CROWDSOURCING FOR REQUIREMENTS GATHERING: STATE OF PRACTICE

MUHAMMAD FAHMI ISMAIL
SABRINA TIUN
NOOR HASRINA BAKAR

Faculty of Information Science and Technology, National University of Malaysia

ABSTRACT

Requirements gathering plays vital role in software development, for software to be usable and likeable it must cater the best to communicate the needs, ideas and wishes from millions of stakeholders, hence always rely heavily on user perspectives. Thus, crowd involvement is becoming significant in requirements gathering phase of software development be it for software evolution or software development. Nowadays, requirement from user in the form of review of the software can be accessed in many reliable sources which is part of the crowdsourcing process. Therefore, this paper discusses the implementation of natural language text processing on crowdsource review and clustering algorithm approach such as K-Means to show result of the most frequent need of the user for a software requirement.

INTRODUCTION

Requirements Elicitation or requirements gathering is defined as the progression of obtaining a complete understanding of stakeholder's requirements. It is the initial and main process of requirements engineering phase where "what is to be done" is elicited and modeled (Leite 87). Elicitation process usually involves interaction with stakeholders to obtain their actual needs: what and how they imagine the software to be developed will solve their current problems. The requirements elicitation activities help stakeholders to express their needs and expectations of the new system. (Masooma, Asger 2015). One of the traditional methods of requirement elicitation is interview. For many years, team projects interview users and stakeholders for developing a new system or improving existing systems. It is true that interviews can be very successful in many occasions and it does help developers to understand the users need. However, there are also some drawbacks of interviews such as interviewee did not reveal their actual needs due to incomplete understanding of their needs. Additionally, sometimes requirements captures are ambiguous and its scope are ill-defined. There are also cases of miscellaneous miscommunication issues during Requirement Engineering (RE) process. Most of the time, important stakeholders who are from managerial team usually are busy people. Their time for RE interviews can be very limited. This may result that some important information might not be delivered during the RE process. Due to fundamental problems during traditional RE activities, developers option for an easier RE activity, that is via online collaboration including mining user reviews from the crowd (crowdsourcing). This option can provide benefits for software evolution and new software upgrade releases.

PROJECT MOTIVATION

Reuse of requirement is very tedious and error prone if done manually. Thus, an automated approach is needed to expedite the reuse process. Requirement documents from legacy systems are usually left in the archive, went wasted. Although improvements to the retired system are usually done from the weaknesses of previous system, very little work proposed on modernizing the new system through reuse of legacy documentations with crowd (potential users) involvement, for example through automated text extractions. The requirement engineering phase of traditional software development usually did not consider crowd involvement in gathering requirements. As a result, the developed software products sometimes did not meet crowd expectations. It is essential to encourage crowd involvement at the early stage of software development to ensure the software produced can meet potential users' expectations. Using unstructured data to represent requirements may cause lower precision values. Consequently, the existing approach failed to automatically cluster similar features that were extracted from unstructured textual data. Therefore, the objective of this project carry out an experiment to extract out or discover user's requirement on based on a crowdscource data using NLP approach. The following section will describe how the experiment was conducted as well as the data that has been used in the experiment.

REQUKM EXTRACTION EXPERIMENT

REQUKM

Derived from its name, REQUKM are abbreviation from the "Requirement UKM" which means the process of gathering requirement from the user. This application is built from Python programming language. The objective of this application is to read user review and identify the software functionality from the selected review from the result of K-Means clustering that implement in this system. This application use to extract the features that reside from the selected reviews. One of its outputs is to produce features in forms of noun phrases extracted via text processing. Then, the term is being count to find the weight of the term and to obtain phrase relatedness via number of its occurrences, this show how frequent the term in the review. Next is applying the TFIDF to find the coordinate for all the phrases in the document space. Lastly K-Means process take place to clustering into top term by group the noun phrase together. Then be visualize via word cloud to project the most frequent term that are known as functionality of the software that user need.

