Adapting Cross-Genre Author Profiling to Language and Corpus

Notebook for PAN at CLEF 2016

Ilia Markov, Helena Gómez-Adorno, Grigori Sidorov, and Alexander Gelbukh

Instituto Politécnico Nacional,
Center for Computing Research,
Mexico City, Mexico
markovilya@yahoo.com, helena.adorno@gmail.com, sidorov@cic.ipn.mx,
gelbukh@gelbukh.com

Abstract This paper presents our approach to the Author Profiling (AP) task at PAN 2016. The task aims at identifying the author's age and gender under cross-genre AP conditions in three languages: English, Spanish, and Dutch. Our preprocessing stage includes reducing non-textual features to their corresponding semantic classes. We exploit typed character n-grams, lexical features, and non-textual features (domain names). We experimented with various feature representations (binary, raw frequency, normalized frequency, second order attributes (SOA), tf-idf) and machine learning algorithms (liblinear and libSVM implementations of Support Vector Machines (SVM), multinomial naive Bayes, logistic regression). For textual feature selection, we applied the transition point technique, except when SOA was used. We found that the optimal configuration was different for different languages at each stage.

1 Introduction

Author Profiling (AP) is the task that aims at identifying profiling aspects of an author based solely on a sample of his or her writing. From the machine-learning perspective, AP can be viewed as a multiclass, single-label classification problem, when automatic methods have to assign class labels (e.g., male, female) to objects (texts). The AP methods can be useful for security and marketing applications, as well as contribute to forensics purposes, when part of the evidence refers to texts.

The rapid growth of social media in past years has significantly contributed to the increased interest in the task, giving rise to a large number of substantial work in this field. Most of these approaches concerned with exploring different sets of features to distinguish between specific profiles. According to the AP task literature, character n-grams and lexical features have proved to be highly discriminative for this task, regardless of the language the texts are written in [7, 8, 14, 17].

Recently, different types of character n-grams were proposed by Sapkota *et al.* [16] to tackle the task of Authorship Attribution (AA). The authors showed that some types of character n-grams distinguish better between stylistic properties of an author than other types, both under single and cross-topic AA conditions. In this study, we apply

the approach proposed by Sapkota *et al.* [16] to the task of AP. We demonstrate that using typed character n-grams along with lexical and non-textual features is also helpful for distinguishing between profiling aspects of authors under cross-genre AP conditions, that is, the training corpus is on one genre, while the test set is on another genre. We propose several pre-processing steps and apply transition point technique based on Zipf's law [19] to fine-tune the feature set. We examine various feature representations, including second order attributes (SOA) [10], which is known to provide good results for this task [1, 11].

The rest of this paper is organized as follows. Section 2 presents the proposed methodology. Section 3 provides some characteristics of the PAN Author Profiling 2016 corpus. Section 4 describes the conducted experiments. Section 5 provides the obtained results and their evaluation. Section 6 draws the conclusions and points to the possible directions of future work.

2 Methodology

2.1 Pre-processing steps

Since the provided training corpus (described further in Section 3) consists of Twitter messages, and the evaluation corpus will be on another genre, we introduce the following pre-processing steps, which are applied before the extraction of features, aiming to reduce non-textual features to their semantic classes:

Digits We replace all digits with a single symbol (e.g., 345 $\rightarrow$ 0), which allows capturing information about their occurrence, discarding the actual numbers, since the numbers do not represent useful information concerning profiling aspects.

URLs In order to keep information about the presence of URL mentions and not to extract character n-grams from them, we replace all URL mentions with the same symbol. However, we use the information regarding the particular domain name in order to form our feature set of domain names (e.g., https://www.instagram.com $\rightarrow$ 1, "instagram" $\rightarrow$ feature set of domain names).

@mentions We replace all @mention instances with the same symbol in order to keep track of their occurrence and remove information related to the specific username mentioned (e.g., @mention $\rightarrow$ 2). If there is a space after the "@" symbol, in most cases, it is followed by a specific location. Location mention is usually user specific and does not carry useful clues for distinguishing between communities of people who share common profiling aspects. Therefore, we replace @_mention with a different symbol (e.g., @_mention $\rightarrow$ 3).

Picture links For the same purposes as the previous steps, all picture links are replaced with a single symbol (e.g., pic.twitter.com/vYpLShlHs7 $\rightarrow$ 4).

Emoticons Emoticons can provide useful information about sentiments of a specific

user; however, we consider them not to be helpful for author profile identification, especially under cross-gender conditions. Therefore, we are only interested in capturing their presence (e.g., :) $\rightarrow$ 5).

