Word Unigram Weighing for Author Profiling at PAN 2018

Notebook for PAN at CLEF 2018

Pius von Däniken[1], Ralf Grubenmann[2], and Mark Cieliebak[1]

[1] Zurich University of Applied Sciences, Switzerland
{vode, ciel}@zhaw.ch
[2] SpinningBytes AG, Switzerland
rg@spinningbytes.com

Abstract We present our system for the author profiling task at PAN 2018 on gender identification on Twitter. The submitted system uses word unigrams, character 1- to 5-grams and emoji unigrams as features to train a logistic regression classifier. We explore the impact of three different word unigram weighing schemes on our system's performance. Our submission achieved accuracies of 77.42% for English, 74.64% for Spanish, and 73.20% for Arabic tweets. It ranked 15th out of 23 competitors.

1 Introduction

The rise of the internet and social media has brought a plethora of user generated content. Since an important amount of users post in pseudonymity, author profiling tasks, such as age and gender identification, have become a compelling area of study. For example, in the case of online harassment, one might be interested in identifying the perpetrator. This naturally extends to other applications in fields of forensics and security. Similarly, social sciences might be interested to use this as a jumping-off point to study how different demographics interact with media.

In this work, we describe our submission to the author profiling task at PAN 2018 [8,9] on gender identification based on text and images posted by users of social media. We compare different unigram weighing schemes for this task, which are the basis of our approach. Our submitted system achieved accuracies of 77.42% for English, 74.64% for Spanish, and 73.20% for Arabic Twitter messages.

1.1 Task Description

The goal of the author profiling task at PAN 2018 is to identify the gender of a user based on based on two input data: text written by the user, and images posted by the user on social media (not necessarily showing themselves). There are 3 different languages in the training data: *English* (3000 users), *Spanish* (3000 users), and *Arabic* (1500 users). The splits of male and female labeled authors are balanced for every language. For every user there are 100 messages and 10 images that the user posted to Twitter. The competition consists of three subtasks: *gender_txt*: identify gender from text only, *gender_img*: identify gender from images only, and *gender_comb*: identify gender from both text and images.

We participated in the *gender_txt* subtask for all three languages.

1.2 Related Work

This year's author profiling task is a continuation of a series of related tasks from previous years [5,6]. Most similar edition is the 2017 instance of the task [5], as it was using a very similar multilingual text data set based on Twitter and also had a gender identification subtask. The authors of [1] achieved the highest average accuracy over languages in the gender identification task, attaining accuracies of 82.33% for English, 83.21% for Spanish, and 80.06% for Arabic. They use word and character n-grams weighted by TF-IDF. Our work follows a similar approach. The VarDial evaluation campaign [4] is a similar competition, which focuses mainly on dialect identification, which has been a topic of previous tasks at PAN.

2 System Description

Our system uses the same approach for every language. First we preprocess the tweets to handle idiosyncrasies such as hashtags and user handles. Then we extract word unigram features, character n-gram features and emoji unigram features. Finally, we train a logistic regression classifier with those features.

2.1 Preprocessing

We use the same basic preprocessing pipeline for all languages.

First we substitute user mentions, email addresses, and URLs with special tokens. We use the regular expression `'@\S+'` to find and replace user mentions and `'\S+@\S+'` for email addresses. Inspired by the *URLExtract* [3] library we identify top-level domain names in the text and check the boundaries to find URLs. To handle Twitter's hashtags, we remove all *'#'* characters from the text and replace *'_'* (underscore) by a space character. Next we tokenize the text using the *WordPunctTokenizer* provided by the *Natural Language Toolkit (NLTK, version* 3.3*)* [3]. Finally we lowercase all tokens.

2.2 Feature Extraction

TF-ID$_u$F: The TF-ID$_u$F score was introduced in [2] as an alternative weighting scheme to the traditional *TF-IDF* weighting, based on a user's document collection. For a given term t it is computed as: $TF - ID_uF = tf(t)\log(\frac{N_u}{n_u(t)})$, where $tf(t)$ is the term frequency of t, N_u is the number of documents for user u, and $n_u(t)$ is the number of documents for user u that contain the term t. We decided to apply this method because, since we handle all of one author's texts at once, we can implement it in a stateless fashion.

