Author Identification Using Semi-supervised Learning
Notebook for PAN at CLEF 2011

Ioannis Kourtis and Efstathios Stamatatos

University of the Aegean
83200 – Karlovassi, Samos, Greece
{ikourtis; stamatatos}@aegean.gr

Abstract. Author identification models fall into two major categories according to the way they handle the training texts: profile-based models produce one representation per author while instance-based models produce one representation per text. In this paper, we propose an approach that combines two well-known representatives of these categories, namely the Common *n*-Grams method and a Support Vector Machine classifier based on character *n*-grams. The outputs of these classifiers are combined to enrich the training set with additional documents in a repetitive semi-supervised procedure inspired by the co-training algorithm. The evaluation results on closed-set author identification are encouraging, especially when the set of candidate authors is large.

1 Introduction

Nowadays, there is a rapid growth of text in electronic form in blogs, social media, forums, etc. Most of this content is provided anonymously or under unverified names. In the framework of forensic applications it is needed to group texts written by the same author or track texts written under different names but belonging to the same person. Moreover, there are numerous copyright dispute cases where multiple people claim the authorship of texts. Authorship identification supported by computational analysis of texts attracts increasing attention since it may offer quick answers to these problems [11].

The vast majority of approaches to author identification consider this problem as a closed-set classification task. That is the training set includes samples for all possible authors and each text of unknown authorship has to be assigned to one candidate author. However, in many practical applications it is not possible to know a priori all the candidate authors or it is not possible to have sample texts for all of them. Therefore, a more practical but less studied setting is the open-set classification where the classifier is allowed to answer "I don't know" for some texts of unknown authorship [6]. In addition, the vast majority of author identification methods assume that the only available information for building a classification model comes from a fixed and stable training set. However, there are many cases where we need to decide about the authorship of groups of texts. Alternatively, a long text (a book) of unknown authorship can be segmented into multiple parts. In such cases, it is possible to use the test sets as unlabeled examples and use some information from them to improve the classification models. An attempt to this direction was proposed by Guzman et al. [2] where unlabeled examples from the Web were used to enrich the training set.

In automated author identification we need a set of candidate authors and text samples for each one of them. Since we care more about style rather than topic, one main task is to adequately measure the stylistic choices of the authors. To this end, several text representation methods have been proposed [11] related to lexical information (e.g., function word frequencies), character information (e.g., character *n*-grams), syntactic information (e.g., part-of-speech frequencies), and semantic information (e.g., synonyms). In addition, application-specific features can be used when all texts are of the same type, format, or topic. A number of independent studies have found that character *n*-grams are very effective in author identification [6,8,9]. Moreover, they are language-independent features and their extraction requires minimal text processing.

Author identification methods fall into two major categories [11]:

- The *profile-based paradigm* attempts to capture the style of authors. It disregards the differences between training texts by the same author and produces one single representation (i.e., profile) per author. Each text of unknown authorship is then compared with the profile of each author and is assigned to the most likely one.
- The *instance-based paradigm* attempts to capture the style of texts. It produces one representation per training text and builds a classification model that can estimate the most likely author of a text of unknown authorship. The machine learning algorithms used in this paradigm (e.g. SVM, neural networks, etc.) usually require multiple instances per class, so in case there is only one training text for one author it should be segmented into smaller pieces.

Each of these paradigms has its strengths and weaknesses. Profile-based approaches are more robust when there is an uneven distribution of training texts in the candidate authors (i.e., the class imbalance problem) [8]. On the other hand, instance-based approaches are more accurate when there are enough training texts for all the candidate authors. Profile-based approaches can better handle very short texts since they concatenate all the texts by the same author. On the other hand, in instance-based approaches it is easier to combine different text representation features and they are more robust when the candidate authors set size is large.

In this paper we propose an algorithm that combines two well-known representatives of these paradigms: the Common N-Grams (CNG) method [4] and an SVM classifier using character *n*-grams [10]. The main idea is to combine the outputs of these classifiers in the test set and augment the training set with additional documents. Therefore it is a semi-supervised approach since it uses both labeled and unlabeled examples. The idea of using a couple of independent classifiers in a repetitive semi-supervised procedure is inspired by the co-training algorithm [1].