Furthermore, we apply the following normalization:

Slang words We expand slang words with their corresponding meanings, since slang words are not used in the same way by all authors, especially taking into account that the test set will be on another genre (e.g., 4u $\rightarrow$ for you).

Punctuation marks We split punctuation marks from adjacent words and from each other to be able to capture their presence separately when using character n-grams features (e.g., ." $\rightarrow$. ").

2.2 Features

Our approach is based on the character n-grams categories introduced by Sapkota *et al.* [16]. The authors defined 10 different character n-gram categories based on affixes, words, and punctuation. Following the practice of Sapkota *et al.* [16], we examine three cases according to what kind of n-gram categories are used:

1. **Untyped** - traditional approach to extracting n-grams, where the categories of n-grams are ignored. Any distinct n-gram is a different feature.
2. **Typed** - when n-grams of all the categories (affix+word+punctuation) are considered. Instances of the same n-gram may refer to different features.
3. **Affix+punctuation** - when the n-grams of the word category are excluded.

The main conclusion of Sapkota *et al.* [16] is that for the Authorship Attribution task, models based on affix+punctuation features are more efficient than models trained on all the features. In this study, we apply these three models to the task of AP and examine which one of them is more appropriate for the AP task.

In addition, we examine whether the effectiveness of the proposed models can be enhanced when combined with lexical and non-textual features, since combining different feature sets usually improves the performance of classification models [3].

2.3 Transition point technique for feature selection

Zipf's law states that given a large enough corpus, the frequency ranks of words (terms) are inversely proportional to the corresponding frequencies [19]. Transition point (TP) technique is based on Zipf's law and word occurrences. This technique splits the vocabulary of a document into two sets of terms (low and high frequency). According to Pinto *et al.* [12], the terms whose frequency is closer to the transition point value (medium-frequency terms) have a higher semantic value, and therefore, are more appropriate for document representation. These medium-frequency terms can be obtained by setting lower (U_1) and upper (U_2) threshold values through selecting appropriate neighbourhood values of transition point (NTP).

The formula to obtain the transition point is given in equation (1):

$$TP = \frac{\sqrt{1 + 8 \times I_1} - 1}{2}, \tag{1}$$

where I_1 represents the number of words with frequency equal to 1 [12].

The lower (U_1) and upper (U_2) threshold values are calculated by a given neighbourhood value of TP ($NTP \in$ [0-1]):

$$U_1 = (1 - NTP) \times TP, \tag{2}$$

$$U_2 = (1 + NTP) \times TP. \tag{3}$$

Transition point technique has been used in various areas of Natural Language Processing (NLP) and has proved to perform better than traditional feature selection methods for several classification tasks [12]. In this work, we apply the transition point technique to our character n-grams and lexical sets of features. We further demonstrate that this feature selection method can enhance the performance of cross-gender AP system.

3 Corpus

The Author Profiling task at PAN 2016 consisted in predicting age and gender of authors under cross-gender AP conditions [15]. The provided training corpus is composed of Twitter messages in English, Spanish, and Dutch. The English and Spanish training datasets are labeled with age and gender, whereas the Dutch dataset is labeled only with gender. The following age classes are considered: 18-24, 25-34, 35-49, 50-64, and 65-xx. The distribution of age and gender over the instances of the training set can be seen in Table 1.

Table 1. Age and gender distribution over the PAN Author Profiling 2016 training corpus.

		English	**Spanish**	**Dutch**
Age	18-24	26	16	–
	25-34	136	64	–
	35-49	182	126	–
	50-64	78	38	–
	65-xx	6	6	–
Gender	Male	214	125	192
	Female	214	125	192
Total:		**428**	**250**	**384**

The PAN Author Profiling 2016 training corpus is perfectly balanced in terms of represented gender groups; however, it is highly unbalanced in terms of age classes. The majority of participants falls into the 35-49 age category, when there are only few instances for the 65-xx age category, which makes the task more challenging.

4 Adapting Procedures to Language and Corpus

For the evaluation of the proposed approach, we conducted our experiments on both, the provided training dataset under 10-fold cross-validation and the PAN Author Profiling 2014 training corpus composed of English and Spanish blogs, social media, and reviews. We used the PAN Author Profiling 2014 training corpus as a test set for our experiments. Following the proposed performance measure, we evaluated our system by measuring classification accuracy on both corpora (PAN 2016 and PAN 2014).

