Authorship Verification Using the Impostors Method
Notebook for PAN at CLEF 2013

Shachar Seidman
Bar Ilan University
shachar9@gmail.com

Abstract. This paper describes the evaluation of the *GenIM* method, which participated in the PAN' 13 authorship identification competition. The approach is based on comparing the similarity between the given documents and a number of external (impostor) documents, so that documents can be classified as having been written by the same author, if they are shown to be more similar to each other than to the impostors, in a number of trials. The method showed competitive results, achieving the overall 1st ranking in the competition.

1 Introduction

The task we wish to solve is that of determining whether a given document is an outlier in a set of documents known to have been written by a single author. Like all other methods that have been suggested for this task, we define some distance measure of the given document from the other documents in the collection. The novelty of our method is that our distance measure is not based directly on the actual similarity between two texts (as is customary), but rather on a second-order measure defined in Koppel and Winter (2013). The measure defined there was designed to solve the problem of determining whether two documents, say X and Y, are by the same author. The method proposed there, known as the "Impostors Method" (*IM*) is to check whether X is more similar to Y than to each one of a set of impostors $<I_1,\ldots,I_n>$, where the comparison is made for each of 100 different feature sets. The "similarity" of X to Y is the percentage of feature sets for which X is more similar to Y than to any of the impostor sets.

For the PAN' 13 competition, we adapted the *IM* method to support different languages. Our assumption was that the *IM* method itself is language independent, but its parameters should be optimized for each language separately. In addition, we generalize the *IM* to support the comparison of a document to a number of documents in the most effective way.

2 Our Method: *General Impostors Method*

2.1 Original *Impostors Method* (*IM*) Implementation

Koppel and Winter suggested and evaluated a few variations of the *IM*. In our work we considered the following approach, which was based on the *Many-Candidate* method suggested by Koppel et al. (2011):

```
Impostors Method
Input: <X,Y>: A pair of documents. S: A set of impostors.
Output: <same-author> or <diff-author>
   1. Set Score=0
   2. Repeat k times
         a. Randomly choose rate% of the features from the full
            feature pool.
         b. Randomly choose n impostors from S: I_1,…,I_n.
         c. Score = Score + 1/k if Sim(X,Y) * Sim(Y,X) > Sim(X,
            I_i) * Sim(Y, I_i), for each i∈{1,…,n}.
   3. Return <same-author> if Score > Δ*; else <diff-author>.
```

The given pair <X, Y> and impostor set S are represented as frequency feature vectors. We experimented with various feature sets for each language, including function words, unigrams, n-grams and n-character-grams. We evaluated the features with different frequency representations, including binary, numeric and tf-idf. Different distance/similarity measures were tested, including Euclidean, Manhattan and Min-Max distance. The number of iterations k as well as parameters *rate%* and Δ^* were optimized. All parameters were evaluated per language.

Koppel and Winter (2013) suggest a few approaches for selecting or generating the impostor set *S*. We generated such impostor sets from the web, using a search engine. We downloaded a single web impostor corpus for each language and it was used as the impostor set *S* for all problems. We used the following information retrieval technique, suggested by Koppel and Winter (2013), to generate the impostors: We chose a few (3-4) seed documents and randomly chose small sets (3-5 words) of words from them (excluding function words). For each such word set, we ran a web search query and added the top 10 returned documents to the web impostor corpus, repeating the process until we had a large enough corpus. The returned documents were stripped of html and such. Only the first 1,500 words for each web impostor were considered.

2.2 *General Impostors Method* (*GenIM*) Implementation

The *IM* method was designed to deal with a pair of documents, but our problem was measuring whether one document is an outlier in a set of documents, so we had to adapt *IM* to support it. We considered several approaches, and the highest performing and most robust approach was running *IM* on all pairs consisting of the questioned and a single known document and aggregating the results. Formally, the process is as follows:

```
General Impostors Method
Input: X: The unknown document.  Y = {Y1,…,Yn}: Known documents.
Output: <same-author> or <diff-author>.
   1. For each pair of documents <X, Yi> in set D:
         a. Run original IM on the pair to obtain a similarity
            binary score S(X, Yi).
   2. Score = Avg over similarity scores ([S(X, Y1)… S(X, Yn)]).
   3. Return <same-author> if Score > θ*; else <diff-author>.
```

3 Evaluation

We used the training set in this year's competition as an evaluation set for both *IM* and *GenIM*. Since our method consists of these two phases, we had to measure performance and optimize parameters at each step.

3.1 IM Parameters Optimizations

25%-33% of the training documents of each language were used to measure and optimize *IM*, while the others were used to evaluate *GenIM*. For the *IM* evaluation set we used 3-4 documents as seed documents for the web impostor retrieval. The best parameter values for *IM* are presented below in Table 1, while the optimum threshold Δ* is shown in Table 2 along with its performance:

	English	Greek	Spanish
Features	Unigrams-Tfidf	Unigrams	Character 4-Grams
Similarity Function	Directional-MinMax	MinMax	MinMax
#Iterations	25	25	50
Imps Corpus Size	1095	1294	1289
#Imps per problem	250	250	250
#Imps per iteration	100	50	50
%Features per it	40%	60%	40%

Table 1. *IM* Optimized Parameters per Language

	English	Greek	Spanish
Best threshold Δ*	0.7	0.84	0.35
<*diff*> F1 score	96.15%	87.27%	95.85%
<*same*> F1 score	95.83%	84.44%	96.15%

Table 2. *IM* Optimized Threshold and Performance

We used most of the training data to choose the aggregation function and to optimize θ* for *GenIM*. Results are shown in Table 3.

	English	Greek	Spanish
Best threshold θ*	0.75	0.75	0.75
<*diff*> F1 score	100%	82.82%	100%
<*same*> F1 score	100%	77.78%	100%

Table 3. *GenIM* Optimized Threshold and Performance

3.2 Results

The results for the 2013 competition data are summarized in Table 4.

	English	Greek	Spanish	Total
Accuracy - Training	90%	75%	100%	**82.86%**
Accuracy - Test	80%	83.33%	60%	**75.3%**

Table 4. Results for the 2013 Competition Data

4 Conclusions

The test results are fairly consistent with the training results, apart from Spanish, where we suspect that the parameters were not optimized correctly, as a result of lack of training examples.

We showed that the *GenIM* method is a competitive method for the problem of authorship verification and that it is language independent (aside from the feature selection phase). The method can be improved by better feature selection and impostor generation.

References

1. Keselj, V., Peng, F., Cercone, N., and Thomas, C. N-gram-based author profiles for authorship attribution. *Proceedings of PACLING (pp. 255-264)*. 2003.
2. Kotlerman L., Dagan I., Szpektor I., and Zhitomirsky-Geffet M. Directional distributional similarity for lexical expansion*. Proceedings of the ACL-IJCNLP.* 2009.
3. Koppel, M., Schler J., and Argamon S. Computational Methods in Authorship Attribution. *JASIST, 60(1): 9-26*. 2009.
4. Koppel M., Schler J., and Argamon S. Authorship attribution in the wild. *Language Resources and Evaluation, 45(1): 83–94*. 2011.
5. Koppel M. and Winter. Y. Determining If Two Documents Are by the Same Author, *JASIST* , in press
6. Stamatatos, E. A Survey of Modern Authorship Attribution Methods. Intrinsic Plagiarism Detection Using Character n-grams, *JASIST, 60(3): 538-556*. 2009.