INSA LYON and UNI PASSAU's participation at PAN@CLEF'17: Author Profiling task

Notebook for PAN at CLEF 2017

Guillaume Kheng, Léa Laporte, and Michael Granitzer

Institut National des Sciences Appliquées Lyon and Universität Passau
guillaume.kheng@gmail.com, lea.laporte@insa-lyon.fr, michael.granitzer@uni-passau.de

Abstract This paper describes the participation of INSA Lyon and UNI Passau at the PAN 2017 Author Profiling task. Given the language and tweets from an author, the goal is to predict his/her gender and language variety. We consider two strategies : a "loose" classification that learns one predictive model for the gender and another one for the variety, and a "successive" classification that first predict the gender then learn a predictive model for variety, given the gender. We consider all the languages. We experiment various features representations and machine learning algorithms used in previous PAN Author Profiling editions in order to learn the models. We adapt the features and machine learning algorithm used for each language and each classification task by selecting the configuration that provides the best results in terms of prediction performance.

1 Introduction

Thanks to the expansion of social networks and the progress of Internet and the related technologies, social media are now part of our daily life. A large amount of content, especially textual content, is thus produced and read every day, but without any certitude about the real identity of the author. Indeed, on the internet, people can easily hide their identity, even lie about it or usurp someone else's. One may want to know who authored a given content, or simply profile the content author in order to know to know more about him/her. Author Profiling (AP) is a text forensics field which try to tackle this latest issue. AP studies aim at retrieving some characteristics of an author (e.g. his/her age, gender, personality, etc) by analyzing only the texts he/she writes. Multiple applications of Author Profiling exist in various fields [9]. Indeed AP could help investigators profile criminals and use written content as evidence (forensics) or prevent malicious behavior on social network by profiling criminals (security). On the other hand, profiling the users of a product for a company could improve its consumer segmentation and yield a more accurate advertising campaign (marketing).

The CLEF PAN track have been offering an Author Profiling task for the last 5 years [6]. The task settings differ each year in terms of text genres, languages and author's characteristics. A summary of the previous AP editions' settings is provided in Table 1. As shown in this table, the aim of the 2017 PAN Author Profiling task is to retrieve the gender of the author and his/her language variety i.e. the "specific variation of his/her language", due to the geographical location he/she comes from. Gender and language variety predictions can be considered as two subtasks of the PAN 2017 Author Profiling Task. Participants are given the choice to participate to the both subtasks or only one of them. Furthermore, they can consider all languages or only a subset. In our approach, we choose to participate to the both subtasks, for all the languages.

Table 1: Evolution of PAN AP task settings 2013-2017

		2013	2014	2015	2016	2017
Text Genre	Blogs	X	X		X	
	Reviews		X		X	
	Social Media		X		X	
	Tweets		X	X	X	X
Authors Features	Age	X	X	X	X	
	Gender	X	X	X	X	X
	Language Variety					X
	Personnality			X		
Languages	Arabic					X
	Dutch			X	X	
	English	X	X	X	X	X
	Italian			X		
	Portuguese					X
	Spanish	X	X	X	X	X

One core aspect of our approach was to analyse the impact of different combinations of feature representation techniques and classification algorithms in terms of classification accuracy. We implemented various features extraction techniques such as n-grams, TF-IDF and LSA, and various machine learning algorithms, including Support Vector Machines, Naïves bayes and Random Forests.
We also wanted to study the dependency between gender and variety. To do so we performed a "gender then variety" successive classification and compare it to a loose classification. Successive classification starts by predicting the gender only, then uses the results of this classification to predict the language variety. On the other hand, loose classification predicts the gender and variety labels independently.

This paper is structured as follows. In section 2, we present our proposed approach. In section 3, we detail the experimental protocol and settings. Section 4 present the the results. In section 5, we discusses the perspectives and conclude this paper.

2 Overview of our proposed approach

Our approach consists in 3 main steps: preprocessing, feature extraction and learning using a machine learning algorithm. In a first part, we present the preprocessing step. Then we introduce the different features we consider in order to achieve the best classification possible. Finally, we detail the learning step, including the machine learning algorithm we used in the context of the task and the successive learning strategy that we tried.

