Kalpa Publications in Computing

Volume 14, 2023, Pages 124–126

Proceedings of V XoveTIC Conference. XoveTIC 2022

Towards an automatic requirements classification for Spanish software requirements

María Limaylla-Lunarejo[1], Nelly Condori-Fernandez[23], and Miguel R. Luaces[1]

[1] Fac. Informática,Database Lab. Universidade da Coruña, CITIC, Elviña, 15071, Spain
{maria.limaylla, miguel.luaces}@udc.es
[2] Universidad de Santiago de Compostela, Spain
n.condori.fernandez@usc.es
[3] Vrije Universiteit Amsterdam, The Netherlands
n.condori-fernandez@vu.nl

Abstract

Several machine learning (ML) algorithms in combination with natural language processing (NLP) techniques have been used in recent years in a promising way for the automatic classification of software requirements. Nevertheless, several works have focused on the English language. Due to the lack of work in the Spanish language, we performed a controlled experiment using ML algorithms in combination with text vectorization techniques to investigate the best combination for Spanish requirements classification. Based on f1-score metrics, we found the combination of SVM with TF-IDF performs better than other combinations, with a value of 0.95 for functional and 0.79 for non-functional classification.

1 Introduction

Spanish is currently the second mother tongue in the world by number of speakers [3]. Hence, it is important to expand the knowledge on the performance of automatic classification for software requirements written in Spanish. The present research reports the performance metrics for functional/non-functional classification previously introduced in [5] and presented here with more details. To evaluate the performance of our classification models, we performed an experiment using a similar strategy to the one defined by Dalal and Zaveri (2011) [2]. For data pre-processing, we used some traditional techniques of NLP like tokenization, stopwords, and stemming for Spanish. We selected a set of Shallow ML algorithms (i.e. NB, GNB, LR, and SVM) and two Deep Learning (DL) algorithms (CNN, BETO) for training and testing. BOW and TF-IDF in combination with n-grams: Unigram, Bigram, and Trigram were used for text vectorization in shallow algorithms. The CNN architecture is based on [4] and BETO [1] is a pre-trained model based on Bert and training on a Spanish corpus.The classification performance was evaluated using very well-known metrics (i.e., accuracy, precision, recall, and f1-score). A new Spanish requirements Dataset was presented in [5]. This dataset is a collection of 389 requirements from 13 final degree and 2 master's projects from the University of

A Coruña, labeled in functional (300) and non-functional (89) and published in Zenodo[1] and Hugging Face[2].

2 Results

The results obtained from the Shallow ML algorithms in combination with the text vectorization techniques for 10 folds are presented in Table 1. GNB and LR perform better in combination with BOW than TF-IDF, and SVM performs better with TF-IDF than BOW. Bigram gets better results than Unigram for GNB, LR, and SVM in combination with TF-IDF, and NB and GNB in combination with BOW. For LR and SVM in combination with BOW, Bigram get lower values than Unigram. Trigram gets the same or lower results than Bigrams in almost all models. Table 2 shows the results of training CNN and BETO algorithms. Both models give the same results in 10-folds for functional and a difference of 0.03 for non-functional classification.

		BOW + Unigram		BOW + Bigram		BOW + Trigram		TF-IDF + Unigram		TF-IDF + Bigram		TF-IDF + Trigram	
		F	NF	F	NF	F	NF	F	NF	F	NF	F	NF
NB	a	0.88		0.87		0.87		-	-	-	-	-	-
	p	0.94	0.73	0.96	0.69	0.96	0.68	-	-	-	-	-	-
	r	0.91	0.79	0.88	0.87	0.87	0.88	-	-	-	-	-	-
	f1	**0.92**	**0.75**	**0.92**	**0.76**	0.91	0.76	-	-	-	-	-	-
GNB	a	0.84		0.87		0.87		0.83		0.86		0.85	
	p	0.87	0.73	0.87	0.87	0.88	0.84	0.86	0.73	0.86	0.86	0.86	0.82
	r	0.92	0.54	0.97	0.53	0.96	0.55	0.93	0.49	0.97	0.47	0.96	0.47
	f1	0.90	0.60	**0.92**	**0.65**	**0.92**	**0.65**	0.89	0.57	0.91	0.60	0.91	0.58
LR	a	0.90		0.89		0.89		0.84		0.85		0.86	
	p	0.92	0.83	0.91	0.80	0.92	0.83	0.83	0.95	0.84	0.97	0.85	0.97
	r	0.95	0.72	0.95	0.69	0.95	0.71	1.00	0.29	1.00	0.37	1.00	0.39
	f1	**0.93**	**0.76**	**0.93**	**0.74**	0.93	0.75	0.90	0.43	0.91	0.53	0.92	0.55
SVM	a	0.87		0.87		0.87		0.91		0.92		0.92	
	p	0.92	0.72	0.92	0.73	0.92	0.72	0.92	0.89	0.92	0.91	0.92	0.91
	r	0.91	0.73	0.92	0.72	0.92	0.72	0.97	0.70	0.97	0.72	0.97	0.72
	f1	0.92	0.72	0.92	0.71	0.92	0.71	0.94	0.77	**0.95**	**0.79**	**0.95**	**0.79**

