Authorship Identification with Modality Specific Meta Features

Notebook for PAN at CLEF 2011

Thamar Solorio[♮], Sangita Pillay[♮], and Manuel Montes-y-Gómez[♮†]

[♮]University of Alabama at Birmingham, USA.
[†] National Institute of Astrophysics, Optics and Electronics, Mexico
{solorio,rsangita,mmontesg}@cis.uab.edu

Abstract This paper presents the approach used in the PAN '11 authorship identification competition. Our method extracts meta features from several independently generated clustering solutions from the training set. Each clustering solution uses a disjoint set of features that represent a specific linguistic modality. The different clustering solutions encode similarities in writing styles of authors across specific dimensions. The final classifier is trained with a combination of the meta features with first level features. Our approach has outperformed a more syntactic oriented state-of-the-art method on web forum data. We achieved moderately successful results on this PAN competition, with better results on the test set with a smaller number of authors. However, considering that our system was not fine tuned for the PAN evaluation data we found our results very encouraging.

1 Introduction

Authorship Identification (AI) assumes the existence of individualized and identifiable writing styles. The success of previous work empirically supports this assumption, at least on the collections that have been used to test these approaches. However, when looking at writing samples from different authors it is clear that similarities exist among them. For instance, when analyzing web forum data we can see that several authors share emoticon patterns, even the absence of emoticons is in itself a pattern shared by many authors. Similarly, other authors use punctuation marks in similar ways (some like to use more than one exclamation point to highlight emotion) or tend to use similar words when writing about the same topic. Our modality specific meta feature approach was motivated by these observations. The idea is to extract these similarities across authors in a modality specific way. We refer to the different linguistic dimensions we analyze from the documents (syntactic, lexical, stylistic) as "modalities". By contrasting authors' characteristics in a modality specific way, we allow authors to share patterns with subsets of authors along a specific dimension (e.g. emoticons) while sharing writing patterns with a different subset of authors across another dimension (e.g. similar use of adverbs). These similarities are extracted by independently clustering, in an unsupervised way, the training instances using the subset of features from each modality. The combination of the meta features generated by our approach with the first level features outperformed the method presented in [9] on the task of AA on web forum data.

On the PAN competition, our system reached a micro-averaged F-measure of 14.8% on the large test set, and 44% F-measure on the small test set. Even though these results were not the best results achieved on the competition, we believe they are very positive results as our system was not tuned for the PAN task. It was applied as it was developed for the web forum data. We believe that better results can be achieved by our system using a set of features and modalities adapted for the data of the PAN task.

2 Previous Work

Following Stamatatos' survey [12], we can categorize AI approaches by the way the training set is processed. In a *profile-based approach* the training instances (text from each author) are concatenated into a single large document per author. Training under the profile-based approaches consists of extracting an author profile from the concatenated document. The final prediction in this scheme is based on measuring similarity between a test profile and the training profiles generated. Examples of recent work using a profile based approach include [5,6,3,4]. The major advantage of these approaches over the instance based ones is scalability.

Instance-based approaches, on the other hand, follow the traditional framework of text classification, where each text sample in the training set is an instance of the problem. In this setting, the samples are typically represented individually by a feature vector, then a learning algorithm will be trained on this set of vectors. This approach has been successfully used in combination with a wide variety of learning algorithms, such as Support Vector Machines [2,11], decision trees [15], and memory based learners [7]. The main difference with a profile-based approach is the need for a sufficiently large number of samples with known authorship.

As Stamatatos shows, both approaches have strengths and weaknesses [12] and the choice of what to use may depend on the application, the amount of available data, computer resources, and the like. In this paper we use an instance-based approach with Support Vector Machines as the underlying algorithm. Our modality specific meta features approach is different from previous machine learning approaches to AI in that it has an intermediate step where we generate meta features from clustering the training instances per modality. Solorio et al. have shown empirically that using these meta features results in higher prediction accuracy [11] than that achieved on using only first level features.

3 Meta Features for Authorship Identification

Our approach starts with the extraction of first-level features to generate a feature vector representation for each instance. However, we generate m smaller vectors, each containing features from a specific modality describing the instances. Typical instance-based approaches extract a single feature vector for $\mathbf{x} \in R^n$. We call these subvectors modality specific because they represent different characteristics of the author's text. Each modality refers to a particular linguistic dimension, such as lexical, syntactic, or stylistic. More formally, an instance $\mathbf{x}$ is now represented as $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m\}$ where each $\mathbf{x}_i$ is a vector with $|\mathbf{x}_i|$ features in modality i. Note that $\textbf{union}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m) = \mathbf{x}$ and

intersection$(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m) = \emptyset$ since we are only generating sub vectors (or complementary views) from the original feature set.