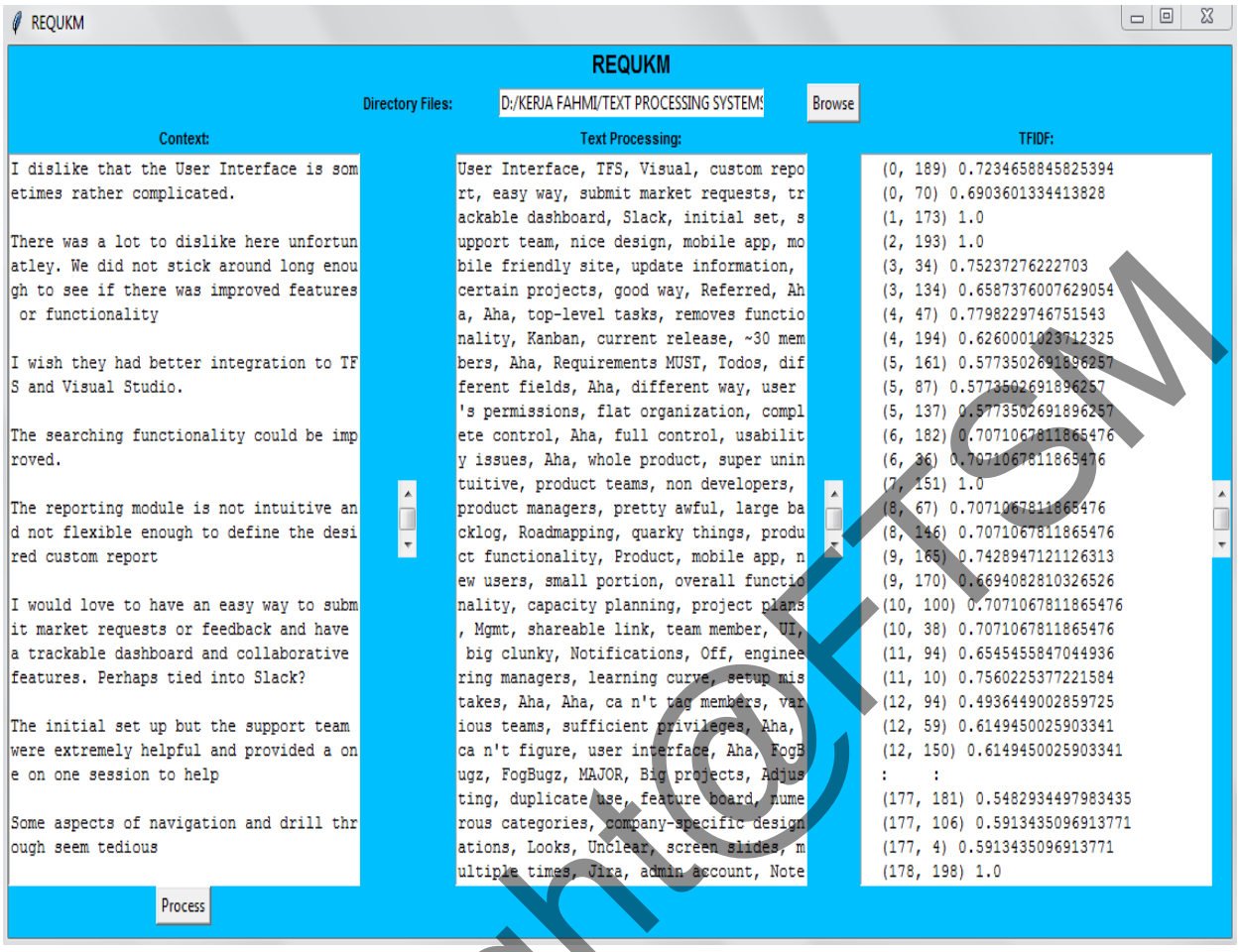

Figure 1: REQUKM interface

Flowchart of REQUKM

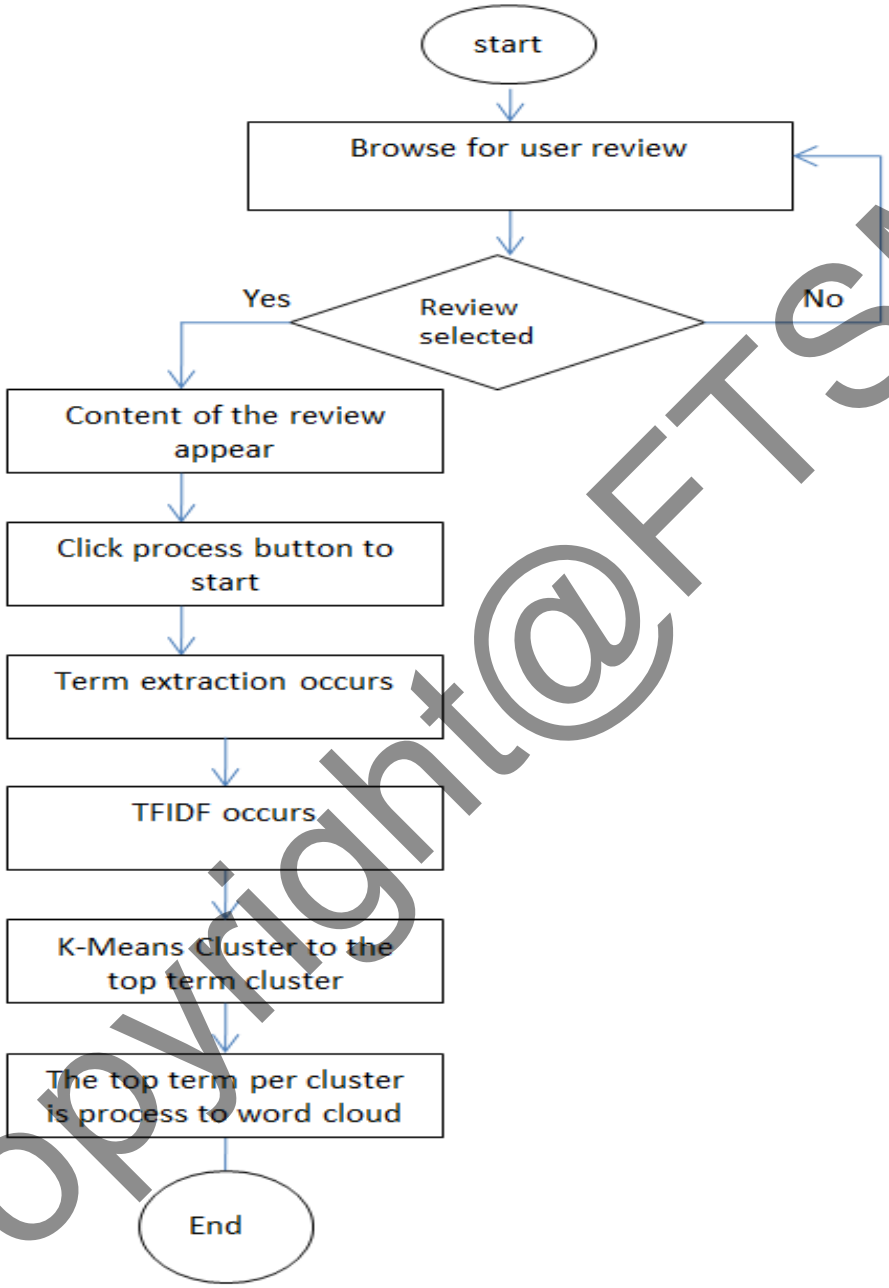

Figure 2: Flowchart of REQUKM

DFD of REQUKM

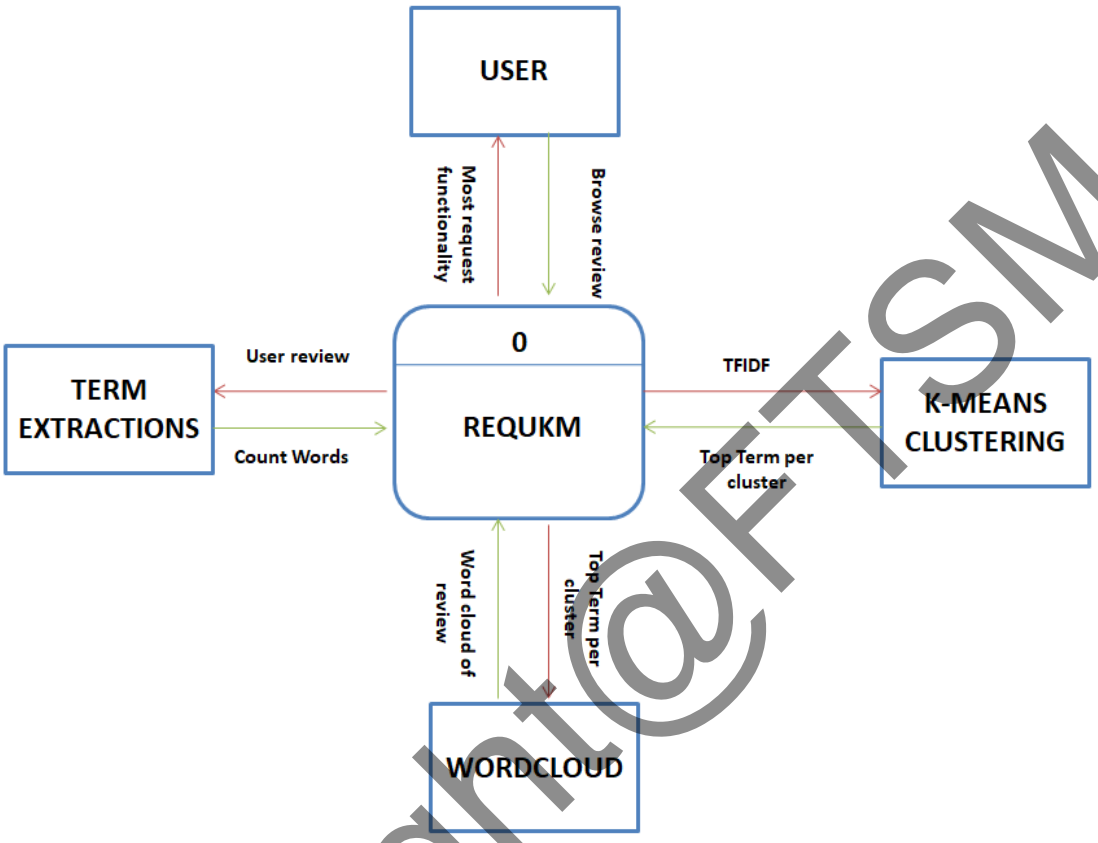

Figure 3: DFD of REQUKM

INTERFACE DESIGN

Figure 4 show the GUI of REQUKM. In this interface, user can select data of the review by clicking the browse button. After that, the directory of the selected files will be display in the text box to show where the data is located. At the same time, the content of the selected review will be automatically read and display in the first text box. Second, user need to click the process button to start the text processing. Then, the term from user review and TFIDF will be display on the second and third text box to indicate the process that occurs is successful.