2.1 Preprocessing

Several approaches have been proposed in the literature in order to process tweets for further information extraction [7]. Based on an analysis of previous PAN Author profiling editions, we choose to consider the following preprocesses:

Removal of short tweets: we remove all tweets with letter count below 10 characters (special characters included)

Removal of the @user: On Twitter, '@' are used to adress twitter users. When a user start a discussion with another '@user', this specific Twitter id will appear multiple times, which might cause over-fitting, so we remove it.

Removal of URLs: URLs might be too user-specific (causing over-fitting) and they might enrich the vocabulary too much for a poor gain in information as they are likely to occur only once.

Lowercase of #hashtags' body: People might use the same hash-tag with different repartition of the upper/lowercase letters. For instance, without this technique, *#AuthorProfilingRocks* and *#authorProfilingROCKS* supposedly mean the same but will end up as different "words".

Removal of stop words: we made optional the removal of stop words as tweets are really short messages and some of the stop words might carry more meaning than in a longer text.

The corpus we obtain after the preprocessing step is described in table 2. One can also notice that the number of tweets removed never amounts to more than 2% of each language sub-corpus.

Table 2: Characteristics of the corpus before and after preprocessing

language	Arabic	English	Spanish	Portuguese
number of tweets in the initial dataset	240,000	360,000	420,000	120,000
number of "empty" tweets (len <10 chars)	4,219	1,555	1,910	1,895
number of tweets post processing	235,781	358,445	418,090	118,105

2.2 Features Extraction

Once the corpus has been preprocessed, we extract features from the corpus and use them to represent the tweets. Most of the representations we considered have been used by the winners of the previous PAN AP tasks editions [8,7,9]. We consider there representation of tweets based on n-grams, TD-IDF and LSA. We also planned to consider stylometric features, but we finally do not implement them for technical reasons.

N-grams models. These models consists in establishing a vocabulary for the documents based on sequences of n items (characters, words, associations of words (for n > 1), POS tags) extracted from text. Then, the features are the frequencies of the n-grams in the vocabulary. N-gram-based features have proven to be highly useful indicators of various linguistic differences between authors [13].
We implemented features with unigrams, bigrams and trigrams at the word level.

TF-IDF. Text Frequency-Inverse Document Frequency (TF-IDF) is a well-established technique in Information Retrieval. TF-IDF computes the apparition scores of each word by highlighting those appearing a lot in few documents which helps the learning algorithm selecting words with high discriminative power between labels. This approach have been widely used for the Author Profiling task.

Latent Semantic Analysis (LSA). LSA allows to capture semantic relations between groups of words as described in [5]. It produces a set of concepts linking words to documents and by extensions to profiles. LSA also proceeds into a dimensionality reduction which in our specific case is quite useful given the potential huge size of the vocabulary. The features provided by this technique allow us to train classifiers with deeper understanding of the tweets content. In 2015, the PAN AP task winners [7] yielded the top results with this approach.

Stylometry. We planned to consider features related to the Natural Language Processing field, including Part-Of-Speech features and Stylometric features (average word count per sentence, average number of letters per word, etc), that have been shown to perform well for this task [4]. Those features can be taken alone or unified with other types of features in order to carry more information and potentially achieve a better classification. Unfortunately, given the time constraint and some technical issues, we haven't been able to implement them.

2.3 Machine learning algorithms

Our goal was to reproduce the state of the art approaches from the previous editions of PAN, more precisely, the classification techniques they considered. According to the PAN overviews [PAN16,15,14], SVM, Naive Bayes classifier and Random Forest are the most common learning techniques used in the context of the PAN AP task. Fortunately, they are also the ones achieving the best results on the AP task. We chose to implement these 3 classifiers, ie. Support Vector Machines, Naive Bayes classifer and Random Forest, in order to compare their respective results and pick the best one. As they are well-known machine learning algorithms [12,10,2], we do not describe them in details in this paper. In the case of SVM and Naives Bayes Classifier, we only indicate which variants we used.

