (category), e.g., petroleum, civil, biomedical, environmental and chemical engineering, we create a corresponding corpus by Wikipedia crawling. Next, we estimate the ambiguity potential of words in list L when used in different subdomains of engineering. In other words, for each subdomain we find out which of the commonly used computer science words have a very different meaning and therefore should be cautiously used by the analyst in the requirements document. This technique can also be used for detecting ambiguous words during the requirements elicitation phase. Note that these subdomains have been selected to represent different application areas of engineering for which a computer based problem solving may be required.

The remainder of the paper is structured as follows: Section II discusses the related work. Section III provides the required background. Section IV explains our approach. We present the results and findings in Section V. Finally, Section VI concludes the paper and provides pointers for future research.

978-1-7281-5165-6/19/$31.00 ©2019 IEEE
DOI 10.1109/REW.2019.00048