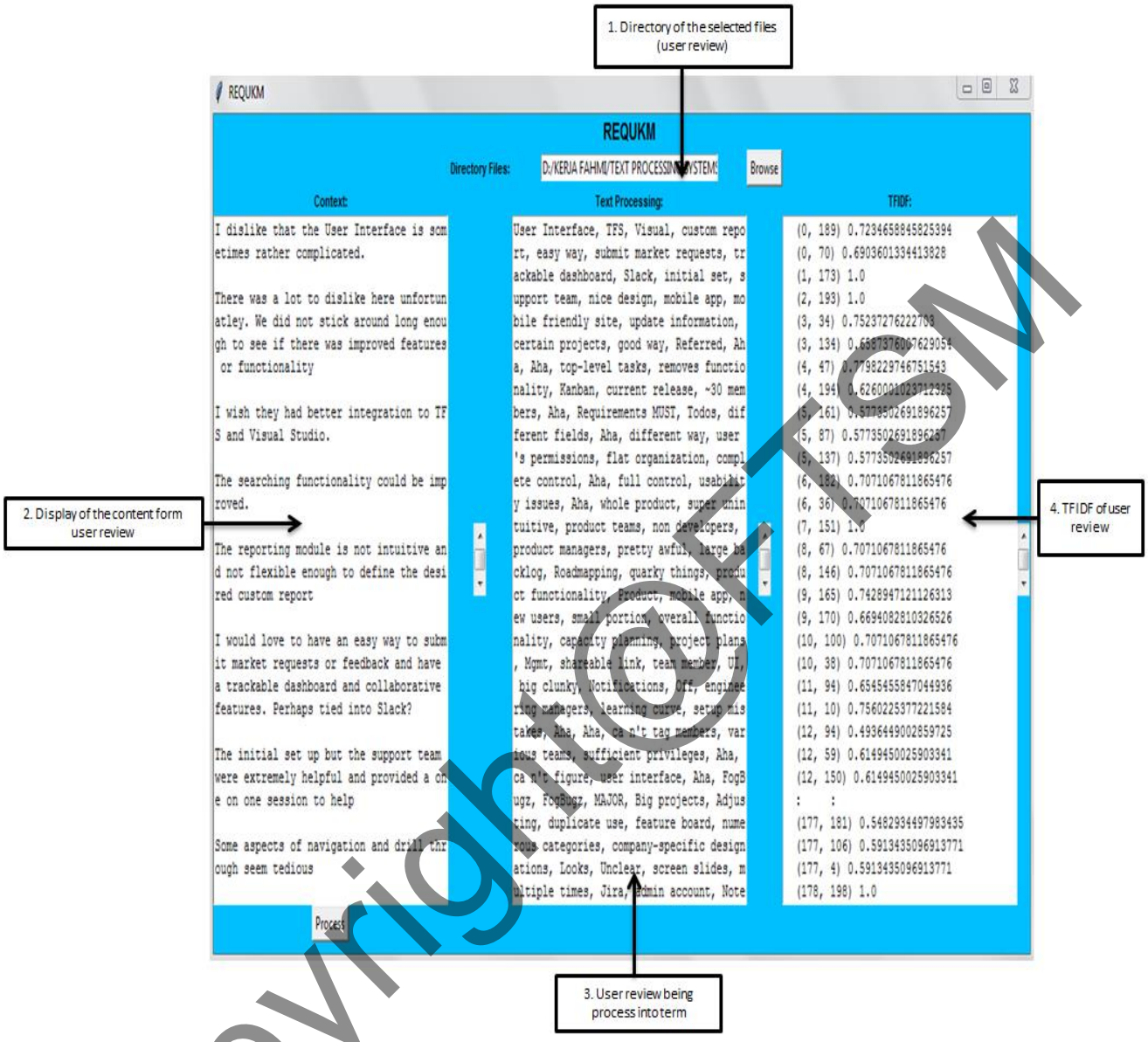

Figure 4: REQUKM GUI

Dataset

Our objective is to identify and cluster the software modernization from the selected software review that publishes by the crowds from variety of online platform such as G2CROWD, GetApp and Capterra. Therefore, the dataset are from crowdsource activity which is the consumer of the software give information, critics, and requirement towards the modernization of software. The total of 28 software reviews are scrapped as a dataset for this project and this raw data are then stored in the text files to be process through REQUKM tools. Figure 5 shows the example of website software review.

