Department of Computer Science

CPCS 351: Software Engineering (I)

Fall 2023

Research component

Summary:

Software Requirements Classification Using Machine Learning Algorithms

Name :Anfal sultan alshehri

ID:

Section : B8A

Instructeur :Dr. Mai Fadel

Table of Contents:

Table of figure:

Table of Table :

Introduction

Software engineer analyzes the user requirements and design software that will satisfy those requirements. To understand those requirements software engineers go throw a set of procedures to comprehend the user's needs and requirements. Some of those procedures are identify business needs, and define business requirements. Those activities are in the planning phase of the software life cycle.

Defining business requirement plays a big role in the planning phase since it's a contract between the software engineer and the customer "what the project should deliver? ". Moreover, classifying requirements in software engineering became a critical task for engineers to perform. This study shows the comparison between differents methods to classify user requirements using ML algorithms.

Edna Dias Canedo and Bruno Cordeiro Mendes study Software Requirements Classification Using Machine Learning Algorithms. That works by applying text feature extraction techniques for the purpose of classifying requirements. In this study there are three modeling techniques for text feature extraction will be used :

- Bag of Words (BoW).
- Term Frequency–Inverse Document Frequency (TF-IDF).
- Chi Squared (CHI2).

Canedo and Mendes experiment's classify Software Requirements into :

- Functional Requirements.
- Non-functional Requirements.
- sub-classes of Non-Functional Requirements.

The ML algorithm that will be used are:

- Logist Regression (LR).
- Support Vector Machine (SVM).
- Multinomial Naive Bayes (MNB).
- k-Nearest Neighbors (kNN).

Summary

1. Text Classifying :

Text classifying is a machine learning technique that classifies and groups documents that have the same attributes and features under the same class.

The application of this technique is extremely wide such as :

1- Gmail spam classifier.

2- Facebook's hate speech detection.

3- Linkedin's inappropriate profile flagging.

The value of this technique is making the search and managing text data easier and saving time and effort.

In this research, the software requirements will be classified that has been mentioned above. Software requirements must be well-known and well-defined.

However, classifying the subcategory of Non-functional requirements and the automatic classification of functional requirements is quite difficult to do. Applying the ML techniques will save the effort of the software engineers and make it an easier task to do.

In this experiment, we will use the most effective modeling techniques in text classifying according to many studies which are (bag of word, Term Frequency and Inverse Document Frequency, and chi-squared).

To apply those techniques, first of all, the text should be transferred in a form that can be consumed by any Natural Language Processing (NLP). This transformation pross called *Text Normalizing*. Text Normalizing includes many prosses (Case conversion, spelling correction, removal of irrelevant words and other unnecessary terms, such as articles and pronouns, stemming and lemmatization, etc..).

Secondly, for the sake of the Machine Learning algorithm to perform the operation of classifying requirements text should be transformed into a two-deamination array. And this pross is called *Vectorization Text.*

In this experiment we will use three vectorization text which are :

1- Bag of Words.

2- Term Frequency—Inverse Document Frequency (TF-IDF).

3- Chi Squared (Feature Selection).

This study use four supervised algorithms of ML(k-NN (k-Nearest Neighbor), SVM (Support Vector Machines), Logistic Regression, and Multinomial Naive Bayes).

The performance will be measured by the following matrix :

1- The percentage of right classified requirement (Precision).

2- The number of the requirements that are detected correctly by the classifier (Recall).

3- The companies between Precision and Recall (F-1 Score).

2. *Experiment purpose:*

This experiment aims to answer two questions :

Q1. Which feature extraction technique works best (BoW vs. TF-IDF vs. CHI2), for classifying Software Requirements into Functional Requirements and Non-Functional Requirements, and the sub-classes of Non-Functional Requirements?

Q2. Which Machine Learning Algorithm provides the best performance for the requirements classification task?

3. *Design and procedures :*

The experiment of classifying the software requirement will be done in three main steps first set up the data , perform classifying on the data ,then measure the performance of classifying.

1- Clean the textual data (Normalization Phase).

2- Translate normalized text into numerical vectors (Feature Extraction Phases).

3- Get rid of noise data before starting classifying (Feature Selection Phase).

4- Classifying the data using ML algorithms (Classification Phase).

5- Calculate the performance of each ML algorithm's efficacy (Performance Measure Phase).