In order to perform the pre-processing steps as described in Section 2, we expanded slang words and replaced emoticons using the dictionary developed by Gómez-Adorno *et al.* [6].

The examined features, machine learning algorithms, feature representations, and threshold values are shown in Table 2.

Table 2. Examined system configurations. U_1 and U_2 correspond to the lower and the upper threshold values; TP - transition point.

Features	**ML algorithm**	**Feature representation**	**Threshold**
Typed char. n-grams	Liblinear	Binary	Freq. $\geq U_1$
Untyped char. n-grams	LibSVM	Raw freq.	Freq. $\geq U_2$
Affix+punct char. n-grams	Multinomial naive Bayes	Normalized freq.	$U_1 \leq$ Freq. $\leq U_2$
Word unigrams	Logistic regression	SOA	Freq. $\geq TP$
Word bigrams	Ensemble	Tf-idf	Freq. $\leq TP$
Stems			
Domain names			

We also experimented with Latent Semantic Analysis (LSA) of words and stems, which did not yield good results. Furthermore, we measured the impact of tackling the task as a single-labeled 10 class classification problem using 10 age-gender profiling classes.

We evaluated the performance of each of the feature sets separately and in combinations. Regarding the character n-grams features, we conducted experiments with different values of n ranging from 3 to 6 for untyped and from 3 to 4 for typed and affix+punctuation character n-grams. In addition, we examined the contribution of each category of character n-grams separately, as well as the performance of our system when n-grams of different length are combined.

Typed character trigrams generally provided a higher level of classification accuracy than untyped and affix-punctuation character n-grams. They also have proved to be more predicative than typed character n-grams with a higher values of n, and therefore, were included in the final system. Furthermore, their combination with word unigrams (for English, Spanish, and Dutch) and domain names (for English and Dutch) features allowed us to further enhance system performance. However, it is necessary to mention that the models based on untyped and affix+punctuation character n-grams produced nearly as high levels of classification accuracy as the model based on typed character n-grams. Moreover, different values of n yielded only slight accuracy variations.

We examined the performance of the machine learning classifiers, shown in Table 2, using their scikit-learn [2] implementation. These classification algorithms are considered among the best for text classification tasks [9, 14]. We evaluated the performance of each of the classifiers separately, as well as examined an ensemble setup, which combines the probability distributions provided by the individual classifiers based on majority voting scheme.

Feature representations used in this work are shown in Table 2. We exploited second order attributes (SOA) computed as in [11] with age-gender pairs as profiles calculated separately for n-grams and word unigrams. Applying SOA, we reduced the number of features to 10 for each of the feature sets (n-grams and word unigrams).

Gelbukh and Sidorov [4] showed that Zipf's law coefficients depend on language. Therefore, when applying the transition point technique to our character n-grams set of features, we evaluated threshold values for each of the languages separately based on grid search. We selected all the n-grams with a frequency greater than or equal to the upper threshold (U_2), with the NTP values of 0.90, 1.00, and –0.95 for the English, Spanish, and Dutch datasets, respectively. We also used a fixed frequency cutoff, which consisted in discarding 10 most frequent n-grams for each language.

In order to compose our lexical set of features, first, we discarded 100 most frequent words from the English and Spanish datasets and 50 most frequent words from the Dutch one. In the same way as for character n-grams, we estimated the most appropriate threshold values and selected all the words with a frequency greater than or equal to the lower threshold (U_1). The lower threshold NTP values for our lexical set of features were 0.75, 0.90, and 0.90 for the English, Spanish, and Dutch datasets, respectively.

Our non-textual set of features was composed of 30 most frequent domain names for each of the languages.

We submitted three systems for the final evaluation on the PAN Author Profiling 2016 test corpus. The best results were obtained with the configurations shown in Table 3.

Table 3. Best systems configurations. U_1 and U_2 correspond to the lower and the upper threshold values; NTP - neighbourhood value of transition point; N - number of features.

Language	**Features**	**ML algorithm**	**Feature representation**	**Typed char. trigrams threshold**	**Word unigrams threshold**	**N**
English	Typed char. trigrams, word unigrams, domain names	LibSVM	Binary	Freq. $\geq U_2$, NTP = 0.90	Freq. $\geq U_1$, NTP = 0.75	7,715
Spanish	Typed char. trigrams, word unigrams	Liblinear	SOA	All (34,583)	All (33,276)	20
Dutch	Typed char. trigrams, word unigrams, domain names	Liblinear	Binary	Freq. $\geq U_2$, NTP = –0.95	Freq. $\geq U_2$, NTP = 0.90	18,441

The best performing system, by a small margin, was a system consisting of training libSVM (for English) and liblinear (for Dutch) classifiers on the combination of typed character trigrams, word unigrams, and domain names features using their binary representation. The Spanish system consisted of training liblinear classifier on typed character trigrams and word unigrams features using SOA representation. Our final setup for libSVM classifier employed a linear kernel. Both libSVM and liblinear classifiers used the "balanced" class weight mode.