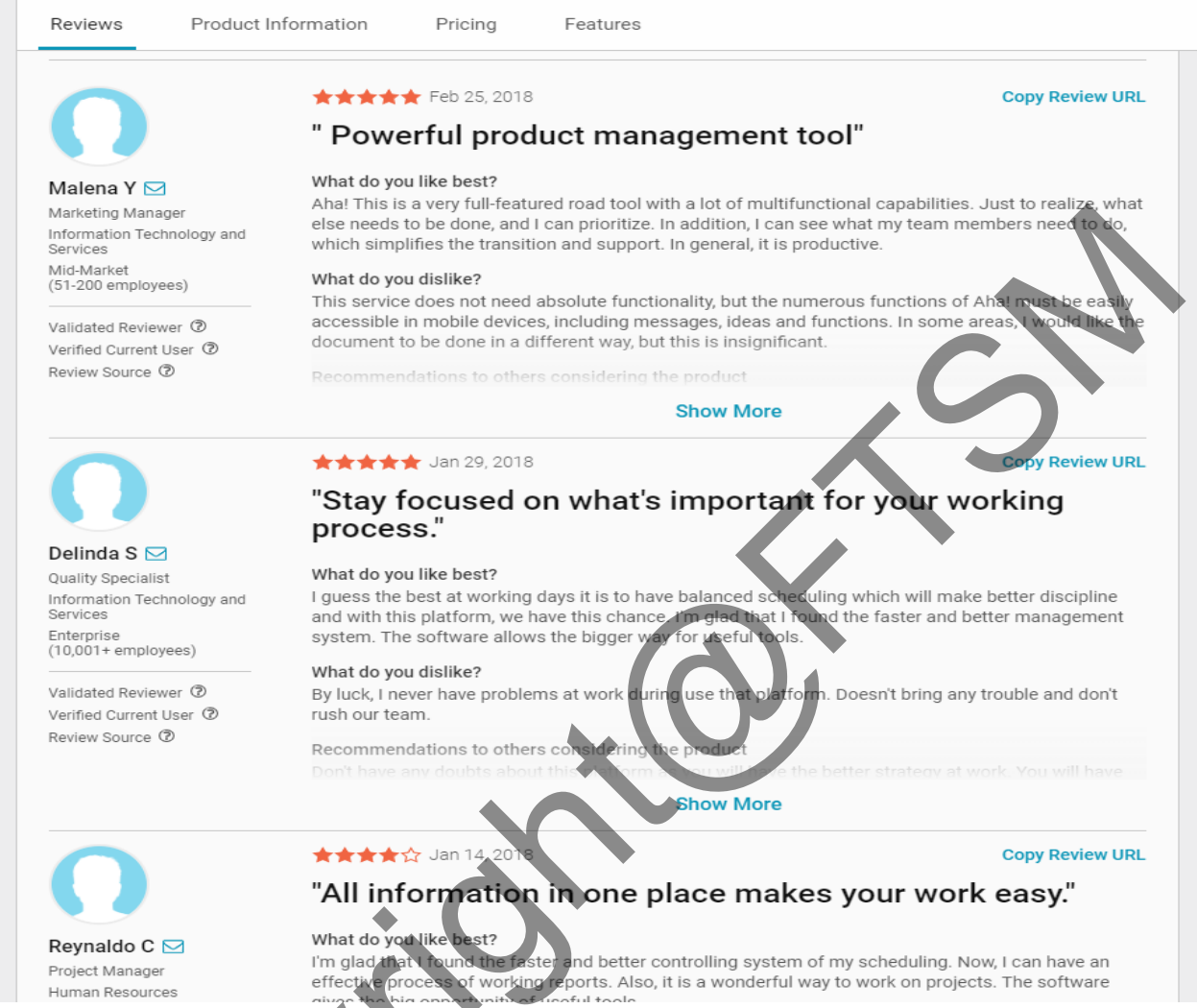

Figure 5: G2CROWD website software review

Term extractions

First of all, we clean the user review file via splitting into terms or word and handling punctuation and case, we clean the text to make it ready for modeling or cluster with machine learning. The step consists of split whitespace, remove punctuation, normalization, tokenization and split into words. Therefore the result show the sentence become like a term or word so we can progress to the next process to find the frequent and weight of those word that derived from user reviews.

