INAOE's participation at PAN'13: Author Profiling task

Notebook for PAN at CLEF 2013

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

[1] Department of Computer Science,
Instituto Nacional de Astrofísica, Óptica y Electrónica, México
{pastor, mmontesg, hugojair, villasen}@ccc.inaoep.mx
[2] Information Technologies Department,
Universidad Autónoma Metropolitana-Cuajimalpa, México
evillatoro@correo.cua.uam.mx

Abstract This paper describes the participation of the Laboratory of Language Technologies of INAOE at PAN 2013 evaluation lab. We adopted second order representations for facing the problem of Author Profiling (AP). This representation tackles two shortcomings of the typical Bag-of-Terms: i) the sparsity and high dimensionality of document representations, and ii) the assumption of total independence between terms in documents. In order to overcome these problems the proposed representation builds document vectors in a space of the different profiles, which represent the relationships of each document with the different profiles (say, age and gender). In order to evaluate our approach, we compare the proposed representation against a standard Bag-of-Terms representation using the PAN 2013 corpus for AP. We found that the second order attributes using a low computational cost, show evidence of being useful to determine genre and age profile.

1 Introduction

The Author Profiling (AP) task consists in knowing as much as possible about an unknown author, just by analyzing a given text [2]. The interest in AP task is growing in recent years, this is due, in part, to the huge amount of information in plain text available on internet. In this context, several applications related to AP are emerging, some of them have to do with business intelligence, computer forensics and security. One way to address the AP task is to approach it as a single-label multiclass classification problem, where profiles represent the classes to discriminate.

The representation of documents is a key procedure for AP. Currently, one of the most common approaches for document representation is the Bag of Terms (BOT). BOT representation builds feature vectors of documents, taking each term in the vocabulary as an attribute. However, BOT like representations have some drawbacks:

1. *Terms are considered independent of other elements in the problem:* We believe that valuable information that may help to deal with the AP problem is being ignored. In this context, we propose taking into account relationships between profiles and terms.

2. *High dimensionality and high sparsity of vectors:* both affect the representation and the performance of the classification algorithms, and could be impractical in some situations. In this way, we focus in a representation based in second order attributes rich in representativeness, which represents relations with each profile.

In summary, to overcome the above issues we propose to use a low dimensional representation with high level of representativeness. In this way, in our proposal we follow some ideas from Concise Semantic Analysis (CSA) [3] to achieve relationships between documents and profiles. Thus, our approach intends to exploit the use of second order attributes for the AP task. For this, we use: i) the term frequency *tf* weighting scheme in order to capture the use of the stylistic terms (e.g., stopwords, puntuation marks, etc.), and ii) we provide an effective normalization to deal with the high imbalanced data.

In summary, The rest of this paper is organized as follows: Section 2 introduces the proposed representation, Section 3 explains how we performed the experiments and the results we obtained, finally Section 4 shows our conclusions.

2 Second order attributes for Author Profiling

From a general point of view, the proposed representation is built through two main stages: i) To build term vectors in a space of profiles, and ii) To build document vectors in a space of profiles. The rest of this section explains both steps in detail.

2.1 Term Representation

Estimating the relationships between each term and profiles is the first step to get the second order attributes. In this way, it is necessary to construct a vector representation for each term. Terms could be any textual unit used as document feature, for example, words, n-grams, punctuation marks, etc.

The main idea behind this first step is to capture the relation that each term maintains with different profiles. In other words, we compute a value that shows how a term t_j is used in each profile p_i. Let $\{t_1, \ldots, t_m\}$ denote the vocabulary in the collection, and $\{p_1, \ldots, p_n\}$ be the set of profiles to be analyzed. For each term t_j in the vocabulary, we build a term vector $\mathbf{t_j} = \langle tp_{1j}, \ldots, tp_{nj} \rangle$, where tp_{ij} is a real value representing the relationship of the term t_j with the profile p_i. For computing tp_{ij} we mainly take into account those documents that belong to the profile p_i. The relationship of a term with a profile considers the relative term frequency just in the documents of this profile. Thus, high frequencies will show more preference for the term in a given set of documents. Equation 1 follows the above idea and computes a relative weight as:

$$w_{ij} = \sum_{k:d_k \in P_i} \log_2 \left(1 + \frac{tf_{kj}}{len(d_k)}\right) \tag{1}$$

where P_i is the set of documents that belong to profile p_i, tf_{kj} is the number of occurrences of the term t_j in the document d_k, and $len(d_k)$ is the length of the document

d_k. The function in equation 1 is to soften the most frequent terms of the corpus. Finally, we apply a simple normalization (Equation 2.1) for computing tp_{ij}. Note that this normalization takes into account the weights computed for other profiles, causing each weight being relative to all profiles.

