Using Intra-Profile Information for Author Profiling
Notebook for PAN at CLEF 2014

A. Pastor López-Monroy, Manuel Montes-y-Gómez,
Hugo Jair Escalante, and Luis Villaseñor-Pineda

Laboratory of Language Technologies, Department of Computer Science,
Instituto Nacional de Astrofísica, Óptica y Electrónica,
Luis Enrique Erro No. 1, C.P. 72840, Pue. Puebla, México
{pastor, mmontesg, hugojair, villasen}@ccc.inaoep.mx

Abstract In this paper we describe the participation of the Laboratory of Language Technologies of INAOE at PAN 2014. We address the Author Profiling (AP) task finding and exploiting relationships among terms, documents, profiles and *subprofiles*. Our approach uses the idea of second order attributes (a low-dimensional and dense document representation) [4], but goes beyond incorporating information among each target profile. The proposed representation deepen the analysis incorporating information among texts in the same profile, this is, we focus in subprofiles. For this, we automatically find subprofiles and build document vectors that represent more detailed relationships of documents and subprofiles. We compare the proposed representation with the standard Bag-of-Terms and the best method in PAN13 using the PAN 2014 corpora for AP task. Results show evidence of the usefulness of intra-profile information to determine gender and age profiles. According to the PAN 2014 official results, the proposed method was one of the best three approaches for most social media domains. Particularly, it achieved the best performance in predicting age and gender profiles for blogs and tweets in English.

Keywords: Age Identification, Gender Identification, Subprofiles Generation, Subclass Information

1 Introduction

For several years the scientific community has been discussing the following basic question: How much information can be known from an author's document?, Commonly known as the Author Profiling (AP) [3,6,1], this task is of great interest because of its wide applicability to problems in different areas, such as: business intelligence, criminal law, computer forensics, etc.

In this paper, we use well-known textual features for AP [3,6], but we focus in the representation of documents, exposing its key role in the problem. For this, we mainly consider the Second Order Attributes (SOA) proposed in [4], SOA is an approach that builds document vectors in a space of profiles. Under this representation, each value in the vector represents the relationship of each document with each target profile. Notwithstanding the usefulness of the approach in [4,5], the method has an evident shortcoming: this approach basically assumes that the relationship between a vocabulary term (i.e., a word) and a group of authors (a target profile) can be represented using one single value. For example, in López-Monroy et. al. (2013), the representation of the word *linux* is highly related to the *male* profile, therefore its occurrence in a given document, causes a dramatically increase in the probability of belonging to the *male* profile. This behaviour could make difficult to identify the correct profile for some authors (e.g., classifying documents belonging to *females* writing about technology). We believe that such assumption is in some extend naive and can be alleviated through the automatic generation of subprofiles.

In this work, we generate new highly informative attributes that represents relationships among terms, documents, profiles and also *subprofiles*. In order to automatically generate the aforementioned subprofiles we propose dividing each target profile into several groups using a clustering algorithm. Then we build the final document representation on the top of the generated subprofiles, using them as the new target profiles. This approach improves even more the representation of documents in the AP task, and also mitigates common problems of other standard representations (e.g. the Bag-of-Terms, BOT), for example: i) high dimensionality, and ii) the sparseness of the representation. Results using the latter ideas also seem promising and competitive compared with other approaches and systems in PAN 2014 forum.

The rest of this paper is organized as follows: Section 2 introduces the proposed approach, Section 3 explains the evaluation and the obtained results, finally Section 4 outlines the conclusions.

2 Computing Intra-Profile Relationships

The proposed method has three main stages to represent documents: i) representing terms in a space of profiles, ii) representing documents in a space of profiles, and iii) generating subprofiles and re-compute steps 1 and 2 using subprofiles as the new target profiles. The rest of this section explains the above steps in detail.