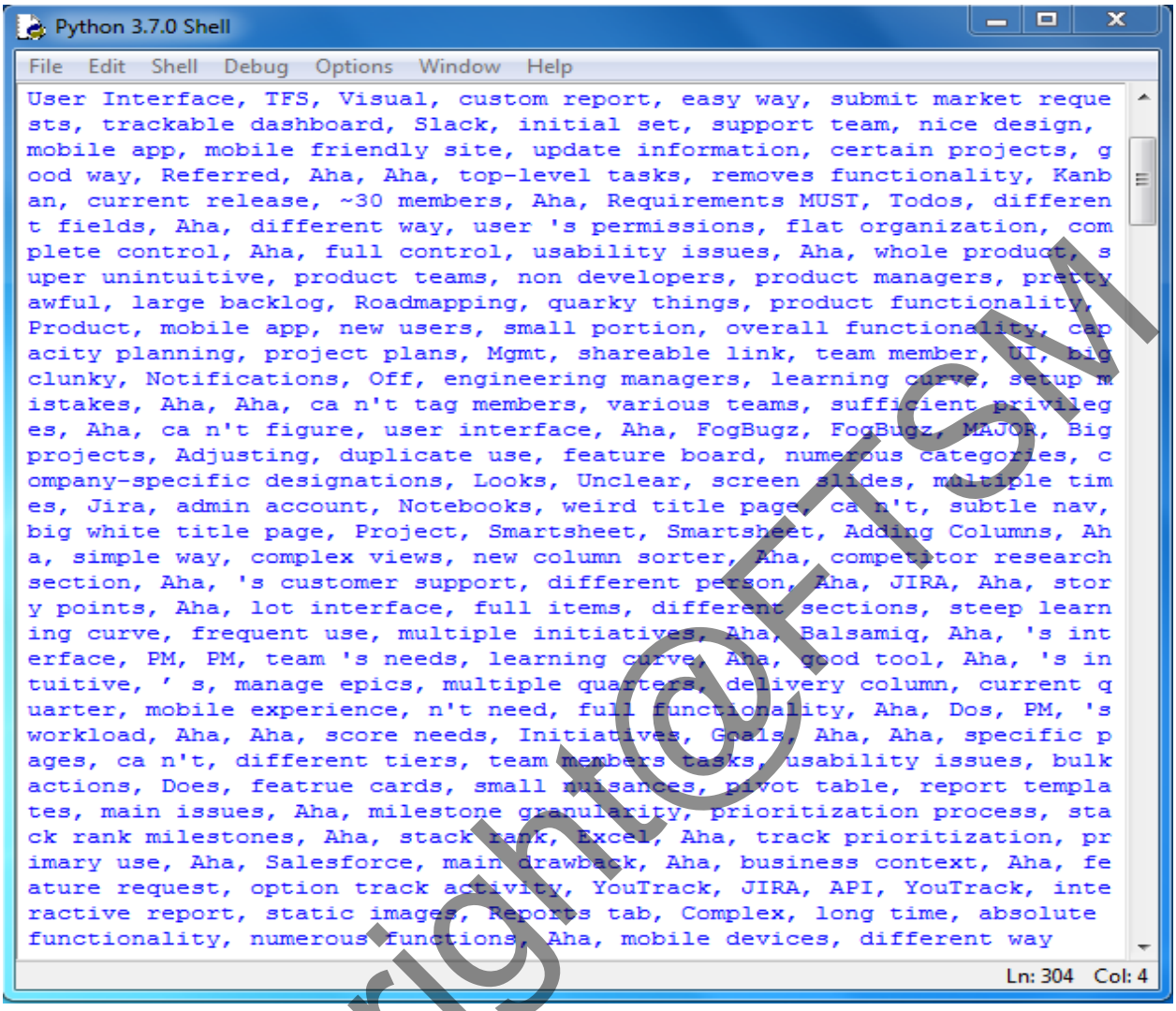

Figure 6: the term extractions

Count Words

Count word approach is use to count the word that derive from user review to find the most frequent word thus give an insight on what the requirement the user most needed. Figure 7 show the result of print from count word process

