Tweets Classification Using Corpus Dependent Tags, Character and POS N-grams

Notebook for PAN at CLEF 2015

Carlos E. González-Gallardo, Azucena Montes, Gerardo Sierra,
J. Antonio Núñez-Juárez, Adolfo Jonathan Salinas-López, Juan Ek

Grupo de Ingeniería Lingüística, Instituto de Ingeniería, UNAM, México

{cgonzalezg,amontesr,gsierram,anunezj,asalinasl,jekc}
@iingen.unam.mx

Abstract. This paper is part of the Author Profiling task at PAN 2015 contest; in witch participants had to predict the gender, age and personality traits of Twitter users in four different languages (Spanish, English, Italian and Dutch). Our approach takes into account stylistic features represented by character N-grams and POS N-grams to classify tweets. The main idea of using character N-grams is to extract as much information as possible that is encoded inside the tweet (emoticons, character flooding, use of capital letters, etc.). POS N-grams were obtained using Freeling and certain token were relabeled with Twitter dependent tags. Obtained results were very satisfactory; our global ranking score was of 83.46%.

1 Introduction

Author Profiling focused on Internet texts has been growing for the last years, one of the reasons is the big amount of information produced every minute in social networks or blogs. These Internet texts have their own characteristics that are hardly comparable with literary texts, documentaries or essays; this is because of the necessity of having a quick communication and the liberty of publishing unrevised content. For March 2015, Facebook reported having about 936 million daily active users on average [1].

As part of the PAN 2015 contest, this year, the Author Profiling (AP) task was about tweets in Spanish, English, Italian and Dutch [2]; for Spanish and English, the objective was to predict gender, age and personality traits of a Twitter [3] user. Moreover, for Italian and Dutch was only needed to predict gender and personality traits.

Twitter has its own rules and characteristics that users explode to express themselves and communicate to each other. These rules can be extracted to create dependent tags *(corpus dependent tags)* that will help the classifier to improve its performance.

2 Dataset

The dataset provided this year consisted of tweets in Spanish, English Italian and Dutch. Regard to gender, the corpus was balanced in all four languages (50% of tweets were label as "female" and the other half as "men").

Table 1. Female and male distribution of the corpus

Language	Female		Male		Total samples
	Samples	Percentage	Samples	Percentage	
Spanish	50	50%	50	50%	100
English	76	50%	76	50%	152
Italian	19	50%	19	50%	38
Dutch	17	50%	17	50%	34

For the case of Spanish and English, age classes were defined in four groups (18-24, 25-34, 35-49 and 50-xx). In this case the corpus was not balanced, having a lot of "25-34" samples (around 40%) and just a few samples for "50-xx" (around 10%).

Table 2. Proportion of age-group samples for Spanish and English

		Spanish	English
18-24	Samples	22	58
	Percentage	22%	38%
25-34	Samples	46	60
	Percentage	46%	40%
35-49	Samples	22	22
	Percentage	22%	14%
50-xx	Samples	10	12
	Percentage	10%	8%
	Total samples	100	152

There were five personality traits to predict: extroverted, stable, open, conscientious and agreeable; each one of them with a possible value between -0.5 and +0.5. It is important to mention that the samples for personality traits were totally imbalanced. For example: in Italian, for the conscientious personality trait there were just 5 labels of the 11 possible ones (-0.5, -0.4, … , +0.4, +0.5), and the number of samples of these existing labels varied a lot.

Table 3. Number of samples per label in each personality trait

		-0.5	-0.4	-0.3	-0.2	-0.1	0.0	0.1	0.2	0.3	0.4	0.5
Spanish	Extroverted			3		5	5	28	32	9	9	9
	Stable			2	10	26	9	12	19	10	10	2
	Agreeable				3	16	6	16	40	12	2	5
	Conscientious				2		21	7	20	12	21	17
	Open					7	10	37	15	9	14	8
English	Extroverted			1	4	10	17	41	37	20	13	9
	Stable			11	5	22	9	19	37	19	18	12
	Agreeable			5	2	12	19	44	46	13	7	4
	Conscientious				1	4	30	38	27	33	12	7
	Open					2	1	47	39	23	19	21
Italian	Extroverted						8	13	9		3	5
	Stable				1	3	3	8	4	12	5	2
	Agreeable					1	3	11	9	7		7
	Conscientious						3	18	6	5	6	
	Open						1	14	9	2	7	5
Dutch	Extroverted						3	5	11	7	6	2
	Stable				1	5	3	3	4	6	8	4
	Agreeable				2	1	5	10	10	2	4	
	Conscientious					2	4	15	6	5	2	
	Open							4	11	4	12	3

3 Features

One of the main characteristics of the AP task for PAN 2015 was that it was not about classifying tweets but **classifying Twitter users based on a group of their tweets**; let us call each one of these groups a *document*. So based on this fact, the vectors for the training algorithm and the vector for each one of the tests were a *document-vector* formed by a group of individual tweets; considering this, if a tweet can have a maximum length of 140 characters and in average each *document* was about 100 tweets, the length of each *document* could be about 14000 characters; a length quite acceptable for extracting a good number of features [4].