The rest of this paper is organized as follows. The next section describes in detail the proposed algorithm. In Section 3 we report evaluation results based on the PAN-11 training corpus. In Section 4 the main conclusions drawn by this study are described and future work directions are given.

2 The Proposed Method

In this paper, we describe an author identification method that can be applied to closed-set tasks and is based on semi-supervised learning. Our approach combines two well-known approaches: the CNG model and the SVM model. Both models use character 3-grams to represent the stylistic properties of texts. We first describe these models and then the proposed semi-supervised algorithm is presented.

2.1 Common n-grams

CNG [4] is a profile-based method, that is, first all the available training texts per author are concatenated into one file and, then, a single representation is extracted for each author. The representation (profile) is based on the *L* most frequent character *n*-grams of the file. The same representation is used for each individual text of unknown authorship. The classification model is based on the distance (dissimilarity) of the profile of the text from each of the profiles of the candidate authors. This procedure is illustrated in Figure 1 (for just one candidate author).

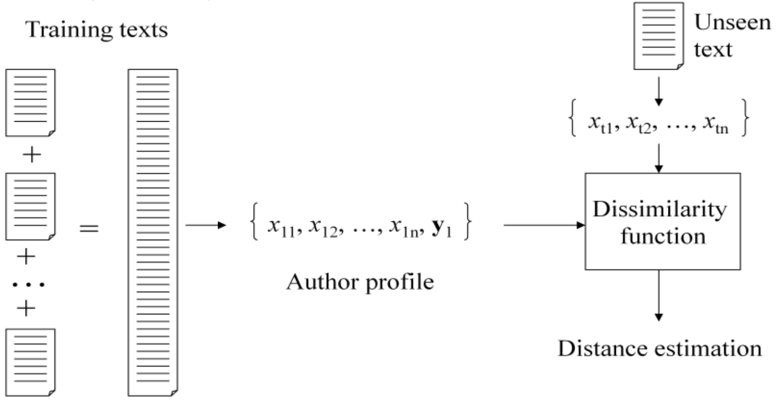
Training texts
Unseen text
$\{x_{t1}, x_{t2}, \ldots, x_{tn}\}$
$\{x_{11}, x_{12}, \ldots, x_{1n}, \mathbf{y}_1\}$
Author profile
Dissimilarity function
Distance estimation

Fig. 1. Overview of the CNG method (only one candidate author is shown).

This method has two significant parameters that should be tuned: n, that is the order of *n*-grams, and L, that is the size of profile. According to previous studies [9, 10], we selected *n*=3 since it provided good results in authorship attribution and it is less likely to capture thematic information in comparison with longer *n*-grams. On the other hand, character 3-grams cannot easily capture contextual information (i.e., sequences of words). The profile length L should be selected carefully and in combination with the dissimilarity function since previous studies have shown that CNG may be unstable when the profile of one candidate author is shorter than L. This may happen when there are very limited training texts for one candidate author (i.e., the class imbalance problem). In this study, we used the following dissimilarity function [9] that is stable when *L* increases:

$$d_1(P(x), P(T_a)) = \sum_{g \in P(x)} \left(\frac{2(f_x(g) - f_{T_a}(g))}{f_x(g) + f_{T_a}(g)} \right)^2$$

where $P(x)$ and $P(T_a)$ are the profile of the text of unknown authorship and the profile of the candidate author a, respectively, while $f_x(g)$ and $f_{Ta}(g)$ are the normalized frequencies of the *n*-gram g in the text of unknown authorship and the concatenated training texts of candidate author a. Since the sum is defined over the *n*-grams that belong to the profile of the text of unknown authorship, the dissimilarity function will contain the same number of terms for each candidate author, so it will be more stable in case of class imbalance.

2.2 Support Vector Machines

SVM is one of the most effective machine learning algorithms for text categorization. In authorship attribution, it has been used in combination with character *n*-grams providing very good results [10]. Essentially, it is an instance-based method that is for each individual training text a representation is produced, usually based on the frequencies of the *d* most frequent character *n*-grams of the training corpus. Then, the SVM algorithm can be used to learn the boundaries between classes (i.e., authors). The learned model can then be used to guess the author of another text. This procedure is demonstrated in Figure 2. In this paper, we used the LIBSVM implementation of this algorithm. Since the dimensionality is high (several thousands of features), the linear kernel has been used. Moreover, we used character 3-grams as in the case of CNG.