2.1 Representing terms in a space of profiles

The intuitive idea is to capture the relationship of each term (i.e., a word, an n-gram, a punctuation mark, etc.) with each one of the target profiles. Let $\{t_1, \ldots, t_m\}$ be the vocabulary in the corpus, and $\{p_1, \ldots, p_n\}$ be the set of target profiles. We build term vectors $\mathbf{t_i} = \langle tp_{i1}, \ldots, tp_{in} \rangle$, where tp_{ij} represents the relationship of the term t_i with the profile p_j. Equation 1 reflects the idea for computing tp_{ij}.

$$w_{ij} = \sum_{k:d_k \in P_j} \log_2 \left(1 + \frac{\mathit{tf}_{ki}}{\mathit{len}(d_k)} \right) \tag{1}$$

where P_j are training documents with profile p_j, tf_{ik} is the term frequency of the term t_i in the document d_k, and $len(d_k)$ is the number of terms in the document d_k. To avoid high cumulative term frequencies in high unbalanced data, a normalization that considers the proportion of each term in each profile is performed (Equations 2.1 and 2.2).

$$(2.1)\ \ tp_{ij} = \frac{w_{ij}}{\sum\limits_{i=1}^{TERMS} w_{ij}} \qquad (2.2)\ \ tp_{ij} = \frac{w_{ij}}{\sum\limits_{j=1}^{PROFILES} w_{ij}} \tag{2}$$

2.2 Representing Documents in a space of profiles

The intuitive idea is to capture the relationship of each document (i.e., a blog, a tweet, a review etc.) with each one of the target profiles. For this we use the previously computed term vectors to build document vectors in a space of profiles. Thus, to build the representation of each document we add its term vectors weighted by their frequency tf_{kj}. Thus, we build document vectors $\mathbf{d_k} = \langle dp_{1k}, \ldots, dp_{nk} \rangle$, where n is number profiles, and dp_{ik} reflects the relationship of the document d_k and the profile p_i. Equation (3) expose the latter ideas.

$$\boldsymbol{d}_k = \sum_{t_i \epsilon D_k} \frac{tf_{ik}}{len(d_k)} \times \boldsymbol{t}_i \tag{3}$$

where D_k is the set of terms that belongs to document d_k

2.3 Generating subprofiles

The latter ideas generate vectors where each value represents the relationship between a document and each target profile. The intuitive idea is that the representation assumes certain homogeneity in documents belonging to the same target profile, then a single relationship value per profile is computed [4]. In spite of the usefulness of this assumption [4,5], it is in some extend naive, because even when a group of authors could share the same general profile (e.g., females), there could be more specific subgroups of females with finer differences (say, young-gamer females and housewife females). Thus, each target profile is in some extend heterogeneous among its authors.

Generating subprofiles involves discovering natural subgroups among authors belonging to the same profile. In this regard, we decide to use the latter generated document representation to build document vectors in a space of profiles, then cluster documents in the same profile. The intuitive idea of this approach, is to use an appropriated base representation for AP to find documents similar to each other inside that space. Once a set of clusters (subprofiles) for each target profile are generated, we rebuilt the SOA using all found clusters as the new target profiles. In this way, as indicated in formula (3), we end up with a set of attributes that represents relationships between documents and detailed subprofiles. In order to build the aforementioned subprofiles, we have used the Expectation Maximization Clustering (EMC) algorithm.

3 Experimental Results

We approached the AP problem in a separated way: i) age, and ii) gender prediction. Thus, we have five age profile classes; *18-24, 25-34, 35-49, 50-64,* and *65-more*. Also we have two gender profiling classes; *male*, *female*. Given this context, we build subprofile attributes for age, and different subprofile attributes for gender, then we train two classifiers, one for each representation. In order to evaluate and compare this proposal, we have used the following experimental settings for the training dataset: i) the most 3,000 frequent terms as features, and ii) the standard LibLINEAR classifier without any parameter optimization [2]. As terms we use words, contractions, words with hyphens, punctuation marks and a set of common slang vocabulary. From Table 1 it can be seen how the proposed approach (n SOA per profile) outperforms BOT and the best PAN 13 approach [4] (1-SOA per profile), using the PAN 14 corpus over different social media domains. We believe this is because finding subprofiles in the target profiles (10-fold cross validation over the training dataset was performed), provides a more detailed perspective for documents. In this regard, n-SOA is a novel representation that capture more important details about profiles and subprofiles, in contrast to 1-SOA proposed in [4], which captures more general information of profiles.