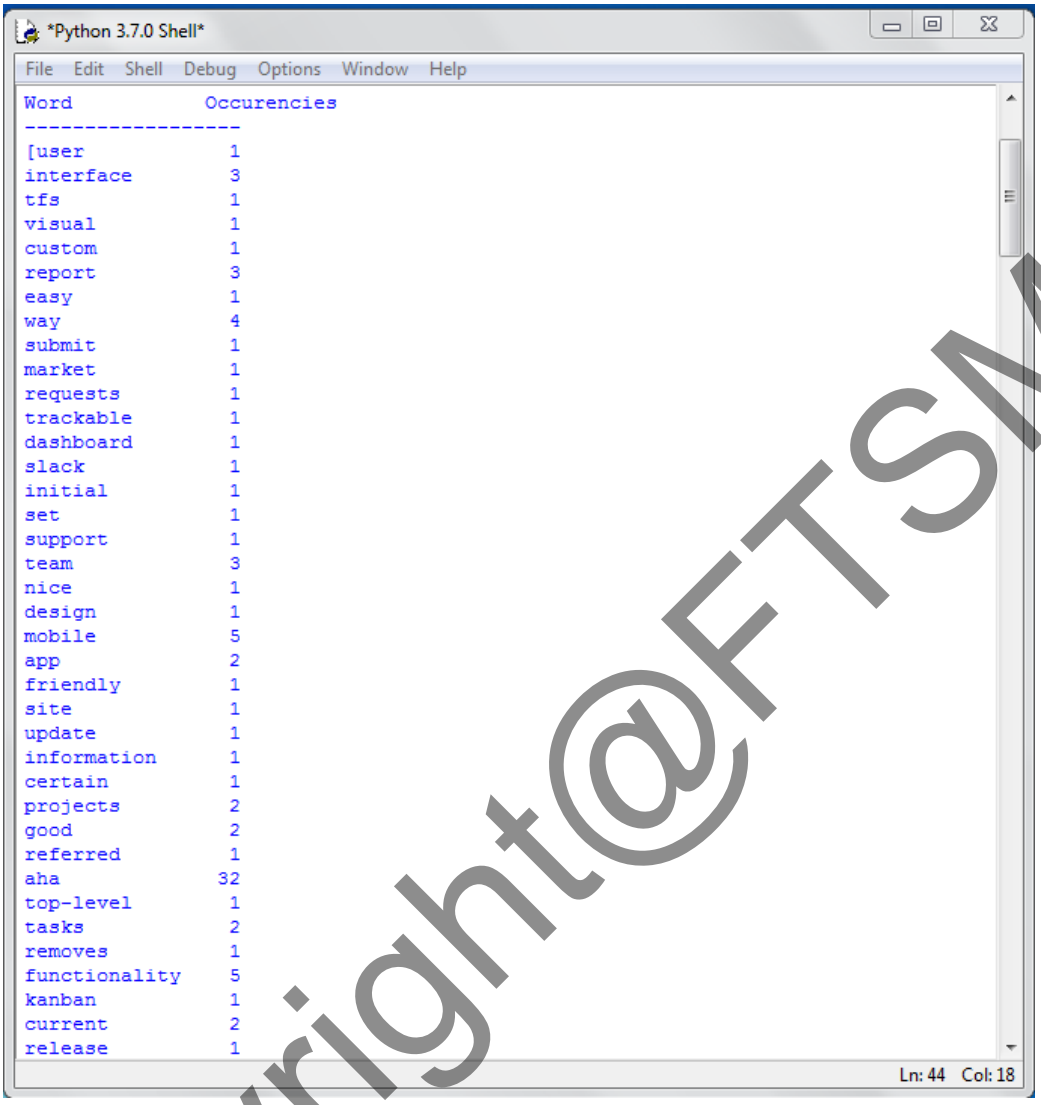

Figure 7: Count Words

TFIDF

The use of Term Frequency Inverse Data Frequency (TFIDF) in this program is to find the frequency and calculate the weight of rare words for each user review. After the text processing is done, to make sure the process is accurate, we apply TFIDF and export it to the text file and compare with both method, which is analyze the text file and applying the K-means cluster to gain result. Figure 8 show the TFIDF process that been done to the user review.

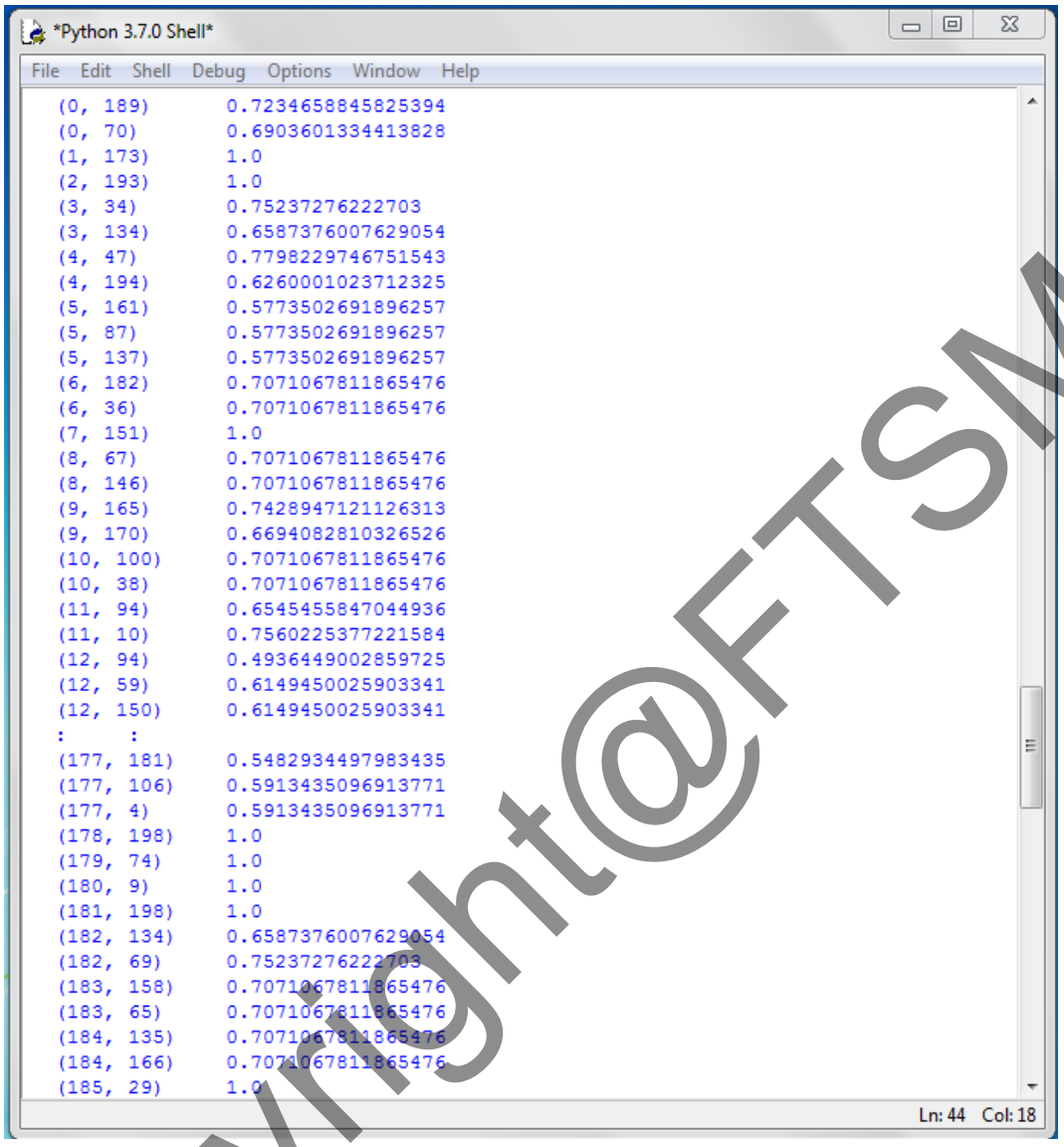

Figure 8: TFIDF

K-Means

This program use K-Means which is of one unsupervised machine learning algorithm. This approach will create a cluster automatically from the derived user review that been process through text processing, count word and TFIDF. We use step known as feature extraction because K-Means deal with numbers that derived from TFIDF process before, thus give us the statistical numerical for implementation. Even though clustering can belong to many cluster. We decide to only apply one cluster to gain the most top term cluster. Figure 9 show top term cluster