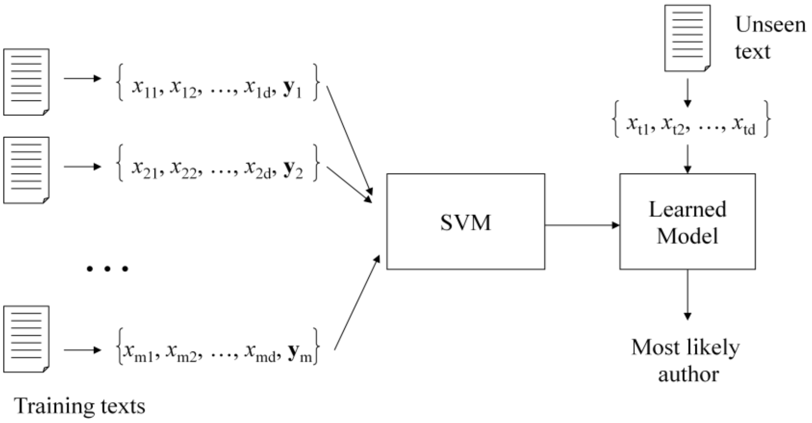
Unseen text
$\{x_{11}, x_{12}, \ldots, x_{1d}, \mathbf{y}_1\}$
$\{x_{21}, x_{22}, \ldots, x_{2d}, \mathbf{y}_2\}$
$\{x_{m1}, x_{m2}, \ldots, x_{md}, \mathbf{y}_m\}$
$\{x_{t1}, x_{t2}, \ldots, x_{td}\}$
SVM
Learned Model
Most likely author
Training texts

Fig. 2. Overview of the SVM method.

One crucial decision when using this method is about *d*, the dimensionality. It has been proved that the most frequent character *n*-grams are good style markers [3, 5] but it is not clear how many character *n*-grams should be used. In this paper, we propose the use of the *intrinsic dimension* as a criterion to define the appropriate dimensionality of the text representation. More specifically, in many cases high-dimensional datasets can be efficiently summarized in a space of much lower dimension without losing much information. This is especially true for text representation since many features correlate. Intrinsic dimension provides an estimation of the variables we need to represent the high-dimensional data. Therefore, intrinsic dimension can be used to indicate the richest representation. That is, the higher the intrinsic dimension, the better the representation of texts (it captures more hints about their properties). So, given a training corpus, we sort the character *n*-grams in decreasing frequency of appearance and then a frequency threshold can be applied to select the features of the representation. By varying this threshold, we get varying sizes of *d*. For each threshold value, we measure the intrinsic dimension and the representation that corresponds to the maximal intrinsic dimension value is selected. The maximum likelihood estimator [7] was used for the intrinsic dimension.

3.3 The Semi-supervised Learning Algorithm

The main idea of the proposed algorithm is inspired by co-training [1] where two independent classifiers are used and help each other with their predictions on the set of unlabeled examples. Given a set of training documents (labeled examples) and a set of test documents (unlabeled examples) our algorithm repetitively selects some members of the test set and adds them to the training set. In each step, the CNG and the SVM methods are trained based on the training set and the acquired models are used to predict the labels of the test set. Then, the test documents that both classification models agree on their predicted label are selected. Moreover, the text size of these documents should be larger than a threshold since in general it is hard to capture the stylistic properties of very short texts. In other words, we consider that even when CNG and SVM agree on their predictions, these predictions are unreliable when the text length is very short. When at least one test text is selected and added to the training set using the predicted label as its true label, the procedure is repeated. When this repetitive procedure stops and the test set is not empty, one of the classifiers can be used to predict the labels of the rest of the texts of test set. In this paper, we used the SVM as this default classifier since it is more reliable when there are enough training data. In other words, it is expected that after a few repetitions the training set will be enriched with new documents. If this is not true, and the training set still under-represents some of the candidate authors, the CNG classifier would be a better choice.