Because of the nature of the task that involved four different languages and our idea of making the algorithm the most language independent as possible, it was going to be difficult and unpractical to use content-based features; so we opted for the stylistic features and just keep it simple. It is possible to divide the used features in two groups: character N-grams and POS N-grams.

Using character N-grams could be seen very basic and naive, but it has shown to be very useful and practical in previous experiments [5]. Besides the usefulness shown in

the past by the character N-grams, implicitly a great amount of stylistic features are extracted; for example: if 3-grams are used with in a *document* the frequency of all punctuation marks, characters flooding (!!!, ???, …, etc.), word flexion and derivation, diminutives, superlatives and prefixes are being extracted [6,7].

Regard to the POS N-grams, it is possible to obtain the grammatical sequence of the writer. For the POS tagging we used Freeling [8] with its corresponding configuration for Spanish, English and Italian; for Dutch, Freeling doesn't have a module so in this case we set up a basic assumption: **if we use the English Freeling module, errors are going to be performed; but if these errors follow a certain stable pattern, it is possible that some grammatical information is being extracted from the tweets.**

Depending of the input language to analyze, our software selects the best configuration of *extraction parameters* (based on a series of tests) that maximize its own performance. These configurations take into account the number of N in the character N-grams (*num_gramas*) and POS N-grams (*num_POS*), if they are retroactive or not (*retro_gramas* and *retro_POS*), the N-gram representation (*modo*) and if the N-grams should be represented in a logarithmic scale (*frec_log*).

Table 4. *Extraction parameters* and its possible values

PARAMETER	VALUES
num_gramas	$0 \leq n$
num_POS	$0 \leq n$
retro_gramas	0 →NO 1 → YES
retro_POS	0 →NO 1 → YES
modo	frec → frequency mode bin → binary mode
frec_log	0 →NO 1 → YES

The best configuration established is shown in the next table.

Table 5. Best configuration of the *extraction parameters (GA: gender and age, P: personality traits)*

		num_gramas	num_POS	retro_gramas	retro_pos	modo	frec_log
Spanish	GA	3	3	1	1	frec	1
	P	3	3	1	1	frec	0
English	GA	2	3	1	1	frec	1
	P	3	3	1	1	frec	1

Italian	GA	3	1	1	1	frec	1
	P	3	1	1	1	frec	1
Dutch	GA	3	1	1	1	frec	1
	P	3	1	1	1	frec	1

4 Algorithm

As mentioned before, the objective of the task was to predict gender, age and personality traits of a Twitter user in four different languages: Spanish, English, Italian and Dutch (for these last two languages was only needed to predict gender and personality traits). For the case of gender and age it is obvious that the type of learning algorithm to implement is a classifier. By the other hand, given the nature of the personality traits values that can be considered continue real values it is possible to think in a regression problem. This is a good approach, but considering the few samples of certain points (values) for the regression in some of the personality traits and the bad performance of some regression algorithms we used to check the behavior of the corpus, we decided to make the prediction of personality traits problem a classification task.

For Spanish and English the goal was to predict age and gender of a Twitter user, so we decided to classify both characteristics at the same time getting the next 8 possible classes: **_F_20s, _F_30s, _F_40s, _F_50s, _M_20s, _M_30s, _M_40s and _M_50s**. Age groups are explained in the next table. For Italian and Dutch two classes where just created: **_F** and **_M**.

Table 6. Age groups for Spanish and English

Age group	Age range
20s	[18,25)
30s	[25,35)
40s	[35,50)
50s	[50, +oo)

The training phase is divided in 5 entities: extraction, labeled, POS generation, vectors creation and training.

- **Extraction**: The truth file (truth.txt) is read and analyzed. One file is created for each one of the possible classes: *class file*. The tweets are preprocessed based on the substitution rules showed in Table 7; hashtags are not preprocessed because we consider they can provide stylistic information.

Table 7. Substitution rules

Token	Token explanation	Substitution
@username	Reference to another Twitter user	@us
http[s]://…	Link to an external site	htt
\n	New line character	*Space character*

Note: Each one of these *class files* is separated by *documents* (group of tweets of an author).