5 Experimental Results

In Table 4, we present the results on the PAN Author Profiling 2016 test corpus for the three submitted systems evaluated in TIRA [5]. Systems 1 and 2 are based on binary feature representation and liblinear and libSVM classifiers, respectively. System 3 is composed of SOA and liblinear classifier. The three systems were evaluated for each of the languages in order to examine their performance on the test set. The best results for each language are in bold.

Table 4. Evaluation results in terms of classification accuracy on the PAN Author Profiling 2016 test corpus.

	System 1			**System 2**			**System 3**		
	Binary, liblinear			Binary, libSVM			SOA, liblinear		
Language	**Age**	**Gender**	**Joint**	**Age**	**Gender**	**Joint**	**Age**	**Gender**	**Joint**
English	0.4103	0.6026	0.2436	**0.4487**	**0.6154**	**0.2949**	0.4487	0.5641	0.2949
Spanish	0.5000	0.6250	0.3571	0.4821	0.5000	0.2679	**0.4464**	**0.6607**	**0.3750**
Dutch	–	**0.5100**	–	–	0.5000	–	–	0.4880	–

Table 5 shows the best results on the PAN Author Profiling 2016 test corpus for the three different languages.

Table 5. Best results in terms of classification accuracy on the PAN Author Profiling 2016 test corpus.

Language	**Age**	**Gender**	**Joint**
English	0.4487	0.6154	0.2949
Spanish	0.4464	0.6607	0.3750
Dutch	–	0.5100	–

In case of age classification, the obtained results for English and Spanish were almost equal, in spite of different approaches used to tackle these two languages. The accuracy of gender classification for Spanish was good, even though it had fewer instances for training. The obtained results for the Dutch language were rather low; this

can be due to the fact that we did not tune the system under cross-genre conditions for this language, as we did for English and Spanish.

The main lesson learned was that each language required different configuration at each stage.

6 Conclusions

In this paper, we presented an approach for cross-genre age and gender identification. Our final system for the English and Dutch languages combined typed character n-grams, lexical features, and non-textual features. LibSVM and liblinear classifiers were used for the English and Dutch languages, respectively. We employed binary feature encoding and the transition point technique to fine-tune the size of the feature set depending on language. For the Spanish language, the system was composed of typed character n-grams and lexical features to build a liblinear classifier. We employed the second order attributes (SOA) technique, which yielded a higher classification accuracy for this language than others examined feature representations. For all the three languages, we applied the same pre-processing steps, which includes reducing non-textual features to their corresponding semantic classes.

One of the directions for future work would be to conduct experiments combining the proposed features with others of a distinct nature such as syntactic [13, 18] and corpus statistics features: lexical diversity, lexical sophistication, and lexical density, among others. Moreover, we intent to develop a method for automatic definition of optimal neighbourhood values of the transition point technique depending on both language and corpus.

Acknowledgements

This work was done under partial support of the Mexican Government (CONACYT project 240844, SNI, COFAA-IPN, SIP-IPN 20161947).

References

1. Álvarez-Carmona, M.A., López-Monroy, A.P., Montes-y-Gómez, M., Villaseñor-Pineda, L., Jair-Escalante, H.: INAOE's participation at PAN'15: Author profiling task. In: Working Notes Papers of the CLEF 2015 Evaluation Labs. CLEF '15, vol. 1391. CEUR (2015)
2. Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., Varoquaux, G.: API design for machine learning software: experiences from the scikit-learn project. In: ECML PKDD Workshop: Languages for Data Mining and Machine Learning. pp. 108–122 (2013)
3. Estival, D., Gaustad, T., Hutchinson, B., Pham, S.B., Radford, W.: Author profiling for English emails. In: Proceedings of the 10^{th} Conference of the Pacific Association for Computational Linguistics. pp. 263–272. PACLING '07 (2007)
4. Gelbukh, A., Sidorov, G.: Zipf and Heaps laws' coefficients depend on language. In: Proceedings of the 2^{nd} International Conference on Intelligent Text Processing and Computational Linguistics. pp. 332–335. CICLing '01, Springer Berlin Heidelberg (2001)

