# UniNE at CLEF 2015: Author Identification

## Notebook for PAN at CLEF 2015

**Mirco Kocher, Jacques Savoy**
University of Neuchâtel
rue Emile Argand 11
2000 Neuchâtel, Switzerland
{Mirco.Kocher, Jacques.Savoy}@unine.ch

**Abstract.** This paper describes and evaluates an unsupervised authorship verification model called SPATIUM-L1. The suggested strategy can be adapted without any problem to different languages (such as Dutch, English, Greek, and Spanish) with their genre and topic differ significantly. As features, we suggest using the *k* most frequent terms of the disputed text (isolated words and punctuation symbols with *k* may vary from 200 to 300). Applying a simple distance measure and a set of impostors, we determine whether or not the disputed text was written by the proposed author. Moreover, based on a simple rule, we can define when there is enough evidence to propose an answer with a high degree of confidence or when the attribution scheme is given without certainty. The evaluations are based on four test collections (PAN AUTHOR IDENTIFICATION task at CLEF 2015).

## 1 Introduction

Automatic authorship identification aims to determine, as accurately as possible, if the proposed author of a document or a text excerpt is the real one [9]. To achieve this, a sample of texts written by the proposed author and each of the possible impostors must be available. The verification problem knows some interesting historical questions such as "are all the *Letters of Paul* written by the same person?", "is President L. Johnson the real author of the 1964 *State of the Union Address* (just weeks after the assassination of Kennedy)?", or "is Madison the true author of the 12 disputed *Federalist Papers*?". With the Web 2.0 technologies, the number of anonymous or pseudonymous texts is increasing and in many cases we face a single possible author (*e.g.*, is John the real author of this blog post or tweet?). Therefore, proposing an effective algorithm to the verification problem presents a real interest. A justification supporting the proposed answer and a probability that the given answer is correct can be given to improve the confidence attached to the response [6].

This authorship verification question seems simpler than the classical authorship attribution problem, but it is not. For example, if we want to know if a newly discovered poem was really written by Shakespeare [11], the computer needs to compare a model based on Shakespeare's texts with all other possible representative non-Shakespeare models. This second part is hard to generate. Are we sure we have included all other writers having a style similar to Shakespeare?

This paper is organized as follows. The next section presents the test collections and the evaluation methodology used in the experiments. The third section explains our proposed algorithm called SPATIUM-L1. In the last section, we evaluate the proposed scheme and compare it to the best performing schemes using four different test collections. A conclusion draws the main findings of this study.

## 2 Test Collections and Evaluation Methodology

The experiments supporting previous studies were usually limited to one language, one author, and one or a few texts. For real cases, this limitation makes sense; for example we have only one newly discovered poem that might be attributed to Shakespeare [11]. To evaluate the effectiveness of a verification algorithm, the number of tests must however be larger. To create such benchmarks, and to promote studies in this domain, the PAN CLEF evaluation campaign was launched [10]. The evaluation was performed using the *TIRA* platform, which is an automated tool for deployment and evaluation of the software [2]. The data access is restricted such that during a software run the system is encapsulated and thus ensuring that there is no data leakage back to the task participants [5]. This evaluation procedure may raises some difficulties (possible system compatibilities) but offers also a fair evaluation of the time needed to produce an answer.

During the PAN CLEF 2015 evaluation campaign, four test collections were built, each containing at least 200 problems (training + testing). In each collection, all the texts matched the same language but can be cross-topic or cross-genre and may differ significantly. In this context, a problem is defined as:

> *Given a small set of "known" documents (no more than seven, possibly as few as one) written by a single person, is the new "unknown" document also written by that author?*

The four benchmarks are composed of a Dutch and Spanish cross-genre collection and an English and Greek cross-topic corpus. An overview of these collections is depicted in Table 1. The training set will be used to evaluate our approach and the test set will be used in order to be able to compare our results with those of the PAN CLEF 2015 campaign.

| | | Training | | | Test |
|---|---|---|---|---|---|
| Language | Type | No of Problems | Mean document | Mean words | No of Problems |
| Dutch | cross-genre | 100 | 1.8 | 449 | 165 |
| English | cross-topic | 100 | 1.0 | 341 | 500 |
| Greek | cross-topic | 100 | 2.9 | 688 | 100 |
| Spanish | cross-genre | 100 | 4.0 | 976 | 100 |

**Table 1.** PAN CLEF 2015 corpora statistics

The number of problems is given under the label "No of Problems". The mean number of known documents for each problem is indicated in the column "Mean document", and the mean number of words per known document under the label "Mean words". The mean number of words in the unknown documents is close to the latter.