- **Labeled**: For each *class file* created by the extraction phase, a local instance of Freeling is called to obtain a JSON [9] list[1] that contains each one of the tweets of each one of the *documents*.

- **POS generation**: Once the JSON list is obtained, the POS generation phase creates a *POS file* with the same structure of the *class file* and relabels certain tokens (adding our own tags) that are needed to extract extra grammatical information.

Table 8. *Corpus dependent tags*

Token	Label
@us	REF#USERNAME
htt	REF#LINK
#{something}	REF#HASHTAG

- **Vectors creation**: Character N-grams and POS N-grams are extracted from each *document* from each *class file* and *POS file* respectively based on the *extraction parameters* (Table 5.) to produce the *document-vectors*. Once all the *document-vectors* are created, a general *features-vector* is generated and each *document-vector* is expanded to the *features-vector* length obtaining the *features-matrix* that will be used to train the system.
- **Training**: The *features-matrix* is now passed to the learning algorithm to train the system. The algorithm we used to classify age, gender and personality traits is an implementation of a Support Vector Machine (SVM) with a linear kernel called LinearSVC [10]. Once the system has been trained, the learning model and the *features-vector* is serialized and saved into disk for their later use.

The test phase is also divided in 5 entities: extraction, labeled, POS generation, vector creation and testing.

[1] The Freeling instance is called using an interface that converts the output of Freeling into a JSON string. This interface was developed by Grupo de Ingeniería Lingüística, Instituto de Ingeniería, UNAM

- **Extraction**: The xml file of the Twitter user is read and processed based on the substitution rules mentioned in (Table 7.) obtaining a preprocessed text file; hashtags are not preprocessed because we consider they can provide stylistic information.

- **Labeled**: The preprocessed file done by the extraction phase is passed to a local instance of Freeling to obtain a JSON string.

- **POS generation**: Once the JSON string is obtained, the POS generation phase creates a *POS file* with the same structure of the preprocessed file and relabels certain tokens (adding our own tags) that are needed to extract extra grammatical information (Table 8.).

- **Vector creation**: Character N-grams and POS N-grams are extracted from the preprocessed file and the *POS file* respectively based on the *extraction parameters* (Table 4.) to produce a *document-vector*. Then the *features-vector* is loaded to expand the *document-vector*.

- **Testing**: The learning model created by the training phase is loaded so that the *document-vector* can be tested. Once all 6 classifications (gender/gender_age, extroverted, stable, agreeable, conscientious and open) are done the output xml file is created and written to disk.

5 Results

Two measures were used to evaluate the submissions: accuracy was used to measure age and gender, leaving Root Mean Squared Error (RMSE) to measure personality traits. For the case of Spanish and English, an average between *age* and *gender* was performed (*both*).

Table 9. Global result of our approach

gonzalesgallardo15													
	Global Results												
	Personality												
language	agreeable	conscientious	extroverted	open	stable	RMSE	age	gender	both	GLOBAL $\left(\frac{(1-RMSE)+\frac{gender+age}{2}}{2}\right)$	runtime	place
Spanish	0.1168	0.1709	0.1406	0.1398	0.2094	0.1555	0.7273	0.8977	0.7045	0.7745	00:04:25	3
English	0.1480	0.1101	0.1303	0.1422	0.2151	0.1491	0.7817	0.8521	0.6972	0.7740	00:06:29	2
Italian	0.0745	0.1269	0.0764	0.1572	0.2121	0.1294	-	0.8611	-	0.8658	00:01:31	1
Dutch	0.0952	0.1299	0.0901	0.0637	0.0661	0.0890	-	0.9375	-	0.9242	00:01:20	2
										0.8346		

If we compare our approach against the other two best participants (tables 7.1-7.4), ours is slightly slower because of the grammatical analysis made by Freeling.

Each one of the corpus languages had different number of samples and different distribution of data, so maybe it is not objective to compute a global average but we think is important to do this because the task involved analyzing all four languages; so we took the freedom of computing it obtaining a global score of 83.46%.

Some extra information related to our results is presented in Section 8. Extended Results.

6 Conclusions

Internet has made of communication something very quickly and fluid; example of these is the social network Twitter, in which users have to transmit a complete message in just 140 characters. To accomplish this, it is necessary to compact as much information as possible, making each one of the tweets a *dense text* (short text with a lot of information).

The use of N-grams of characters and POS N-grams, as shown in the results, is a good option with *dense texts* because of theirs extraction capacity. In the case of N-grams it was possible to extract emoticons, exaggeration of punctuation marks (character flooding), use of capital letters and all kind of emotional information encoded in the tweet. With POS N-grams, in Spanish and English we were able to capture the most representative series of two and three grammatical elements; in Italian and Dutch we were able to capture the most frequent grammatical elements.