The proposed algorithm is shown in Figure 3. Note that there are important differences with the co-training algorithm. First, the original co-training algorithm used the same classification method and two distinct subsets of the feature set to produce the two independent classifiers. In the proposed algorithm, we use the same feature types (i.e., character 3-grams) and use two different classification models representing the two basic families of author identification methods (i.e., profile-based and instance-based). Moreover, the proposed algorithm uses the unlabeled cases where the outputs of the two classifiers agree. In co-training a fixed number of the most confident answers from each classifier are considered. In the proposed algorithm the whole test set is examined. On the other hand, co-training examines a subset of the test set in each repetition. As a result, co-training needs more repetitions.

```
% Input: A training set, a test set, and a text-length threshold
% Output: A set of labels for the test set
author_predict(TrainSet,TestSet,Threshold)
{  found = 1;
   while TestSet ≠ Ø AND found == 1
      found = 0;
      CNG_model = train_CNG(TrainSet);
      SVM_model = train_SVM(TrainSet);
      for text ∈ TestSet
         CNG_label = test_CNG(text,CNG_model);
         SVM_label = test_SVM(text,SVM_model);
         if CNG_label = SVM_label AND size(text) > Threshold
            PredictedLabels = PredictedLabels ∪ [text, SVM_label];
            TrainSet = TrainSet ∪ [text, SVM_label];
            TestSet = TestSet - text;
            found = 1;
         end-if
      end-for
   end-while
   for text ∈ TestSet
      PredictedLabels = PredictedLabels ∪ [text, SVM_label];
   end-for
   return PredictedLabels;
}
```

Fig. 3. The proposed semi-supervised learning algorithm.

3 Evaluation

For the evaluation of the proposed method we used the corpora released in 2011 for the evaluation campaign on author identification in the framework of PAN-2011[1]. In more detail, we used 2 parts of these corpora that correspond to the closed-set evaluation setting, namely, PAN11-AA-Small and PAN11-AA-Large. The former includes 3,519 texts from 26 candidate authors (roughly, 85% in the training corpus and 15% in the test corpus) while the latter comprises 10,635 texts from 72 candidate authors (roughly, 88% in the training corpus and 12% in the test corpus). All the texts are parts of email messages and therefore can be short and messy. The size of the messages varies from 23B to 23KB. Both training corpora are highly imbalanced as can be seen in Figures 4 and 5.

[1] http://www.uni-weimar.de/medien/webis/research/events/pan-11/author-identification.html

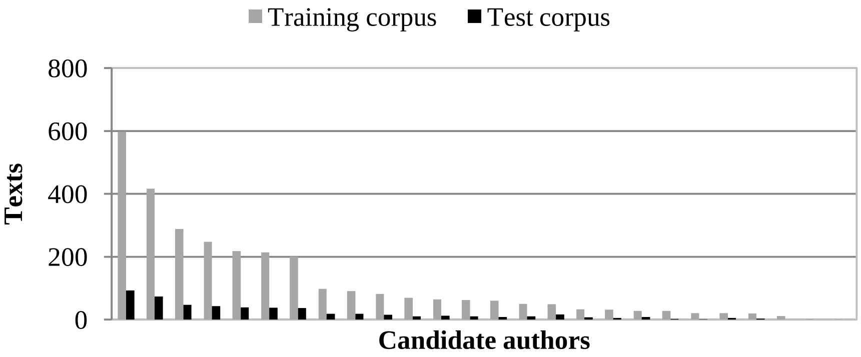

Fig. 4. Distribution of the training/test corpus of the PAN11-AA-Small.

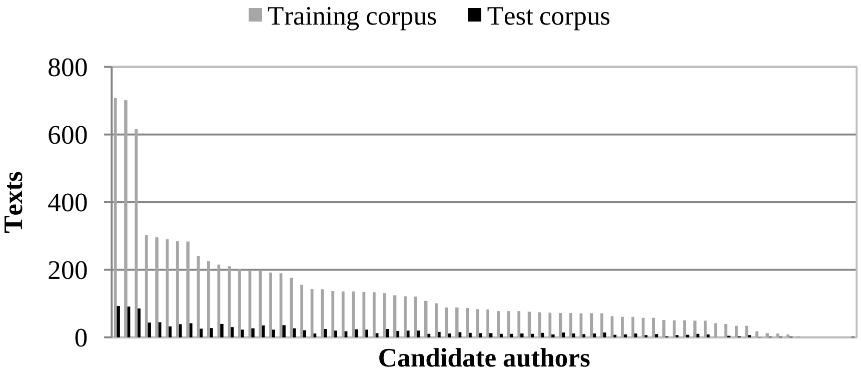

Fig. 5. Distribution of the training/test corpus of the PAN11-AA-Large.