The last two metrics are not available for the test corpora because the datasets remained undisclosed thanks to the *TIRA* system.

When inspecting the English training collection, the number of words available is rather small (in mean 341 words for each document, and exactly one document per problem). Similarly the Dutch collection only provides 808 words in mean per problem (in mean 449 words for each document, and 1.8 documents in each problem). This collection has mostly one or two document per problem but also some with 5, 6 or even 7 known documents. Therefore, we can expect the mean performance for these languages to be lower than for the other languages under the assumption that all languages present the same level of complexity to solve this problem. For the Spanish corpus we have always four documents and rather long ones to learn the stylistic features of the proposed author. A relatively higher performance can be assumed with this benchmark. A similar conclusion can be expected with the Greek collection consisting of longer documents (in mean, 1,995 words).

When considering the four benchmarks as a whole, we have 865 problems to solve and 400 to train (pre-evaluate) our system. When inspecting the distribution of the correct answers, we can find the same number (432 in test and 200 in training) as positive or negative answers. In each of the individual test collections, we can also find a balanced number of positive and negative answers.

During the PAN CLEF 2015 campaign, a system must return a value between 0.0 and 1.0 for each problem with a precision down to a thousandth. A value strictly larger than 0.5 indicates that the query text was written by the proposed author and a value strictly lower than 0.5 the opposite. Returning the value 0.5 indicates that the system is unable to take a decision based on the given information. Of course, a value closer to 1.0 (or to 0.0) is a stronger evidence in favor of (or against) a positive answer.

As performance measure, two evaluation measures were used during the PAN CLEF campaign. The first performance measure is the AUC (Area Under the Curve) of the ROC (Receiver Operating Characteristic) curve [12]. This curve is generated according to the percentage of false positives (or false alarms) in the x-axis and the percentage of true positives in the y-axis over the entire test set. The maximum value of 1.0 indicates a perfect performance. Both the ROC and the AUC measures are, however, rather complex and difficult to interpret by a final user.

As another measure, the PAN CLEF campaign adopts the c@1 measure [4]. This evaluation measure takes into account both the number of correct answers and the number of problems left unsolved in the test set. The exact formulation is given in Equation 1 with a minimal value of 0.0 and 1.0 as an optimum value.

$$c@1 = \frac{nc}{np} \cdot \left(1 + \frac{nu}{np}\right) \tag{1}$$

in which *np* is the number of problems, *nc* the number of correct answers, and *nu* the number of problems left without an answer. This measure differentiates between an incorrect answer and the absence of an answer (indicating that the provided evidence is not enough to take a definitive decision) [10].

## 3 Simple Verification Algorithm

To solve the verification problem, we suggest an unsupervised approach based on a simple feature extraction and distance metric called SPATIUM-L1 (Latin word meaning distance). The selected stylistic features correspond to the top $k$ most frequent terms (isolated words without stemming but with the punctuation symbols). For determining the value of $k$, previous studies have shown that a value between 200 and 300 tends to provide the best performance [1, 7]. Some unknown documents were rather short and we further excluded the words only appearing once in the text. This filtering decision was taken to prevent overfitting to single occurrences. The effective number of terms $k$ was set to at most 200 terms but was in most cases well below. With this reduced number the justification of the decision will be simpler to understand because it will be based on words instead of letters, bigrams of letters or combinations of several representation schemes or distance measures.

In the current study, a verification problem is defined as a query text, denoted Q, and a set of texts (between 1 and 7) written by the same proposed author. The concatenation of these texts forms the author profile A. To measure the distance between Q and A, SPATIUM-L1 uses the L1-norm as follows:

$$\Delta(Q, A) = \Delta_0 = \sum_{i=1}^{k} \left| P_Q[t_i] - P_A[t_i] \right| \tag{2}$$

where $k$ indicates the number of terms (words or punctuation symbols), and $P_Q[t_i]$ and $P_A[t_i]$ represent the estimated occurrence probability of the term $t_i$ in the query text Q and in the author profile A respectively. To estimate these probabilities, we divide the term occurrence frequency ($tf_i$) by the length in tokens of the corresponding text ($n$), $\text{Prob}[t_i] = tf_i / n$, without smoothing and therefore accepting a 0.0 probability.

