Author Profiling and Aggressiveness Detection in Spanish Tweets: MEX-A3T 2018

Mario Ezra Aragón[1] and A. Pastor López-Monroy[2]

[1] Department of Computer Engineering,
Universidad Autónoma de Chihuahua; Chihuahua, Chih., México, 31100.
[2] Department of Computer Science,
University of Houston; Houston TX, USA 77004
p235736@uach.mx, alopezmonroy@uh.edu

Abstract. In this paper we evaluate a set of different well know strategies for text classification at *MEX-A3T: Authorship and Aggressiveness analysis in Twitter case study in Mexican Spanish 2018*. The main idea is to evaluate the performance of the different strategies on Spanish classification tasks that have been successfully for English classification tasks. The strategies that we evaluate are four: 1) a Bag of Terms (BoT), 2) a Second Order Attributes (SOA) representation, 3) a Convolutional Neural Network (CNN) models and 4) a huge ensemble of n-grams at word and character level. The obtained results are in agreement with previous studies that have shown high performance for words and n-grams, and represent a very first attempt to bring these and other ideas, such as distributional representations and neural networks, to the Spanish classification tasks. The strategies presented get a better performance than the average results of other participants, demonstrating significant results for these tasks.

Keywords: Spanish text classification · n-grams · Convolutional Neural Network · SOA · Author Profiling · Aggressiveness.

1 Introduction

In Natural Language Processing (NLP) the Author Profiling (AP) is a well known task, that consist in extracting all the possible demographic information from an author's document. Similarly the detection of Aggressiveness on texts, aims to determine if a text message has an aggressive sense or not. Is an important task as users of different social medias could put their integrity on risk due the high level of violence on this platforms. The scientific community has been interested in both tasks due to the applicability in different problems such as security, prevention, business intelligence, political opinion, etc. The competition MEX-A3T 2018 has the objective of tackling these problems using machine learning for two specific tasks: i) to predict the occupation and location of the users given a group of tweets, and ii) to detect aggressiveness of individual tweets.
In this work we evaluated four general strategies 1) the Bag of Terms (BoT) [6], 2) a Second Order Attributes (SOA) representation [7], 3) Convolutional Neural Networks (CNN) [2] and 4) similar to BoT to get a better performance we used an ensemble of n-grams at word and character level [4]. All of these approaches have been widely

explored to handle various NLP tasks, performs well on classification problems [1, 5, 2] and proved good performance in AP (see PAN2013-2017). The final goal of this work, is to evaluate the individual performance of each one, to devise the best one for texts in Spanish.

2 Evaluated Strategies

2.1 Bag of Terms (BoT)

The BoT is a simple vectorial representation of the text that describes the occurrence of words within a documents, i) first creates a vocabulary of the known words from the training data and then ii) measure the presence of the words by its frequency [6]. In this representation we create an histogram represented by the frequency of the words where the order or the structure are ignored, because the only interest is the occurrence in the document and not the position or order in the document.

2.2 Second Order Attributes (SOA)

Is a representation that build document vectors in a space of profiles, each value in the vector represents the relationship of each document with each target profile and also subprofiles [7]. This with the objective of divide the profiles into several groups using a clustering algorithm. The first step is to capture the relation of each term with the profiles and compute a term vector in a space of profiles, then the next step is to create a term vectors of the terms contained in the documents and is weighted by the relative frequency of the term in the document.

2.3 CNN Representation

We used CNN representations that are based on [2], these are deep neural networks that have been successfully used in NLP tasks. We used three different ways of initialize the weights for the models: 1) CNN-Rand: The model we tested is where all words are randomly initialized and then modified during training. 2) CNN-Static: This model uses word embedding vectors [9] to initialize the embedding layer. The weights of the embedding are kept fixed so they are not modified by training. 3) CNN-NonStatic: This model is same as the previous one, but the embedding weights are allowed to change during training.[3] We used pre-trained word vectors with dimensionality of 300, word vectors were obtained using FastText [8] for Spanish language.

3 Word and Character N-Grams Subspaces

The proposed method has four stages for the creation of the representation: In the first stage we extract groups of n-grams of size one to four at word level and size two to five

[3] For all datasets we used filter windows of size 3,4,5 with 100 feature maps each, dropout rate of 0.5, training is done through stochastic gradient descent over shuffled mini-batches

at character level. The second stage select the best n-grams using chi^2 distribution for each group of n-grams. The third stage then concatenate the best n-grams from each group. In the final stage we used a Support Vector Machine (SVM) for the training and classification. Figure 1 shows the process of extraction, creation and selection of the best n-grams.