The CNG parameters used for these corpora were n=3 and L=3,000. For the SVM method we used n=3 (character 3-grams) and d (dimensionality) is determined by the maximal value of the intrinsic dimension. Figure 6 shows the intrinsic dimension values we get with different frequency threshold values for PAN11-AA-Small and PAN11-AA-Large. In the former case, the intrinsic dimension is maximized for frequency threshold=80 (meaning that all character 3-grams appearing at least 80 times in the training corpus are included in the feature set). In the latter case, the intrinsic dimension is maximized for threshold=20. Therefore, the feature set of PAN11-AA-Large is significantly larger than the feature set of PAN11-AA-Small.

The final parameter is the text size threshold used to select files that will be added in the training set according to our semi-supervised algorithm. Figure 7 shows the distribution of text-size of the test corpus of PAN11-AA-Large where the predictions of CNG and SVM agree and they correspond to the true author of the texts. Also, it shows the cases where the common predictions do not correspond to the true authors. It is evident that a size threshold of 500 bytes excludes most of the cases where the two models agree but the predicted author is not the correct answer.

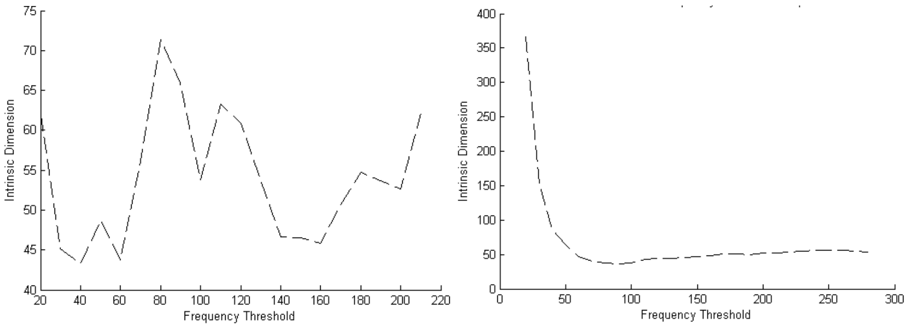

Fig. 6. Intrinsic dimension of PAN11-AA-Small (left) and PAN11-AA-Large (right) for varying freq. threshold.

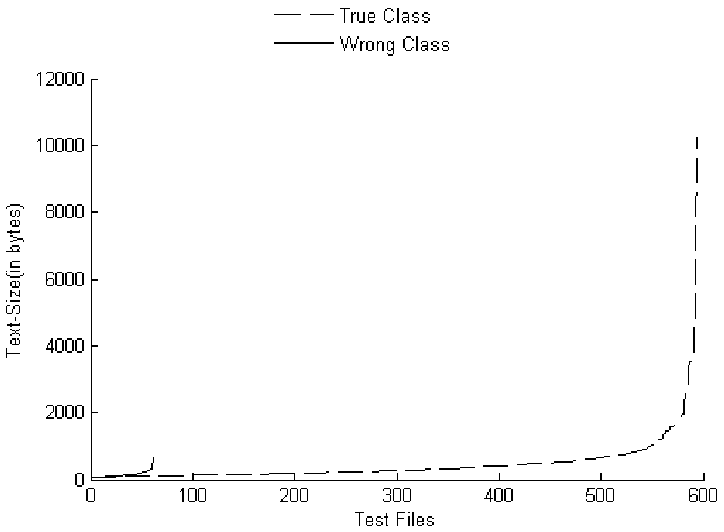

Fig. 7. Text-size distribution of PAN-11-AA-Large test corpus where CNG and SVM predictions agree.

For the PAN11-AA-Small corpus, using the initial training set CNG and SVM models achieved 49.2% and 64.7% microaverage accuracy, respectively. During the semi-supervised learning procedure 57 files were moved from the test set to the training set. For these files both CNG and SVM agreed on their predictions (94.7% accuracy). For the rest of the test files and using the enriched training set, the performance of CNG and SVM models was 43.6% and 61.2% accuracy, respectively. It is obvious that SVM is far better than CNG in this dataset so it was used as the default classifier.