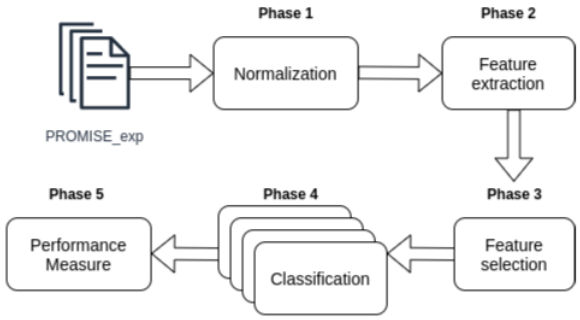

Figure 1:experiment phases

This experiment use *Python* programing language. Also, the textual data was from the *PROMISE_exp* database which is a database that crate to improve ML models in software engineering domains. Table1 shows us the *PROMISE_exp* component.

Type of Requirement	Number of Requirements
Functional Requirement (FR)	444
Availability (A)	31
Legal & Licensing (L)	15
Look & Feel (LF)	49
Maintainability (MN)	24
Operability (O)	77
Performance (PE)	67
Scalability (SC)	22
Security (SE)	125
Usability (US)	85
Fault Tolerance (FT)	18
Portability (PO)	12
Total	969

Table 1:Number of requirements per label in the database.

3.1. **Prepare Data Phases .**

3.1.1.**Normalization Phase:**

The used data in this study has been normalized by the NLTK library. NLTK library done the Normalization by several pross such as converting words into lowercase, deleting all stop-words (the, a, etc..), and transforming all words into root form.

	Text	Class
1	The system shall refresh the display every 60 s.	PE
2	The application shall match the color of the schema set forth by Department of Homeland Security.	LF
3	If projected the data must be readable. On a 10 × 10 projection screen 90% of viewers must be able to read Event/Activity data from a viewing distance of 30.	US
4	The product shall be available during normal business hours. As long as the user has access to the client PC the system will be available 99% of the time during the first six months of operation.	A
5	If projected the data must be understandable. On a 10 × 10 projection screen 90% of viewers must be able to determine that Events or Activities are occurring in current time from a viewing distance of 100.	US
...	...	...
965	The system should be portable to various operating environments.	PO
966	Registered User must be able to maintain his/her session information for at least 60 min of inactive session before the system prompts him to log out of the system. The registered user must be provided with all the options of the E-store regardless of the time when he/she logs in.	F
967	The entire website must be user-friendly and easily navigable. The website must be provided with a site map for quick access to a particular link according to the requirement specification. The user must be able to find what he/she wants from the site without any difficulty. The website must adhere to branding schemes and the layout of the web pages must be uniform throughout.	US
968	The system shall support up to 10,000 simultaneous users against the central database at any given time and up to 5000 simultaneous users against the local servers at any one time. The performance of the website must be optimal increase of huge loads and hence appropriate load balancing must be done to achieve this. There can be any number of mirror servers readily available in case of huge loads without the user getting any delay.	PE
969	The website must provide highest degree of security to the registered users. All the transactions that are made must be secured. The sensitive information passed to and from the website must be secured. Identity theft and other security related issues must be solved. Unauthorized transmission of sensitive information of the user to third party websites for reference must be avoided. On the basis of user agreement the information must be processed. All the information about the registered user must be securely stored in the central database.	SE

Table 3:Data Before The Normalization Phase.

1	System shall refresh display every second.	PE
2	Application shall match color schema set forth department homeland security.	LF
3	Project data must readable projection screen viewer must able read event activity data view distance.	US
4	Product shall available normal business hour long user access client pc system available time first six month operation.	A
5	Project data must understandable projection screen viewer must able determine event activity occur current time view distance.	US
...	...	...
965	System portable various operate environment.	PO
966	Register user must able maintain session information least minute inactive session system prompt log system registered user must provide option regardless time log.	F
967	Entire website must easily navigable website must provide site map quick access particular link accord requirement specification user must able find want site without difficulty website must adhere brand scheme layout web page must uniform throughout.	US
968	System shall support simultaneous user central database give time simultaneous user local server one time performance website must optimal in case huge load hence appropriate load balancing must do achieve number mirror server readily available case huge load without user get delay.	PE
969	Website must provide high degree security registered user transaction make must secure sensitive information pass website must secure identity theft security relate issue must solve unauthorized transmission sensitive information user third party website reference must avoid basis user agreement information must process information registered user must securely store central database.	SE

Table 2:Data After The Normalization Phase.

3.1.2.**Feature Extraction Phase:**

In this phase, the data convert from textual form to vectors using (BoW and TF-IDF) techniques. After that, we can count the most repeated words in the data Table4.

Top 10 Most Important Features		
Position	**BoW**	**TF-IDF**
1º	must	must
2º	information	information
3º	user	sensitive
4º	website	website
5º	security	registered
6º	registered	security
7º	secure	secure
8º	sensitive	user
9º	high	third
10º	agreement	issue

Table 4:Most Frequent words.