Each set of the m different vectors are input to a clustering algorithm to produce m clustering solutions for the training data with k clusters each. As a result, we end up with different arrangements of the training instances into clusters, one arrangement (clustering solution) per modality. From each cluster c_j in each of the m clustering solutions, we compute a centroid by averaging all the feature vectors in that cluster.

$$\textbf{centroid}_{m_j} = \frac{1}{\mid c_{m_j} \mid} \sum_{x_i \in c_{m_j}} \mathbf{x}_i \tag{1}$$

where j above ranges from 1 to k, the number of clusters. We generate meta features by computing the *similarity* of each instance to these centroids using the cosine function. We compute these similarities for training and testing instances. Each instance **x** is now represented by the original set of first level features $\langle \mathbf{x}_{i_1}, ..., \mathbf{x}_{i_{|x_i|}} \rangle$ in combination with the meta features $\langle \mathbf{x'}_{i_1}, ..., \mathbf{x'}_{i_k} \rangle$ generated for each modality j.

We consider four types of first level features: stylistic (sty), lexical (lex), syntactic (syn), and perplexity (ppl) values from character 3-gram language models. That is, in these experiments $m = 4$. Therefore, in our problem we have $\mathbf{x} = \{\mathbf{x}_{sty}, \mathbf{x}_{lex}, \mathbf{x}_{ppl}, \mathbf{x}_{syn}\}$.

The intuition behind our approach is to generate new meta features from clustering the data that can represent the relation, i.e. closeness, between documents from one author and documents from other authors. Thus, no class information is used during clustering, as the idea is to uncover regularities across the documents from authors on individual modalities. New in this work as well is the idea of a modality specific clustering, where each linguistic dimension is clustered separately. Our assumption is that generating clusters by looking at feature subsets separately allows to disentangle authors' characteristics that may be blurred away when clustering the entire feature vectors at once. We expect this information will be captured by the meta features, and will yield higher classification accuracy than the first level features by themselves.

3.1 First Level Features

Table 1 shows a list of the features we used arranged by modality. These are exactly the same features used in Solorio et al. (2011) where the task focuses web forum data [11]. In the *stylistic* modality we include features tuned for written interactions in social networks. We use percentages of non-alphanumeric characters that are commonly used in emoticons. We also include percentages of capitalized words, use of quotations, and use of signature, that we believe allow writers more freedom to express their unique writing style. The *lexical* modality contains the standard bag of words representation used in text classification that has also been commonly used in previous AI work [1,15]. In the *perplexity* modality we use perplexity values from character 3-gram language models. We use the training data to train one language model per author and each model generates a perplexity value per instance. For training the language models and computing perplexity values we used the SRI-LM toolkit [13]. Lastly, in the *syntactic* modality we have unigrams, bigrams, and trigrams of POS tags, and typed dependency relations extracted using the Stanford parser [8], that have been used before in AA.

Modality	Features
Stylistic	Total number of words
	Average number of words per sentence
	Binary feature indicating use of quotations
	Binary feature indicating use of signature
	Percentage of all caps words
	Percentage of non-alphanumeric characters
	Percentage of sentence initial words with first letter capitalized
	Percentage of digits
	Number of new lines in the text
	Average number of punctuations (!?.;:,) per sentence
	Percentage of contractions (won't, can't)
	Percentage of two or more consecutive non-alphanumeric characters
Lexical	Bag of words (freq. of unigrams)
Perplexity	Perplexity values from character 3-grams
Syntactic	Part-of-Speech (POS) tags
	Dependency relations
	Chunks (unigram freq.)

Table 1. Feature breakdown by modality

As mentioned earlier, the features described above were selected based on our belief that they would be useful in the domain of electronic web forum data. We expect that some of these features will be relevant for the PAN task, but no customization of them was done so it is also expected that many of the features in Table 1 will not be that useful.

4 Experimental evaluation

We are presenting results of using Support Vector Machines (SVMs) [10] as the underlying learner as implemented in WEKA [14]. Our modality specific meta features approach can be used in combination with any machine learning algorithm. However since in previous experiments SVMs have shown to yield competitive results, we chose this algorithm for the PAN competition. As described in Section 3, we cluster each of the four types of feature vectors in the training data set separately. We use a k-means clustering algorithm, implemented in CLUTO. The first step in the clustering phase is to choose the number of clusters. Since in our previous experiments we obtained better results with $k =$number of authors $\times 15$, we used this value to fix the parameter k in these experiments. Therefore, for the Large test set $k = 72 \times 15$, and for the small data set $k = 26 \times 15$. It should be noted that since our system cannot handle the out-of-training author scenario, we did not submit results for those test sets.