For the Naives Bayes classifier, as the official sklearn documentation suggest that the "Multinomial" Naive Bayes classifier (MNBC) works well with TF-IDF, we chose to implement this variant. This method is the fastest in terms of training and classifying on the provided data, so we used it in order to achieve a stable basis for our software at the early stage of development.

Regarding SVM, we tested the kernel and linear approaches. However, the linear mode kept yielding significantly better results than the kernel one during the evaluating and comparing phases. As a consequence, we only use binary linear SVM to predict the gender and multiclass SVM with "one-vs-rest" strategy to predict the language.

2.4 Successive and loose classification

As mentioned in the introdcution, we wondered if Gender and Language variety were not linked in some way. Our assumption was that predicting one label and using the results of this classification in order to predict the second label may achieve good results. We chose to call this type of classification : **"successive classification"** as opposed to **"loose classification"** in which one classifies each label regardless of the others (thus predicting gender and variety are two independent subtasks). The protocol for each type of classification is as follows.

For **loose classification**, we consider the gender and variety prediction as two independent subtask and train the corresponding models separately. We thus train a classifier to predict gender (respectively variety) on the whole language corpus, then we use the learned model to predict gender (respectively variety) for each author within the test dataset.

For **successive classification** , in the context of the task, two strategies could be considered: first predict the gender, then predict the variety given the knowledge of the gender ("gender-then-variety" strategy); or, first predict the variety, then predict the gender given the variety ("variety-the-gender" strategy). In this work, we consider only the "gender-then-variety" strategy for successive classification. We did not consider the "variety then gender" successive classification because the number of tweets available for training would have been significantly reduced and would have likely induced overfitting resulting in poor classification rates. In order to achieve "gender then variety" successive classification for each language corpus we proceed as follow :

1. We train a classifier to predict gender on the whole language corpus.
2. We split each language corpus in 2 sub-corpus, based on the ground truth: one for the female authors and another for the male authors.
3. On each sub-corpus we train a classifier to predict variety. This provides us with a male-variety classifier and a female-variety classifier.
4. We classify each author contained within the test-dataset on gender first and sort predicted males predicted and females authors into 2 sub-test-dataset.
5. We classify each author contained within the sub-test-datasets with the associated variety classifier i.e. the female-variety classifier predicts the variety labels for the authors classified as female, idem for the males.

Figures 1 and 2 show the processing of the test dataset with the loose and successive classification procedures respectively.

To simplify the notations we will call "classification units" the classifiers used to predict gender or variety labels. On figures 1 and 2, one can notice that the number of classifying units for each type of classification is different. Indeed, the "gender then