[3] `https://github.com/lipoja/URLExtract`

Word Features: For every tweet of a user, we compute $TF - ID_uF$ features as described above. We compute the vocabulary of considered terms by retaining all terms that appear in the document collections of at least 2 users. In addition we set all non-zero term frequencies to 1 as we expect this to be less noisy than full term frequencies for short texts such as tweets.
Character Features: We extract character n-gram features for n ranging from 1 to 5. Every n-gram is considered at most once per tweet, and we use the hashing trick [10] to get a feature vector of dimension 2^{20}.

We use *HashingVectorizer*, the implementation provided by the *Scikit-learn (sklearn, version* 3.3*)* framework [7]. Since this implementation expects complete strings as input, we join the tokens from the preprocessing step with a whitespace character. This leads to n-grams spanning across word boundaries.
Emoji Features: Using the *emoji* [4] library, we extract emoji from tweets and weigh them using TF-ID_uF with the same settings as for word tokens.

2.3 Classification

We train a separate logistic regression classifier for each language, applying the *LogisticRegression* implementation provided by *sklearn* [7]. We use every text of every user as a separate sample for training, with the gender of the respective authors as labels. At inference time we get predictions for every text of an author from the classifier and predict the majority label.

3 Results

Table 1. Results of the ablation experiments. Best mean accuracies per language are indicated in bold

	English		Spanish		Arabic	
System	mean	std	mean	std	mean	std
tf	0.7633	0.0109	**0.7227**	0.0235	0.7287	0.0287
tf-idf	**0.7787**	0.0145	0.7070	0.0052	0.7460	0.0345
tf-iduf	0.7660	0.0197	0.7070	0.0120	0.7340	0.0255
tf-iduf & char	0.7750	0.0178	0.7210	0.0261	**0.7660**	0.0277
full	0.7650	0.0178	0.7057	0.0180	0.7460	0.0220

We performed ablation experiments to compare different ways of weighing the terms in a document. First we examine the performance of the different unigram weighing approaches on their own: **tf** is the system that uses raw term frequencies directly, **tf-idf** is the standard TF-IDF weighing of terms, and **tf-iduf** is the TF-ID_uF as described

[4] `https://github.com/carpedm20/emoji`

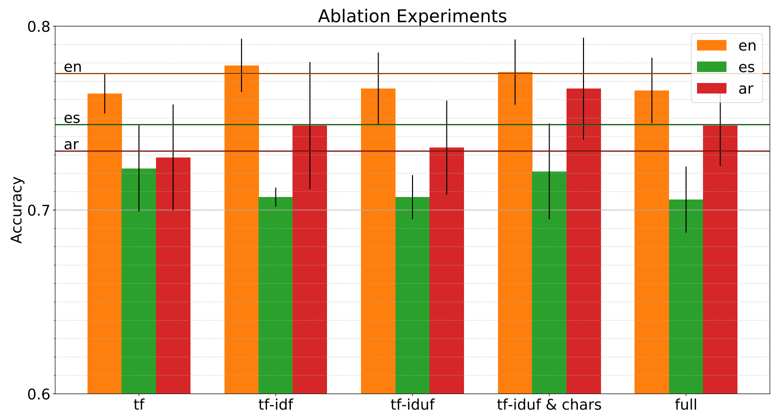

Figure 1. Visualization of the results of the ablation experiments

in Section 2.2. **tf-iduf & chars** uses word unigrams weighed by TF-ID$_u$F and character 1- to 5-grams. Finally, we refer as **full** to the system incorporating all features as described in Section 2. This is the system that we submitted to the competition.

To run the experiment we split the provided training data randomly into a training set and validation set. The split ratio of training to validation size is 80:20, i.e. 2400 authors for training and 600 for validation in the case of English and Spanish and 1200 authors for training and 300 for validation for Arabic.