In Table 2 we present results as reported by the PAN organizers. The baseline system we are presenting here consists of training an SVM classifier using only the first level features. On the Large data set our modality specific meta features (MSMF) approach outperforms the baseline system on the macroaveraged results, but the baseline system reaches higher microaveraged results. On the Small test set, our MSMF approach yields better results than the baseline on all measures reported. These results follow the trend we found in our previous experiments where adding the MSMF information increased prediction accuracy of the system. The overall performance reached in these test sets is much lower than what we saw on different data sets [11]. We believe this drop in

prediction accuracy is due mainly to a lack of customization of our system to the domain of the competition. For instance, we did not make any changes to the features or modalities, and we also did not adapt our preprocessing step. Lastly, the smaller training set available can also be a contributing factor to the weak performance we achieved. In our previous experiments the smallest number of documents per author was 165, higher than the average number of training documents at the PAN competition.

System	TestSet	MacroAvg Precision	MacroAvg Recall	MacroAvg F1	MicroAvg Precision	MicroAvg Recall	MicroAvg F1
Baseline	Large	0.119	0.054	0.041	0.155	0.155	0.155
Baseline	Small	0.440	0.152	0.148	0.384	0.384	0.384
MSMF	Large	0.171	0.084	0.066	0.148	0.148	0.148
MSMF	Small	0.415	0.205	0.185	0.440	0.440	0.440

Table 2. Comparison of micro and macro averaged precision, recall, and F1 values in two PAN'11 test sets. MSMF stands for our modality specific meta features approach.

5 Conclusions

We have described the modality specific meta features approach we used to submit results at the PAN '11 competition. The main idea behind our approach is to explicitly exploit the fact that authors share similarities in writing patterns across different linguistic dimensions. These similarities are encoded in the meta features and are the result of several independent clustering solutions of the training instances. The results obtained at the competition are encouraging and support our claim that adding higher level features is beneficial for the AI task. Our goal is to improve these results by modifying the feature set to better represent the domain of interest.

We are also currently working on characterizing the effect of the meta features, and exploring the combination of this approach with a profile-based approach. So far, we have used only instance-based approaches to AA. We would like to evaluate the benefit of including our modality-specific meta features into a profile based approach, such as the work by [2].

Acknowledgements. This work was partially supported by a UAB faculty development grant and by the UPV, award 1932, under the program Research Visits for Renowned Scientists (PAID-02-11) to the first author. It was also supported in part by the CONACYT-Mexico (project no. 134186) and by the European Commission as part of the WIQ-EI project (project no. 269180) within the FP7 People Programme.

References

1. S. Argamon and S. Levitan. Measuring the usefulness of function words for authorship attribution. In *Proceedings of the Conference of the Association for Computers and the Humanities and the Association for Literary and Linguistic Computing*, 2005.

2. Hugo J. Escalante, Thamar Solorio, and Manuel Montes. Local histograms of character n-grams for authorship attribution. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 288–298. Association for Computational Linguistics (ACL), 2011.
3. P. Joula. Authorship attribution. *Foundations and Trends in Information Retrieval*, 1(3):233–Ű334, 2006.
4. V. Keselj, F. Peng, N. Cercone, and C. Thomas. N-gram based author profiles for authorship attribution. In *Proceedings of the Pacific Association for Computational Linguistics*, pages 255–264, 2003.
5. M. Koppel, J. Schler, and S. Argamon. Computational methods in authorship attribution. *Journal of the American Society for Information Science and Technology*, 60:9–26, 2009.
6. Maarten Lambers and Cor J. Veenman. Forensic authorship attribution using compression distances to prototypes. In Z. J. M. H. Geradts, K. Y. Franke, and C. J. Veenman, editors, *IWCF 2009*, volume LNCS 5718, pages 13–24, 2009.
7. Kim Luyckx and Walter Daelemans. Shallow text analysis and machine learning for authorship attribution. *Proceedings of the Fifteenth Meeting of Computational Linguistics in the Netherlands (CLIN)*, pages 149–160, 2005.
8. M.C. De Marneffe, Bill Maccartney, and Christopher D. Manning. Generating typed dependency parses from phrase structure parses. In *LREC 2006*, 2006.
9. Sindhu Raghavan, Adriana Kovashka, and Raymond Mooney. Authorship attribution using probabilistic context-free grammars. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 38–42, Uppsala, Sweden, July 2010. Association for Computational Linguistics.
10. Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, 2002.
11. Thamar Solorio, Sangita Pillay, Sindhu Raghavan, and Manuel Montes y Gómez. Generating metafeatures for authorship attribution on web forum posts. In *5th International Joint Conference on Natural Language Processing, IJCNLP-2011*, (to appear).
12. Efstathios Stamatatos. A survey on modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, 60(3):538–556, 2008.
13. Andreas Stolcke. SRILM - an extensible language modeling toolkit. pages 901–904, 2002.
14. Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kauffmann, 2nd edition, 2005.
15. Y. Zhao and J. Zobel. Effective and scalable authorship attribution using function words. In *Proceedings of 2nd Asian Information Retrieval Symposium*, volume 3689 of *LNCS*, pages 174–189, Jeju Island, Korea, 2005.