Age and Gender prediction in English copora									
		Blogs		Twitter		Social Media		Reviews	
Dataset	Representation	Age	Gender	Age	Gender	Age	Gender	Age	Gender
Train	BoT	45.57	73.87	39.21	71.52	34.30	54.29	31.17	64.87
	1-SOA	46.72	75.44	43.52	70.52	35.81	55.01	32.63	66.75
	n-SOA	**48.07**	**77.96**	**47.97**	**71.98**	**37.00**	**55.36**	**33.92**	**68.05**
Test	n-SOA	39.74	67.95	49.35	72.08	35.52	52.37	33.37	68.09

Table 1. Accuracy prediction for BOT, 1-SOA [4], and the proposed n-SOA using the train (under a 10 Cross Fold Validation) and test datasets, for age and gender profiles in PAN14 English corpora.

Age and Gender prediction in Spanish corpora							
		Blogs		Twitter		Social Media	
Dataset	Representation	Age	Gender	Age	Gender	Age	Gender
Train	BoT	43.18	62.50	39.88	62.60	37.65	63.83
	1-SOA	45.33	62.91	41.54	62.01	38.88	64.47
	n-SOA	**48.22**	**63.05**	**43.61**	**62.51**	**41.42**	**65.35**
Test	n-SOA	48.21	58.93	53.33	60.00	45.23	64.84

Table 2. Accuracy prediction for BOT, 1-SOA [4], and the proposed n-SOA using the train (under a 10 Cross Fold Validation) and test datasets, for age and gender profiles in PAN14 Spanish corpora.

According to the PAN14 evaluation, the proposed attributes, get the best test-set accuracy performance for age and gender prediction in blogs and tweets domains for English language. Moreover, the reported results are in the top 3 positions for other social media domains. Thus, the approach presented in this paper is an effective alternative to address the AP task in different social media domains, where documents presents challenging difficulties hindered the accurate work of most natural language processing tools.

4 Conclusions

In this paper we presented a novel approach that considers the existing information among documents belonging to the same class. This is, even for authors belonging to the same target profiles (e.g., males), the approach look for more specific subgroups of authors (e.g., male employees and male gamers) in order to consider intra-profile information. To the best of our knowledge, this is the first time that AP is addressed using this kind of intra-class-relationships inside the target profiles. Such relationships help to achieve a better discrimination among several profiles. Using these automatically generated attributes, the classifier can keep good classification rates, even for imbalanced data. This is due to the relations among terms, documents and subprofiles, which provides few but more detailed predictive attributes. We have shown better experimental results than the standard BOT, the best method of PAN13, and most of the approaches participating at PAN14.

Acknowledgements: This work was partially funded by project CONACyT-Mexico 134186 and the program ECOS under the project M11-H04. López-Monroy also thanks for doctoral scholarship CONACyT-México 243957.

References

1. Argamon, S., Koppel, M., Pennebaker, J.W., Schler, J.: Automatically profiling the author of an anonymous text. Communications of the ACM 52(2), 119–123 (2009)
2. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research 9, 1871–1874 (2008)
3. Koppel, M., Argamon, S., Shimoni, A.R.: Automatically categorizing written texts by author gender. Literary and Linguistic Computing 17(4), 401–412 (2002)
4. Lopez-Monroy, A.P., Montes-Y-Gomez, M., Escalante, H.J., Villasenor-Pineda, L., Villatoro-Tello, E.: Inaoe's participation at pan'13: Author profiling task. In: Notebook Papers of CLEF 2013 LABs and Workshops, Valencia, Spain, September (2013)
5. Rangel, F., Rosso, P., Koppel, M., Stamatatos, E., Inches, G.: Overview of the author profiling task at pan 2013. In: Forner, P., Navigli, R., Tufis, D. (eds.) Working Notes Papers of the CLEF 2013 Evaluation Labs, September 2013 (2013)
6. Schler, J., Koppel, M., Argamon, S., Pennebaker, J.: Effects of age and gender on blogging. In: Proceedings of 2006 AAAI Spring Symposium on Computational Approaches for Analyzing Weblogs. pp. 199–205 (2006)