To verify whether the resulting $\Delta_0$ value is small or rather large, we need to select a set of impostors. To achieve this, three profiles from other problems in the test set were chosen randomly with preference to candidates that show the same number of known documents. This value of three is arbitrary and will be denoted by the variable $m$. After computing the distance between Q and each of these $m$ profiles, we retain only the smallest distance.

Instead of limiting the number of possible impostors to $m$, we iterate this last stage $r$ times, and we suggest to fix the value $r = 5$. After this last step, we have $r$ values denoted $\Delta_{m1}$, …, $\Delta_{mr}$, each of them corresponding to the minimum value of a set of $m$ impostors. Instead of working with $r$ values, we compute the arithmetic mean, denoted $\Delta_m$, of the sample $\Delta_{m1}$, …, $\Delta_{mr}$.

Finally, the decision rule is based on the value of the ratio $\Delta_0$ / $\Delta_m$ as follows:

$$\begin{cases} if\ \frac{\Delta_0}{\Delta_m} < 0.975 & same\ author \\ if\ \frac{\Delta_0}{\Delta_m} > 1.025 & different\ author \\ otherwise & don't\ know \end{cases} \tag{3}$$

Thus when the $\Delta_0$ value is similar to $\Delta_m$ (in the range ±2.5%), the system specifies that the solution of this problem cannot be determined with good certainty and provides the answer *don't know*. On the other hand, when $\Delta_0$ is small compared to $\Delta_m$, the evidence is in favor of assuming that the author of the profile A is the real author.

Finally, when $\Delta_m$ is small compared to $\Delta_0$, we conclude that Q and A are written by different authors. The limit of two times 2.5% was chosen arbitrarily but corresponds to a well-known limit value in statistical tests.

## 4 Evaluation

Since our system is based on an unsupervised approach we were able to directly evaluate it using the training set. In Table 2, we have reported the same performance measure applied during the PAN CLEF campaign, namely the final score, which is the product of the AUC and the c@1.

| Language | **Final** | AUC | c@1 | Runtime (h:m:s) |
|---|---|---|---|---|
| Dutch | **0.2161** | 0.4738 | 0.4560 | 00:00:08 |
| English | **0.3450** | 0.6032 | 0.5720 | 00:00:07 |
| Greek | **0.5415** | 0.7648 | 0.7080 | 00:00:12 |
| Spanish | **0.5694** | 0.8320 | 0.6844 | 00:00:12 |

**Table 2.** Evaluation for the four *training* collections

The algorithm returns the best results for the Spanish collection with a final score of 0.5694 closely followed by the Greek corpus possibly due to the fact of the longer and numerous documents in these two languages. The worst result is achieved with the Dutch collection that shows a highly diverging number of known documents per problem. Usually the AUC values should be consistent and comparable with the c@1 values but in some cases the AUC values are a lot higher than the c@1 values (mainly in the Spanish collection but also observable in the English and Greek corpus). As possible reason we saw a few misclassifications that have very high probability scores. The AUC of the ROC is biased in a way that the ROC gives more emphasis on the first position and therefore increases the total AUC. A misclassification with a lower probability is less penalized in this performance measure.

Due to the fact that our algorithm is based on a probabilistic approach (random selection of candidates) the results in Table 2 may vary between runs. To verify the impact of this selection in the reported performance measures, Table 3 shows the standard deviation and the estimated confidence interval covering 95% of the cases for the c@1 value based on 200 restarts with random impostor selection.

| | c@1 | | |
|---|---|---|---|
| Language | Mean | Standard deviation | Interval (95%) |
| Dutch | 0.4769 | 0.0210 | [0.4356 – 0.5181] |
| English | 0.5776 | 0.0237 | [0.5312 – 0.6241] |
| Greek | 0.6932 | 0.0258 | [0.6426 – 0.7437] |
| Spanish | 0.6997 | 0.0308 | [0.6393 – 0.7601] |

**Table 3.** Variation around the c@1 performance for the SPATIUM-L1 system

We can see the possible variation around the reported performance is noticeable but relatively small. Similarly when changing the values of the two numbers *m* (number of impostors) and *r* (number of iterations) then the difference of the best possible combination of the two parameters to the performance reported in Table 2 is not significant.

The test set is then used to rank the performance of all 18 participants in this task. Based on the same evaluation methodology, we achieve the results depicted in Table 4 corresponding to the 865 problems present in the four test corpora.