Each experiment is run 5 times and we report mean and standard deviation. The numeric results are shown in Table 1 and Figure 1 gives a qualitative overview of the results. The bars show the mean accuracy on the validation split for each system and language. The error bars indicate the standard deviation. The horizontal lines show the results of our submission in the competition for reference.

There seem to be no qualitative differences between the explored feature sets and weighing schemes. The mean accuracies stay mostly within the error bars of each other per language. Furthermore for English and Arabic the validation accuracy is close to the accuracy attained by our submission. For Spanish the validation accuracy is apparently lower than the accuracy of our submission, but this might well be due to random chance.

4 Conclusion

We have given an overview of our classification system for gender identification. Our system attained accuracies of 77.42% for English, 74.64% for Spanish, and 73.20% for Arabic Twitter messages at the author profiling task at PAN 2018. We explored different word unigram weighing schemes and found that they all give similar performance when applied to our system.

References

1. Basile, A., Dwyer, G., Medvedeva, M., Rawee, J., Haagsma, H., Nissim, M.: N-GrAM: New Groningen Author-profiling Model. In: Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum, Dublin, Ireland, September 11-14, 2017. (2017)
2. Beel, J., Langer, S., Gipp, B.: TF-IDuF: A Novel Term-Weighting Sheme for User Modeling based on Users' Personal Document Collections. In: Proceedings of the iConference 2017. Wuhan, China (Mar 22 - 25 2017), http://ischools.org/the-iconference/
3. Bird, S., Loper, E.: Nltk: the natural language toolkit. In: Proceedings of the ACL 2004 on Interactive poster and demonstration sessions. p. 31. Association for Computational Linguistics (2004)
4. Nakov, P., Zampieri, M., Ljubešić, N., Tiedemann, J., Malmasi, S., Ali, A.: Proceedings of the fourth workshop on nlp for similar languages, varieties and dialects (vardial). In: Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial). Association for Computational Linguistics (2017), http://aclweb.org/anthology/W17-1200
5. Pardo, F.M.R., Rosso, P., Potthast, M., Stein, B.: Overview of the 5th author profiling task at PAN 2017: Gender and language variety identification in twitter. In: Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum, Dublin, Ireland, September 11-14, 2017. (2017)
6. Pardo, F.M.R., Rosso, P., Verhoeven, B., Daelemans, W., Potthast, M., Stein, B.: Overview of the 4th author profiling task at PAN 2016: Cross-genre evaluations. In: Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, Évora, Portugal, 5-8 September, 2016. pp. 750–784 (2016)
7. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, 2825–2830 (2011)
8. Rangel, F., Rosso, P., Montes-y-Gómez, M., Potthast, M., Stein, B.: Overview of the 6th Author Profiling Task at PAN 2018: Multimodal Gender Identification in Twitter. In: Cappellato, L., Ferro, N., Nie, J.Y., Soulier, L. (eds.) Working Notes Papers of the CLEF 2018 Evaluation Labs. CEUR Workshop Proceedings, CLEF and CEUR-WS.org (Sep 2018)
9. Stamatatos, E., Rangel, F., Tschuggnall, M., Kestemont, M., Rosso, P., Stein, B., Potthast, M.: Overview of PAN-2018: Author Identification, Author Profiling, and Author Obfuscation. In: Bellot, P., Trabelsi, C., Mothe, J., Murtagh, F., Nie, J., Soulier, L., Sanjuan, E., Cappellato, L., Ferro, N. (eds.) Experimental IR Meets Multilinguality, Multimodality, and Interaction. 9th International Conference of the CLEF Initiative (CLEF 18). Springer, Berlin Heidelberg New York (Sep 2018)
10. Weinberger, K., Dasgupta, A., Langford, J., Smola, A., Attenberg, J.: Feature hashing for large scale multitask learning. In: Proceedings of the 26th Annual International Conference on Machine Learning. pp. 1113–1120. ICML '09, ACM, New York, NY, USA (2009), http://doi.acm.org/10.1145/1553374.1553516