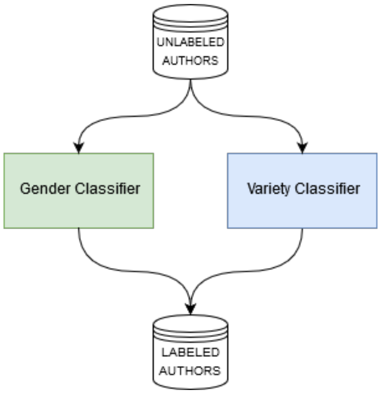

Figure 1: Prediction work-flow of the loose classification

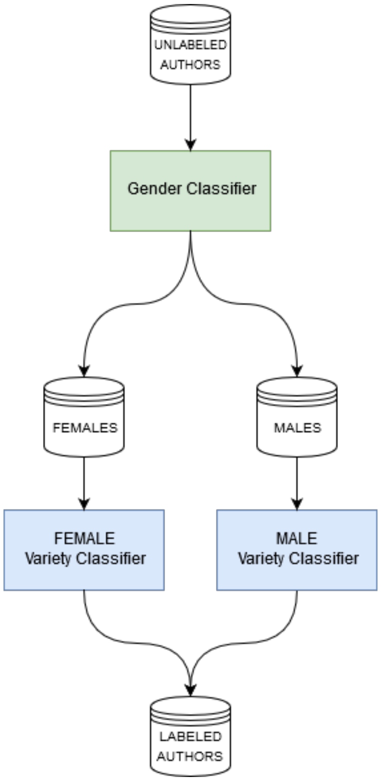

Figure 2: Prediction work-flow of the successive classification

variety" successive classification needs 3 classification units per language (one for the gender, then two variety classifier depending on the gender), whereas the loose classification only needs 2 classification units (one per classification subtask). For each classification unit, we have implemented all the combinations of the sets of features described in section 2.2 and the learning algorithms selected in section 2.3. For each classification unit, we then select the best combination possible according to a given evaluation measure: the F-score which will be detailed in the next section.

3 Experimental Evaluation

In this section, we present the experimental evaluation and results we obtained in the context of the 2017 PAN Author Profiling task edition. First, we present briefly the software implementation. Second, we describe the experimental protocols we followed through the task in order to obtain reliable results. Then, we disclose the different classification units we selected for the loose and the successive classifications, along with the results obtained on the test set provided by the task chairs for each classification type. Finally, we present the results of our submitted final run.

3.1 Software implementation

The software implementation relies on the Python 3 sklearn and NLTK modules.The classifiers and feature extraction tools we used were pre-implemented and documented in the sklearn module. NLTK offered some powerful tools regarding tweet-tokenisation and stop words removal. The source files are available on the following github repo : github.com/SunTasked/profiler

3.2 Experimental Protocol

Regarding the training and evaluation of each classification unit, we respected a certain set of rules in order to achieve reliable results.

Training of classifiers. We tested and optimized all classifiers described in subsection 2.3 using different sets of features: unigrams and bigrams at word level, TF-IDF based on unigrams, bigrams and trigrams at word level, LSA, and a combination of LSA and TF-IDF on unigrams and bigrams at word level. By optimizing, we mean tuning the classifier and feature extractor parameters to achieve the best score possible given this configuration. In order to do so, we used the sklearn "gridsearch" tool which allow you to try different combinations of parameters in a multi-threaded context. In addition, We also tested some combinations of features with and without the stop words removal step. We wanted to detect if these textual element had an impact in the classification process. This represent roughly 24 models trained for each classification unit summing up to a total of 480 models trained, optimized and cross-validated using 10 folds of the training data.

Evaluation measures. We use the micro-averaged and macro-averaged f-measures as the evaluation measures, since we have a corpus with a balanced distribution over the different labels, as recommended by [11]. When comparing one approach to another, we consider the macro measure first and in case of conflicts, we then consider the micro measure. If two configurations lead to same performance in terms of evaluation measure, we use the features which consume less computation power.

3.3 Selected best configurations for the classification units

Regarding the training of the classifiers, we based our approach on a "single tweet" classification. We trained each classifier on tweets as standalone documents i.e. disregarding the fact that each tweet belong to an author along with 99 other tweets.

Then, when proceeding to labels predictions, we classified each of the 100 tweets available for each author separately. Consequently for each author, we obtained 100 labels predictions. In order to obtain the author labels, we summed up the labels predictions and chose the label with the highest score.

In this subsection, we start by presenting the selected models for the gender classification units, as those are the same for loose classification and successive classification. Then we present the selected models for the language variety classification units for the loose and the successive classification.

Gender classification. As shown in table 3, in all languages the best models for gender classification have been obtained by combining TF-IDF features on unigrams and bigrams and a Naive Bayes Classifier (NBC). Surprisingly, the classification over Latin languages seemed to work better when the stop-words were not being removed.

Table 3: Best configurations for Gender classification; for all languages. These configurations have been used for loose and successive classification.