Extract n-grams The first step is to create a groups of n-grams [4] of size one to four at word level and two to five at character level. To extract the n-grams i) first the documents are represented using the occurrences of the group of words, ii) then normalize term vectors and iii) smoothing the inverse document frequency weights by adding one to document frequencies, as if every term was seen in an extra document and zero divisions are prevented.

CHI2 distribution The next step is the selection of the best features of each group of n-grams, for this we used the chi^2 distribution X_k^2 [3]. Using this function we eliminate the features that are the most likely to be irrelevant for classification.

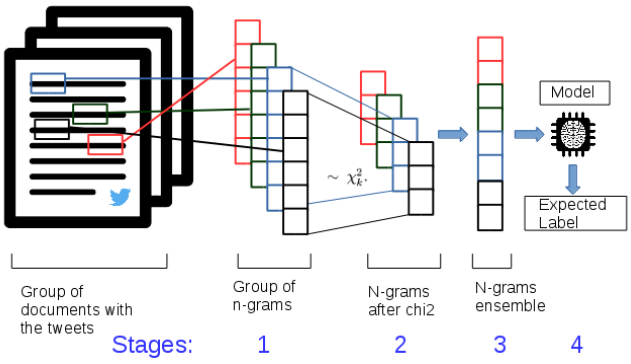

Fig. 1. N-gram Ensemble diagram creation

4 Experimental Settings

The objective for the first task we want to determinate the occupation and the location of a users analyzing the tweets that they post. For the second task is to determinate of a set of different tweets aggressive vs not aggressive.
For the profiling scenario we have two predictions: occupation and location. For the aggressiveness task we have two different classes; non-aggressive or aggressive. We evaluated the different strategies for each task and for the author profiling task we trained two classifiers, one for each prediction. We have 8 classes for the occupation profiling; social, administrative, sciences, arts, sports, student, health and others. For the location

profiling we have 6 classes; north, northwest, northeast, west, center and southeast. In [10] the authors present an overview that describe in detail the tasks, data and evaluation protocols.

5 Experimental Results

We separated the training dataset in 70% for training and 30% for test and evaluate the different strategies, and then we evaluate for the test dataset. For the occupation and location task we obtained the macro F-measure and for the aggressiveness task the F-Measure over the positive class. Table 1 shows the detailed classification results obtained with the different approaches over the different tasks. Results shows better performance by the n-grams ensemble for the three different tasks, for this reason we select this strategy for the test part. SOA and CNN did not perform better in these tasks than the n-grams or BoT. The n-grams subspace and the BoT captures better important words or group of words in the texts than the other strategies. For the CNN it could be due the lack of data and for the SOA a deep search for the creation of subprofiles, but this presents an opportunity for a better performance if we could integrate the same idea as the n-grams. Comparing the final results we can appreciate that our strategy outperform the average results for all participants.

For a deeper analysis of why n-grams performs better than the other methods we get

Table 1. Detailed classification F-Measure

Language	Training Data						Test Data	Average results for all Participants
	BoT	SOA	N-Gram Ensemble	CNN-Rand	CNN-Static	CNN-NonStatic	N-Gram Ensemble	
Aggressiveness	0.7	0.63	**0.74**	0.60	0.61	0.61	0.4312	0.3949
Occupation	0.65	0.66	**0.70**	0.52	0.55	0.53	0.491	0.4657
Location	0.82	0.8	**0.86**	0.54	0.54	0.54	0.8388	0.7618

the best group of n-grams of each task. First we get the best 10 and select five from them. Table 2 shows the best 5 n-grams for the occupation part at word level, and Table 3 also shows the best 5 n-grams at word level. On location we can see that the name of a place is dominant, meaning this location have a lot of users that usually mention the place where they live. For occupation is not as easy as location to distinguish just by the words, as some words just refer to tags or sports. In Table 4 we show the best 5 n-grams at word level for aggressiveness that were obtained using chi^2. We can appreciate a clear selection of words that show aggressiveness, these n-grams give a strong weight in the classification part.