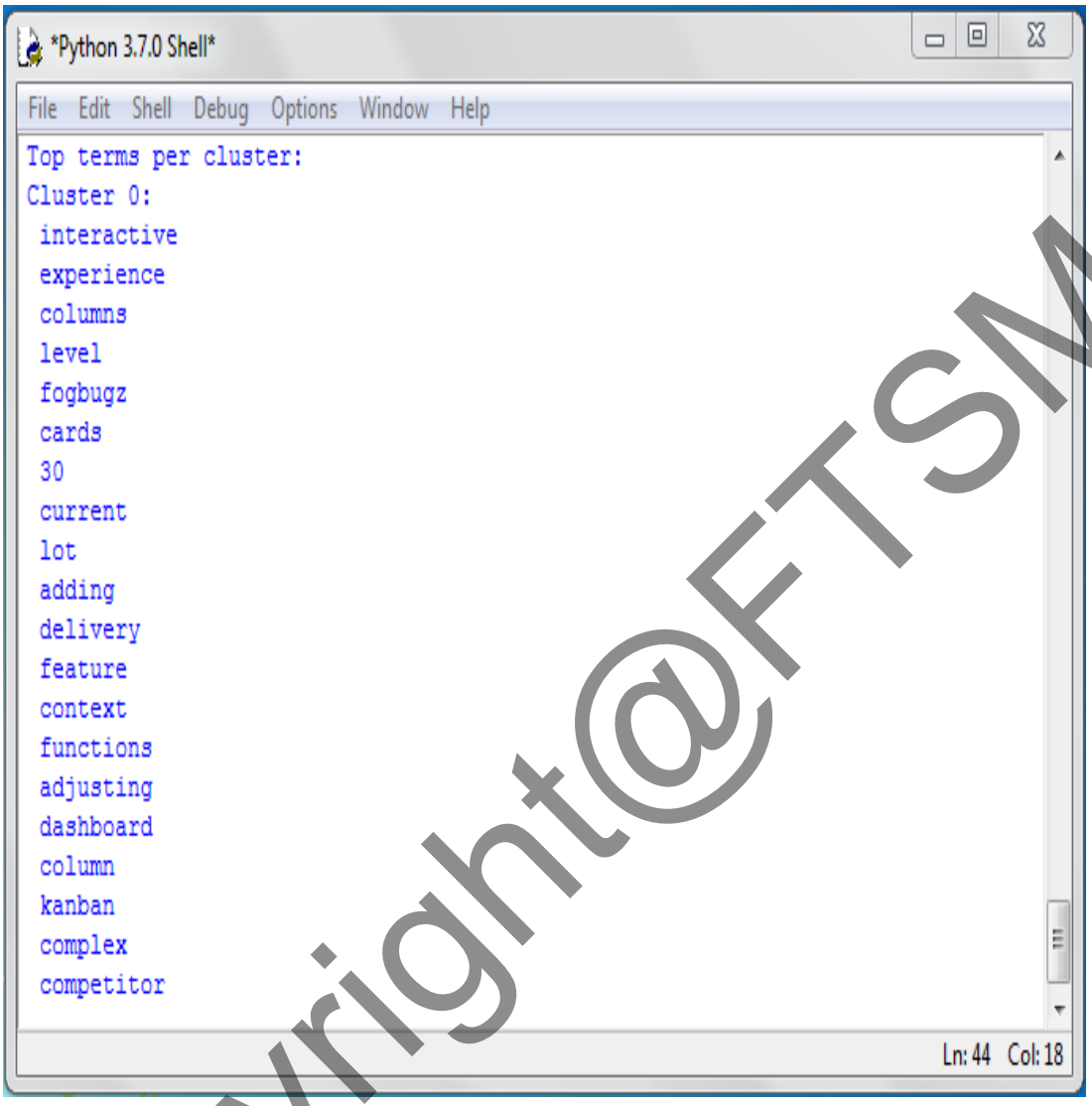

Figure 9: K-Means Cluster

Word Cloud

Figure 10 shows the word cloud that occurs from the K-Means clustering that show the frequent term of the user review which indicates the most word that request from the user.

Figure 10: Word cloud of the top term cluster

CONCLUSION

The purpose of this project is to understanding the automated approach which selecting requirement from user reviews that can be more systematic and easier. With this understanding, application being develop for use the proposed extraction algorithm that can be demonstrate as a useful process model for analyze requirements that encouraging crowd involvement that hoping to help in software development and modernization.

ACKNOWLEDGEMENT

This project is financially supported by UKM grant under research code: GGPM-2017-025.

REFERENCE

Leite, and Julio Cesar S P, 1987. A Survey on Requirements Analysis. Advanced Software Engineering Project Technical Report RTP-071, University of California at Irvine, Department of Information and Computer Science, June 1987.

Masooma and Asger, 2015. Comparison of Various Requirements Elicitation Techniques. International Journal of Computer Applications 116 (4): 8 - 15 April 2015.