Language	Preprocessing	Features	Classifier	F macro	F micro
Arabic	removal of stop words	TF-IDF (1/2-grams)	NBC	0.707	0.708
English	removal of stop words	TF-IDF (1/2-grams)	NBC	0.669	0.669
Spanish		TF-IDF (1/2-grams)	NBC	0.659	0.661
Portuguese		TF-IDF (1/2-grams)	NBC	0.659	0.663

Variety classification. Table 4 describes the best approaches selected for loose variety classification while table 5 describes the best approaches for "gender then variety" successive classification. We observe that the loose classification units for variety prediction yield better overall accuracy scores than the corresponding successive classification units. The English and Spanish classifiers are particularly affected by the division of the Corpus. The halving of the corpus on gender implied a shrinking of the extracted features set. Consequently, classifying units might have less features to discriminate the tweets on and offer poor prediction performances. On the other hand, the Arabic and Portuguese classifying units seem to yield quite equivalent scores in both loose and successive classification contexts.

Table 4: Best configurations for Variety loose classification unit, for all languages

Language	Preprocessing	Features	Classifier	F macro	F micro
Arabic		TF-IDF (1/2-grams) & LSA	SVM	0.684	0.684
English	rm stop words	TF-IDF(1/2-grams)	SVM	0.669	0.669
Spanish		TF-IDF (1/2-grams) & LSA	SVM	0.684	0.684
Portuguese		TF-IDF (uni-, bi- and tri-grams)	SVM	0.879	0.879

3.4 PAN'17 Results

In the context of the PAN'17 AP task, we decided to submit the loose classification approach. Indeed, as we saw in the experimental results section, the loose classification approach yields the best overall results in terms of variety prediction accuracy. It is particularly noticeable when we consider English and Spanish variety prediction. Moreover one of the main issues regarding "gender then variety" successive classification is that one must achieve a high quality classification on gender. Unfortunately, the results

Table 5: Best configurations for Variety successive classification units, for all language

Gender	Language	Preprocessing	Features	Classifier	F macro	F micro
Female	Arabic		TF-IDF (1/2-grams) & LSA	SVM	0.673	0.674
	English	rm stop words	TF-IDF (1/2-grams)	SVM	0.466	0.467
	Spanish		TF-IDF (1/2-grams) & LSA	SVM	0.518	0.52
	Portuguese		TF-IDF (1/2-grams) & LSA	SVM	0.88	0.88
Male	Arabic		TF-IDF (1/2-grams) & LSA	SVM	0.687	0.687
	English	rm stop words	TF-IDF (1/2-grams)	NBB	0.450	0.449
	Spanish		TF-IDF (1/2-grams)	SVM	0.555	0.556
	Portuguese		TF-IDF (1/2/3-grams)	SVM	0.859	0.859

we obtained on this particular label were not very promising. The official results we obtained are exposed in table 6.

Table 6: Official results for the PAN'17 Author Profiling task

Language	**Feature**	**Score obtained**
Arabic	Gender	0.6856
	Variety	0.7544
	Joint	0.5475
English	Gender	0.7546
	Variety	0.7588
	Joint	0.5704
Spanish	Gender	0.6968
	Variety	0.9168
	Joint	0.6400
Portuguese	Gender	0.6638
	Variety	0.9750
	Joint	0.6475

Although we don't have the results of the other participants yet, the results regarding gender classification are not as high as we expected when we consider the result achieved in the previous PAN editions [8,7]. One could justify such gap by the fact that we are lacking some preprocessing steps (POS tags).

In the contrary the quality of the classifiers in terms of variety prediction in Portuguese and Spanish are quite high as they achieved respectively 97,5% and 91,68% of accuracy.

4 Discussion and Perspectives

In this PAN edition we wanted to implement a Multi Layered Perceptron as learning algorithm and compare to the the other classification approaches. Indeed, the use of SVM as a learning algorithm in the PAN literature represents more than 50% of the approaches as opposed to Neural networks which are almost non-existent in that same context. However, according to [3], using a MLP properly tuned can outperform a SVM in such task. Unfortunately, technical issues prevented us from realizing this comparison.

In addition, we would have liked to compare different types of aggregations for the tweets. In our approach, we used only a "single tweet" classification meaning each tweet was considered as a document. As a consequence, the classifier could never grasp the notion of an "author" as a collection of 100 tweets and might have missed some interesting features. One could try to concatenate one author tweets into a single chunk of text or to consider the whole tweet collections as a document.