3.1.3.**Feature Selection Phase:**

In the third phase, the data will be flirting from unnecessary information applying Chi-Squared (CHI2).

3.2. Classification Phase:

Now the data is ready to implement the four ML classifying algorithms(kNN, SVM, MNB, and LR). To classify requirements into three classes (Functional Requirements and Non-Functional Requirements, sub-classes of Non-Functional Requirements, and Functional Requirements and sub-classes of Non-Functional Requirements). For this experiment, all possible combinations of vectorization/classification techniques must be generated. To generate all combinations we will use the *GridSearchCV* function to achieve the experiment aim Q1 and Q2.

3.3. Performance Measure Phase:

This study will measure the efficiency of the flowing classifications:

1- Effectiveness in the binary classification of requirements.
2- Effectiveness in the multiclass classification of non-functional requirements.
3- Effectiveness in the multiclass classification of requirements, including Non-Functional Requirements (NFRs) and Functional Requirements (FRs).

Using all possible companion of vectorization/classification techniques. The performance of each technique will be measured using special matrixes which are: (Precision, Recall, and F1-score).

Precision = $\frac{TP}{(TP + FP)}$.

Recall = $\frac{TP}{(TP + FN)}$.

F1-score= $\frac{2TP}{(2TP + FP+FN)}$.

TP= Ture classifications.

FP= False classifications.

FP= the number of times that the class appears in the test data.

The Dataset was dived into 10 corpus. 9 of them for training the ML algorithm while and 1 corpus for the test.

4. The result of this experiment :

After the classifying software requirements experiment was performed and the performance was measured. The experiment's purpose question can be answered :

Q1. Which feature extraction is the best?

Q2. Which Machine Learning Algorithm provides the best performance for the requirements classification task?

From the records below Table5,6,7, the most effective vectorization/classification techniques for classifying Software requirements are TF-IDF/LR. On a scale of 0 to 1. TF-IDE/LR has a small variation in the performance measurement.

Binary Classification									
	BoW			**TF-IDF**			**CHI^2**		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
SVM	0.90	0.90	0.90	0.91	0.91	0.91	0.90	0.90	0.90
MNB	0.91	0.91	0.91	0.91	0.91	0.90	0.89	0.89	0.89
kNN	0.82	0.82	0.82	0.87	0.87	0.87	0.84	0.84	0.84
LR	0.88	0.88	0.88	0.91	0.91	0.91	0.89	0.89	0.89

Table 5:Binary Classification

Non-Functional Requirements Classification									
	BoW			**TF-IDF**			**CHI^2**		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
SVM	0.68	0.67	0.66	0.73	0.73	0.72	0.72	0.71	0.71
MNB	0.71	0.73	0.72	0.71	0.71	0.71	0.69	0.70	0.68
kNN	0.56	0.48	0.49	0.66	0.66	0.66	0.62	0.63	0.62
LR	0.71	0.71	0.70	0.75	0.75	0.74	0.70	0.71	0.70

Table 6:Non-Functional Requirements Classification

Classification with 12 Granularities									
	BoW			**TF-IDF**			**CHI^2**		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
SVM	0.73	0.73	0.72	0.78	0.78	0.78	0.77	0.77	0.76
MNB	0.77	0.77	0.77	0.76	0.77	0.76	0.74	0.74	0.73
kNN	0.63	0.60	0.60	0.72	0.73	0.72	0.67	0.69	0.68
LR	0.76	0.77	0.76	0.78	0.79	0.78	0.76	0.77	0.76

Table 7:Classification with 12 Granularities

5. *Threats and Validity :*

Data unbalanced the number of classes in each classification was unbalanced In the first classification, there was only two class (functional, and non-functional). The second classification has 11 classes for the non-functional. In the third classification, there were 12 classes 11 are non-functional, and one for functional.

6. *Conclusions :*

Classifying software requirements is an essential task for engineering to do. Using machine learning algorithms will save engineering time and effort. This experiment aims to find the best way to classify requirements. Also, the best model for vectorization of the data.

The experimenter use python language and other tools to accomplish this experiment. By finding all combinations of vectorization /classifying to using four matrices to measure the performance of this study.

In conclusion, the techniques that has highest accuracy are TF-IDF/LR.LR (Logiest Regression) is the Classifying algorithm and TF-IDF (Term Frequency—Inverse Document Frequency) Vectorization.

Reference :

1. Dias Canedo, E., & Cordeiro Mendes, B. (2020). Software Requirements Classification Using Machine Learning Algorithms. *Entropy*, *22*(9), 1057. https://doi.org/10.3390/e22091057
2. Keyword: Machine Learning, Software Requirements.