5. Gollub, T., Stein, B., Burrows, S., Hoppe, D.: TIRA: Configuring, executing, and disseminating information retrieval experiments. In: Proceedings of the 9^{th} International Workshop on Text-based Information Retrieval at DEXA. pp. 151–155. TIR '12, IEEE (2012)
6. Gómez-Adorno, H., Markov, I., Sidorov, G., Posadas-Durán, J., Fócil-Arias, C.: Compilación de un lexicón de redes sociales para la identificación de perfiles de autor [Compiling a lexicon of social media for the author profiling task] (in Spanish, abstract in English). Research in Computing Science (accepted) 115 (2016)
7. González-Gallardo, C.E., Montes, A., Sierra, G., Núñez-Juárez, J.A., Salinas-López, A.J., Ek, J.: Tweets classification using corpus dependent tags, character and POS n-grams. In: Working Notes Papers of the CLEF 2015 Evaluation Labs. CLEF '15, vol. 1391. CEUR (2015)
8. Houvardas, J., Stamatatos, E.: N-gram feature selection for authorship identification. In: Proceedings of the 12^{th} International Conference on Artificial Intelligence: Methodologies, Systems, and Applications. pp. 77–86. AIMSA '06, Springer Berlin Heidelberg (2006)
9. Kibriya, A.M., Frank, E., Pfahringer, B., Holmes, G.: Multinomial naive Bayes for text categorization revisited. In: Proceedings of the 17^{th} Australian joint conference on Advances in Artificial Intelligence. pp. 488–499. AI '04 (2005)
10. López-Monroy, A.P., Montes-y-Gómez, M., Escalante, H.J., Villaseñor-Pineda, L., Stamatatos, E.: Discriminative subprofile-specific representations for author profiling in social media. Knowledge-Based Systems 89(C), 134–147 (2015)
11. López-Monroy, A.P., Montes-y-Gómez, M., Escalante, H.J., Villaseñor-Pineda, L., Villatoro-Tello, E.: INAOE's participation at PAN'13: Author profiling task. In: Working Notes Papers of the CLEF 2013 Evaluation Labs. CLEF '13, CEUR (2013)
12. Pinto, D., Jiménez-Salazar, H., Rosso, P.: Clustering abstracts of scientific texts using the transition point technique. In: Proceedings of the 7^{th} International Conference on Intelligent Text Processing and Computational Linguistics. pp. 536–546. CICLing '06, Springer Berlin Heidelberg (2006)
13. Posadas-Durán, J., Markov, I., Gómez-Adorno, H., Sidorov, G., Batyrshin, I., Gelbukh, A., Pichardo-Lagunas, O.: Syntactic n-grams as features for the author profiling task. In: Working Notes Papers of the CLEF 2015 Evaluation Labs. CLEF '15, vol. 1391. CEUR (2015)
14. Rangel, F., Celli, F., Rosso, P., Pottast, M., Stein, B., Daelemans, W.: Overview of the 3^{rd} author profiling task at PAN 2015. In: Cappelato, L., Ferro, N., Jones, G., Juan, E.S. (eds.) CLEF 2015 Labs and Workshops, Notebook Papers. vol. 1391. CEUR (2015)
15. Rangel, F., Rosso, P., Verhoeven, B., Daelemans, W., Potthast, M., Stein, B.: Overview of the 4^{th} author profiling task at PAN 2016: Cross-genre evaluations. In: Working Notes Papers of the CLEF 2016 Evaluation Labs. CEUR Workshop Proceedings, CLEF and CEUR-WS.org (2016)
16. Sapkota, U., Bethard, S., Montes-y-Gómez, M., Solorio, T.: Not all character n-grams are created equal: A study in authorship attribution. In: Proceedings of the 2015 Annual Conference of the North American Chapter of the ACL: Human Language Technologies. pp. 93–102. NAACL-HLT '15, Association for Computational Linguistics (2015)
17. Schler, J., Koppel, M., Argamon, S., Pennebaker, J.: Effects of age and gender on blogging. In: AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs. pp. 199–205. AAAI (2006)
18. Sidorov, G., Gómez-Adorno, H., Markov, I., Pinto, D., Loya, N.: Computing text similarity using tree edit distance. In: Proceedings of the Annual Conference of the North American Fuzzy Information processing Society and 5^{th} World Conference on Soft Computing. pp. 1–4. NAFIPS '15 (2015)
19. Zipf, G.K.: Human behavior and the principle of least effort. Cambridge, MA, Addison-Wesley (1949)