As we saw in section 3 (Experimental Results), we observed poor results in terms of gender classification. In order to improve those results, one could make use of the doc2vec tool which would improve significantly our results according to [1]. Another way to better the prediction of gender would be to train convolutional neural networks to extract automatically features from the tweets as described in [14].

5 Conclusion

In this paper we have offered a description of our approach in the context of PAN Author Profiling task. Our main aim was to compare loose classification to successive classification. The first one predicts each author's feature independently whereas the latest makes uses of each label prediction to sample the dataset and predict the remaining author's features. We selected the best classification units by comparing the combination of multiple features extractors and multiple classifiers while following a strict experimental protocol. The predictions rates regarding gender constrained us to submit a software implementing the loose classification.

References

1. Bartle, A., Zheng, J.: Gender classification with deep learning. Text-Interdisciplinary Journal (2003)
2. Biau, G.: Analysis of a random forests model. Journal of Machine Learning Research 13, 1063–1095 (2012)
3. Dichiu, D., Rancea, I.: Using machine learning algorithms for author profiling in social media. In: Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum. pp. 858–863 (2016)
4. Grivas, A., Krithara, A., Giannakopoulos, G.: Author Profiling using Stylometric and Structural Feature Groupings. In: CLEF (Working Notes) (2015)
5. Landauer, T.K., Foltz, P.W., Laham, D.: An introduction to latent semantic analysis. Discourse processes 25(2-3), 259–284 (1998)
6. Potthast, M., Rangel, F., Tschuggnall, M., Stamatatos, E., Rosso, P., Stein, B.: Overview of PAN'17: Author Identification, Author Profiling, and Author Obfuscation. In: Jones, G., Lawless, S., Gonzalo, J., Kelly, L., Goeuriot, L., Mandl, T., Cappellato, L., Ferro, N. (eds.) Experimental IR Meets Multilinguality, Multimodality, and Interaction. 8th International Conference of the CLEF Initiative (CLEF 17). Berlin Heidelberg New York (Sep.)

7. Rangel, F., Celli, F., Rosso, P., Potthast, M., Stein, B., Daelemans, W.: Overview of the 3rd author profiling task at pan 2015. In: Cappellato, L., Ferro, N., Jones, G.J.F., SanJuan, E. (eds.) CLEF (Working Notes). CEUR Workshop Proceedings, vol. 1391 (2015)
8. Rangel, F., Rosso, P., Potthast, M., Trenkmann, M., Stein, B., Verhoeven, B., Daeleman, W., others: Overview of the 2nd author profiling task at pan 2014. In: CEUR Workshop Proceedings. vol. 1180, pp. 898–927. CEUR Workshop Proceedings (2014)
9. Rangel, F., Rosso, P., Verhoeven, B., Daelemans, W., Potthast, M., Stein, B.: Overview of the 4th author profiling task at PAN 2016: cross-genre evaluations. In: Balog, K., Cappellato, L., Ferro, N., Macdonald, C. (eds.) CLEF (Working Notes). CEUR Workshop Proceedings, vol. 1609 (2016)
10. Rish, I.: An empirical study of the naive Bayes classifier. In: IJCAI 2001 workshop on empirical methods in artificial intelligence. vol. 3, pp. 41–46. IBM New York (2001)
11. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. Information Processing & Management 45(4), 427–437 (2009)
12. Steinwart, I., Christmann, A.: Support Vector Machines. Springer Publishing Company, Incorporated, 1st edn. (2008)
13. Vollenbroek, M.B.O., Carlotto, T., Kreutz, T., Medvedeva, M., Pool, C., Bjerva, J., Haagsma, H., Nissim, M.: Gronup: Groningen user profiling. In: Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, Évora, Portugal, 5-8 September, 2016. pp. 846–857 (2016)
14. Zhang, X., Zhao, J.J., LeCun, Y.: Character-level convolutional networks for text classification. In: Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015. pp. 649–657 (2015)