7 Future Work

Our approach showed to be good for the gender classification task but not too good for age classification. We will focus in finding some characteristics that we are probably missing and try to give them more emphasis to make them more representative. Probably is a good idea to separate the classification and just classify age and gender separately.

8 Extended Results

Table 10. PAN 2015 Author Profiling results for Spanish

place	user	Personality						age	gender	both	GLOBAL	runtime
		agreeable	conscientious	extroverted	open	stable	RMSE					
1	alvarezcarmona15	0.1113	0.1168	0.1319	0.1257	0.1631	0.1297	0.7955	0.9659	0.7727	0.8215	00:00:44
2	kiprov15	0.1249	0.1386	0.1625	0.1334	0.1884	0.1495	0.7841	0.9091	0.7273	0.7889	00:02:46
3	gonzalesgallardo15	0.1168	0.1709	0.1406	0.1398	0.2094	0.1555	0.7273	0.8977	0.7045	0.7745	00:04:25

Table 11. PAN 2015 Author Profiling results for English

place	user	Personality						age	gender	both	GLOBAL	runtime
		agreeable	conscientious	extroverted	open	stable	RMSE					
1	alvarezcarmona15	0.1305	0.1172	0.1278	0.1202	0.2253	0.1442	0.8380	0.8592	0.7254	0.7906	00:00:59
2	gonzalesgallardo15	0.1480	0.1101	0.1303	0.1422	0.2151	0.1491	0.7817	0.8521	0.6972	0.7740	00:06:29
3	teisseyre15	0.1480	0.1309	0.1371	0.1351	0.1990	0.1500	0.7535	0.8310	0.6479	0.7489	00:03:15

Table 12. PAN 2015 Author Profiling results for Italian

place	user	Personality						age	gender	both	GLOBAL	runtime
		agreeable	conscientious	extroverted	open	stable	RMSE					
1	gonzalesgallardo15	0.0745	0.1269	0.0764	0.1572	0.2121	0.1294	-	0.8611	-	0.8658	00:01:31
2	grivas15	0.1389	0.2461	0.1350	0.1586	0.1930	0.1743	-	0.8333	-	0.8295	00:00:29
3	kocher15	0.1302	0.1093	0.1000	0.1344	0.1555	0.1259	-	0.7778	-	0.8260	00:00:01

Table 13. PAN 2015 Author Profiling results for Dutch

place	user	Personality						age	gender	both	GLOBAL	runtime
		agreeable	conscientious	extroverted	open	stable	RMSE					
1	alvarezcarmona15	0.0000	0.1075	0.0750	0.0354	0.0637	0.0563	-	0.9375	-	0.9406	00:00:24
2	gonzalesgallardo15	0.0952	0.1299	0.0901	0.0637	0.0661	0.0890	-	0.9375	-	0.9242	00:01:20
3	grivas15	0.1427	0.2278	0.1467	0.0973	0.1711	0.1571	-	0.9688	-	0.9058	00:00:29

Acknowledgements: This work was funded by project CONACyT-México No. 215179 "Caracterización de huellas textuales para el análisis forense".

9 References

1. Facebook. «Facebook newsroom», http://newsroom.fb.com/company-info
2. Rangel, F., Rosso, P., Potthast, M., Stein, B., Daelemans, W.: Overview of the 3rd author profiling task at pan 2015. In: Cappellato L., Ferro N., Gareth J. and San Juan E. (Eds). (Eds.) CLEF 2015 Labs and Workshops, Notebook Papers. CEUR-WS.org, (2015)
3. Twitter, Inc., https://twitter.com
4. Schwartz, R., Tsur, O., Rappoport, A., & Koppel, M. (2013). Authorship Attribution of Micro-Messages. In EMNLP (pp. 1880-1891).
5. Doyle, J., & Kešelj, V. (2005). Automatic Categorization of Author Gender via N-Gram Analysis. In The 6th Symposium on Natural Language Processing, SNLP.

6. Stamatatos, E. (2006, August). Ensemble-based author identification using character n-grams. In Proceedings of the 3rd International Workshop on Text-based Information Retrieval (pp. 41-46).
7. Stamatatos, E. (2009). A survey of modern authorship attribution methods. Journal of the American Society for information Science and Technology, 60(3), 538-556.
8. Freeling, http://nlp.lsi.upc.edu/freeling/
9. JSON (JavaScript Object Notation), http://JSON.org/
10. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research*, *12*, 2825-2830.