Table 1: Results of Shallows algorithms in combination with text vectorization techniques

		num_folds = 1		num_folds = 5		num_folds = 10	
		F	NF	F	NF	F	NF
CNN	acc	0.96		0.90		0.89	
	p	0.97	0.94	0.93	0.80	0.92	0.82
	r	0.98	0.89	0.94	0.74	0.95	0.72
	f	0.98	0.91	0.93	0.77	**0.93**	**0.75**
BETO	acc	0.95		0.92		0.88	
	p	0.94	1.00	0.92	0.93	0.91	0.81
	r	1.00	0.78	0.98	0.70	0.95	0.66
	f	0.97	0.88	0.95	0.79	**0.93**	**0.72**

Table 2: Results of DL algorithms

Figure 1 shows the accuracy and loss curves for CNN and BETO model for 1-fold. The training accuracy for the CNN model increases to near 100%, while the validation accuracy obtains approximately 95%. Likewise, the loss decreases gradually throughout the epochs, reaching a value less than 0.2 and with a difference of approximately 0.1. The resulting values show a minimal indication of overfitting in this model. For the BETO model, the accuracy values increased by over 90% (for both training and testing values), and the value of the loss was below 0.3, also indications of overfitting. These results suggest the necessity of validation of the capability of generalization of these models.

[1]https://doi.org/10.5281/zenodo.6556541
[2]t.ly/IilK

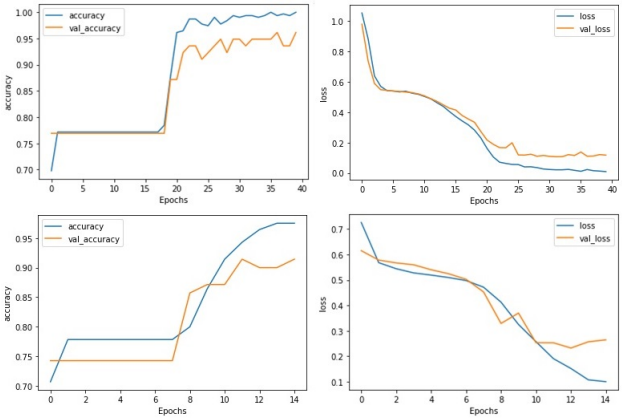

Figure 1: Accuracy and Loss vs Epochs - CNN and BETO model

Another finding was a higher f1-score for functional classification compared to non-functional classification. An analysis presented in [5] suggests this problem can be due to the small number of non-functional requirements and some ambiguity in the requirements specification. Although this fact, we consider this dataset is still representative for those companies with not guidance for specifying NFRs. In general, we found that SVM and TF-IDF (Bigram and Trigram) combinations obtained the highest f1-score for the functional class (0.95) and for the non-functional class (0.79). Leaving aside SVM, LR with BOW, CNN and BETO obtained the best values for functional class (0.93), and NB with BOW (Bigram and Trigram) and LR with BOW (Unigram) for non-functional class (0.76).

3 Conclusions

This study investigates the use of different combinations of ML algorithms and text vectorization techniques for Spanish requirements' classification, revealing that SVM with TF-IDF gives the highest f1-score. The results also show that the use of Bigram in combination with TF-IDF for the Shallow ML algorithms could improve the classification performance. Finally, future research will evaluate whether these models can be generalized, performed a new testing in a dataset with different source.

References

[1] José Cañete, Gabriel Chaperon, Rodrigo Fuentes, Jou-Hui Ho, Hojin Kang, and Jorge Pérez. Spanish pre-trained bert model and evaluation data. In *PML4DC at ICLR 2020*, 2020.

[2] Mita K Dalal and Mukesh A Zaveri. Automatic text classification: a technical review. *International Journal of Computer Applications*, 28(2):37–40, 2011.

[3] Instituto Cervantes. El español una lengua viva, 2021. [Online; accessed 30-November-2021].

[4] Yoon Kim. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on EMNLP*, pages 1746–1751, Doha, Qatar, Oct s2014. ACL.

[5] María-Isabel Limaylla-Lunarejo, Nelly Condori-Fernandez, and Miguel R. Luaces. Towards an automatic requirements classification in a new spanish dataset. (accepted to be published). In *2022 IEEE 30th International Requirements Engineering Conference (RE)*. IEEE, 2022.