For the PAN11-AA-Large corpus, the CNG and SVM models were trained based on the initial training set and their performance (microaverage accuracy) was 37.8% and 60.9%, respectively. Again, SVM seems to be the best choice. One reason for this big difference is that the distribution of the test set is similar to the distribution of training set over the authors. So, an author with many training texts will also have many test texts. SVM can take advantage of this fact but CNG cannot. A total of 108 files were moved from the test set to the training set. For this file the accuracy in the predictions of both classifiers was 88%. For the remaining files of the test set the accuracy of CNG and SVM was 32.7% and 52.1%, respectively.

For the participation of the presented method to the PAN-11 author identification competition the provided training and test corpora were formed the labeled examples while the competition corpus was formed the unlabeled examples. The performance results are shown in the Table 1. For PAN11-AA-Large our approach won the first place indicating that the proposed method is effective for large candidate author sets.

Table 1. The performance of our approach in the PAN-11 competition.

Corpus	MacroAvg Prec.	MacroAvg Recall	MacroAvg F1	MicroAvg accuracy
PAN11-AA-Small	0.476	0.374	0.38	0.638
PAN11-AA-Large	0.549	0.532	0.52	0.658

4 Conclusion

Most of the studies in author identification consider the training corpus as fixed and stable. In this paper we presented a semi-supervised learning approach to author identification that attempts to enrich the training corpus with unlabeled examples taken from the test corpus. Two well-known author identification models work together and their predictions are used to transfer texts from the test corpus to the training corpus. This method can be used when there are multiple texts of unknown authorship or when a single long text can be segmented into multiple parts.

Preliminary results show that the proposed method is effective especially when the candidate author set is large. It can be extended to also handle open-set classification tasks by taking the degree of certainty of the two classifiers into account. This will also allow us to apply a semi-supervised learning procedure that will have more similarities with the original co-training algorithm since the most confident predictions of both classifiers will be transferred to the training set.

References

1. Blum, A., Mitchell, T.: Combining Labeled and Unlabeled Data with Co-training. In Proceedings of the Workshop on Computational Learning Theory, pp. 92--100, Morgan Kaufmann (1998).
2. Guzmán-Cabrera, R., Montes-Y-Gómez, M., Rosso, P., Villaseñor-Pineda, L.: Using the Web as Corpus for Self-training Text Categorization. Information Retrieval, 12(3), 400--415 (2009).
3. Houvardas, J., Stamatatos, E.: N-gram Feature Selection for Authorship Identification. In Proceedings of the 12th International Conference on Artificial Intelligence: Methodology, Systems, Applications, pp. 77--86, Springer (2006).

4. Keselj, V., Peng, F., Cercone, N., Thomas, C.: N-gram-based Author Profiles for Authorship Attribution. In Proceedings of the Pacific Association for Computational Linguistics, pp. 255--264 (2003).
5. Koppel, M., Akiva, N., Dagan, I.: Feature Instability as a Criterion for Selecting Potential Style Markers. Journal of the American Society for Information Science and Technology, 57(11), 1519--1525 (2006).
6. Koppel, M., Schler, J., Argamon, S.: Authorship Attribution in the Wild. Language Resources and Evaluation, 45(1), 83--94 (2011).
7. Levina, E., Bickel, P.J.: Maximum Likelihood Estimation of Intrinsic Dimension. In Advances in Neural Information Processing Systems 17, 777--784 (2005).
8. Luyckx, K.: Scalability Issues in Authorship Attribution. Ph.D. Thesis, University of Antwerp (2010).
9. Stamatatos, E.: Author Identification Using Imbalanced and Limited Training Texts. In Proceedings of the 4th International Workshop on Text-based Information Retrieval, pp. 237--241 (2007).
10. Stamatatos, E.: Author Identification: Using Text Sampling to Handle the Class Imbalance Problem. Information Processing and Management, 44(2), pp. 790--799 (2008).
11. Stamatatos, E.: A Survey of Modern Authorship Attribution Methods. Journal of the American Society for Information Science and Technology, 60(3), pp. 538—556 (2009).