6 Conclusions

In this paper we presented a comparison of different representations for the tasks presented for authorship and aggressiveness analysis over Spanish tweets. We used a BoT, a n-gram representation, a SOA representation, and a CNN model with three different

Table 2. Best n-grams for location at word level

1 word	2 words	3 words	4 words
'aispuro'	'de durango'	'durango https www'	'durango pic twitter com'
'durango'	'durango pic'	'durango user hashtag'	'durango user mentiontwitter user'
'hermosillo'	'en durango'	'mentiontwitter durango https'	'mentiontwitter durango pic twitter'
'notigram'	'los duranguenses'	'mentiontwitter durango pic'	'mentiontwitter durango user hashtag'
'ujed'	'mentiontwitter durango'	'mentiontwitter durango user'	'mentiontwitter durango user mentiontwitter'

Table 3. Best n-grams for occupation at word level

1 word	2 words	3 words	4 words
'caballon'	'compare saludos'	'hashtag mentiontwitter lomejordelsexo'	'hashtag mentiontwitter conocimiento saludos'
$'humbertocota20'$	'juntos difundimos'	'hashtag mentiontwitter masfalsoquetvnotas'	'hashtag mentiontwitter lomejordelsexo es'
'pretemporada'	'mentiontwitter setepasdecirme'	'mentiontwitter pretemporada user'	'user hashtag mentiontwitter lomejordelsexo'
'rumy'	'seguirme juntos'	'mentiontwitter setepasdecirme que'	'user hashtag mentiontwitter masfalsoquetvnotas'
'setepasdecirme'	'setepasdecirme que'	'por seguirme juntos'	'user hashtag mentiontwitter pretemporada'

Table 4. Best n-grams for aggressiveness at word level

1 word	2 words	3 words	4 words
'chinguen'	'chinga tu'	'chinga tu madre'	'de su puta madre'
'hdp'	'de mierda'	'tu puta madre'	'hijo de mil putas'
'pendejo'	'hijo de'	'usuario chinga tu'	'hijo de tu puta'
'putos'	'mil putas'	'de tu puta'	'usuario chinga tu madre'
'usuario'	'tu madre'	'hija de tu'	'que digan esos putos'

way of initialize the weights. We can observe in the overall results of the tasks that the n-grams representation gets the best results in the two different tasks, and we selected it for the test data. When we analyzed the n-grams, we could see the different words that gives weight to the classes, this representation capture important words for the classification especially in the aggressive class where the words shows a clear aggressiveness. Our results show a better performance that the average results of all participants. SOA and CNN did not perform well in these tasks, showing a long way to explore but presents an opportunity to integrate the same idea as the n-grams and look for a better performance.

References

1. Wenpeng Yin, et al: Comparative Study of CNN and RNN for Natural Language Processing. CoRR, (2017)
2. Kim, Y: Convolutional neural networks for sentence classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), (2014)
3. Christian Walck and Fysikum: Hand-book on Statistical Distributions for experimentalists. Internal Report SUFPFY/9601, Stockholm (2007)
4. Daniel Jurafsky and James H. Martin: Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Third Edition draft. Chapter 4, (2014)
5. Marc Moreno Lopez and Jugal Kalita: Deep Learning applied to NLP. CoRR, (2017)
6. Yoav Goldberg: Neural Network Methods in Natural Language Processing (Synthesis Lectures on Human Language Technologies). Graeme Hirst, (2017)

7. Lopez-Monroy, A.P. and Montes-Y-Gomez, M. and Escalante, H.J. and Villasenor-Pineda, L. and Villatoro-Tello, E.: Inaoes participation at pan13: Author profiling task. In: Notebook Papers of CLEF 2013 LABs and Workshops, Valencia, Spain, September (2013)
8. Armand Joulin and Edouard Grave and Piotr Bojanowski and Tomas Mikolovn: Bag of Tricks for Efficient Text Classification, CoRR (2016)
9. Tomas Mikolov and Kai Chen and Greg Corrado and Jeffrey Dean: Efficient Estimation of Word Representations in Vector Space, CoRR (2013)
10. Álvarez-Carmona, Miguel Á and Guzmán-Falcón, Estefanía and Montes-y-Gómez, Manuel and Escalante, Hugo Jair and Villaseñor-Pineda, Luis and Reyes-Meza, Verónica and Rico-Sulayes, Antonio: Overview of MEX-A3T at IberEval 2018: Authorship and aggressiveness analysis in Mexican Spanish tweets, Notebook Papers of 3rd SEPLN Workshop on Evaluation of Human Language Technologies for Iberian Languages (IBEREVAL), Seville, Spain, September (2018)