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

It is worth noting that, until this step, the second order attributes are sensible to highly unbalanced data (as the PAN13 corpus). That is, the relation of each term with each profile attribute will be higher for larger classes just because those classes have more documents and then more occurrences of certain terms. For that reason, we apply a simple but effective normalization over each profile in order to consider the proportion of the term in each profile. Equation 2.2 shows the latter idea. Note that this normalization takes into account the weights computed for other terms, causing each weight being relative to all terms.

2.2 Document Representation

After computing term vectors in a space of profiles, we build relationships between documents and profiles; these are the second-order attributes. We compute these adding term vectors of the terms contained in the documents. In this way, we will have documents represented as $\mathbf{d_k} = \langle dp_{1k}, \ldots, dp_{nk} \rangle$, where n is the total number of existing profiles, and dp_{ik} is a real value representing the relationship of the document d_k with the profile p_i. Additionally, each term vector, before being added, is weighted by the relative frequency of the term t_j in the document d_k. Equation (3) shows the above ideas.

$$\boldsymbol{d}_k = \sum_{t_j \epsilon D_k} \frac{tf_{kj}}{len(d_k)} \times \boldsymbol{t}_j \tag{3}$$

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

3 Experimental Results

We have approached the AP problem as a single-labeled six class classification problem. This means, that we have six *age-genre* profiling classes: *10s-female, 10s-male, 20s-female, 20s-male, 30s-female, 30s-male*. Given this context, we use for each experiment the following configuration: i) a stratified 10 cross fold validation using the training PAN13 corpus, ii) the most 50,000 frequent terms, and iii) a LibLINEAR classifier [1]. For terms we use words, contractions, words with hyphens, punctuation marks and a set of common emoticons. From Table 1 it can be seen how the Second Order Attributes (SOA) outperforms the BOT representation using the PAN 13 corpus, which is an imbalanced dataset (a realistic scenario). We believe this is because the second order

attributes provides a different document perspective beyond the isolated word frequencies. In Table 1 we also show the detailed results for predicting *Age* and *Genre* in the test dataset, and a summary of the averaged results for all participants at PAN 2013.

Detailed classification accuracy										
	Training data				*Test data*			*Averaged results for all participants*		
	SOA			BOT	SOA			AVG		
	Gender	Age	Total	Total	Gender	Age	Total	Gender (st.dv.)	Age (st.dv.)	Total (st.dv.)
English	61.3	63.7	**41.9**	36.6	56.90	65.72	**38.13**	53.76 (3.33)	53.51 (12.50)	28.99 (7.42)
Spanish	70.5	72.7	**54.8**	41.9	62.99	65.58	**41.58**	55.41 (4.99)	49.04 (14.15)	27.67 (9.35)

Table 1. Experiments using the Second Order Attributes (SOA) and BOT computed over the 50,000 most frequent terms on the datasets. We denote in bold our best outcomes for each dataset.

Results in Table 1 demonstrate the performance of our proposal, which overcomes the conventional BOT and holds the first position for both languages (averaged accuracy), and second position for each one. We think this is because SOA are less sensitive to the high dimensionality problem, the scarce data, and the imbalanced classes. Moreover, it is worth knowing that our approach took only 0.22% of the time required by the method in one position below for english corpus (based in accuracy), which means that our proposal was one of the most efficient and effective approaches at PAN 2013.

4 Conclusions

In this paper we have explored a new document representation for AP task. To the best of our knowledge, this is the first time that AP is addressed using attributes that represent relationships with profiles. We found that, with very low computational cost our proposal can build discriminative low dimensional dense vectors for AP. Using these vectors, the classifier can keep good classification rates, even for imbalanced data. We think that this is due to the relations among terms and profiles, which provides few but high predictive attributes. We also presented experimental results that show better performance of the proposed approach against the standard BOT. We further believe that the proposed representation is a feasible and stable representation, quite practical in situations where it is necessary to represent and classify fast and effectively.

References

1. 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)
2. Koppel, M., Argamon, S., Shimoni, A.R.: Automatically categorizing written texts by author gender. Literary and Linguistic Computing 17(4), 401–412 (2002)
3. Li, Z., Xiong, Z., Zhang, Y., Liu, C., Li, K.: Fast text categorization using concise semantic analysis. Pattern Recognition Letters 32(3), 441–448 (2011)