As we can see the final score with the Greek corpus is as high as expected from the training set. The results we achieved in the Dutch collection is as low as in the training set. On the other hand the English results are better than anticipated and the Spanish score is worse than the estimation from the training set. It seems like the system performs better on the two cross-topic corpora (English and Greek) than on the two cross-genre corpora (Dutch and Spanish).

| Language | **Final** | AUC | c@1 | Runtime (h:m:s) | Position |
|---|---|---|---|---|---|
| Dutch | **0.2175** | 0.4495 | 0.4840 | 00:00:07 | 14 |
| English | **0.5082** | 0.7375 | 0.6890 | 00:00:24 | 4 |
| Greek | **0.6310** | 0.8216 | 0.7680 | 00:00:11 | 3 |
| Spanish | **0.3665** | 0.6498 | 0.5640 | 00:00:22 | 10 |

**Table 4.** Evaluation for the four *testing* collections

To put those values in perspective we can see in Table 5 our result in comparison with the other 17 participants using macro-averaging. We have also added a baseline corresponding to a system that always produces the answer yes (trivial acceptor). The bad performance in the Dutch collection clearly worsens our overall results.

| Rank | Run | **Final** | AUC | c@1 | Runtime (h:m:s) |
|---|---|---|---|---|---|
| 1 | bagnall15 | **0.6340** | 0.8199 | 0.7663 | 55:14:16 |
| 2 | moreau15 | **0.6103** | 0.8186 | 0.7409 | 55:24:10 |
| 3 | pacheco15 | **0.5606** | 0.8164 | 0.6833 | 00:26:31 |
| 4 | nissim15 | **0.5416** | 0.7457 | 0.7221 | 00:04:53 |
| 5 | bartoli15 | **0.5182** | 0.7398 | 0.6837 | 00:44:35 |
| 6 | mezaruiz15 | **0.4829** | 0.7218 | 0.6621 | 02:10:28 |
| 7 | halvani15 | **0.4618** | 0.7354 | 0.6282 | 00:01:01 |
| 8 | kocher15 | **0.4308** | 0.6646 | 0.6263 | 00:01:04 |
| … | … | **…** | … | … | … |
| 13 | Baseline (yes) | **0.2500** | 0.5000 | 0.5000 | 00:00:00 |
| … | … | **…** | … | … | … |

**Table 5.** Evaluation over all four test collections using macro-averaging for the effectiveness measures and the sum for the runtimes.

Another pertinent observation is the fast runtime of our system in comparison with other solutions[1]. The median execution time of the other systems is almost one hour. Also the runtime only shows the actual time spent to classify the test set. On *TIRA* there was the possibility to first train the system using the training set which had no influence on the final runtime. Since we have an unsupervised system it did not need to train any parameters, but this possibility might have been used by other participants.

In text categorization studies, we are convinced that a deeper analysis of the evaluation results is important to obtain a better understanding of the advantages and drawbacks of a suggested scheme. By just focusing on overall performance measures, we only observe a general behavior or trend without being able to acquire a better

[1] http://www.tira.io/task/authorship-verification/

explanation of the proposed assignment. To achieve this deeper understanding, we could analyze some problems extracted from the English corpus. Usually, the relative frequency (or probability) differences with very frequent words such as *when*, *is*, *in*, *that*, *to*, or *it* can explain the decision.

## 5 Conclusion

This paper proposes a simple unsupervised technique to solve the authorship verification problem. As features to discriminate between the proposed author and different impostors, we propose using the top 200 most frequent terms (words and punctuations). This choice was found effective for other related tasks such as authorship attribution [1]. Moreover, compared to various feature selection strategies used in text categorization [8], the most frequent terms tend to select the most discriminative features when applied to stylistic studies [7]. In order to take the attribution decision, we propose using a simple distance metric called SPATIUM-L1 based on the L1 norm.

The proposed approach tends to perform very well in two different languages (English and Greek) on cross-topic collections and well in a Spanish cross-genre corpus. Such a classifier strategy can be described as having a high bias but a low variance [3]. Even if the proposed system cannot capture all possible stylistic features (bias), changing the available data does not modify significantly the overall performance (variance).

It is common to fix some parameters (such as time period, size, genre, or length of the data) to minimize the possible source of variation in the corpus. However, our goal was to present a simple and unsupervised approach without many predefined arguments. This turned out to not work well on the Dutch cross-genre corpus. We suspect this to be mostly related to the genre variation than the language itself.

SPATIUM-L1 returns a numerical value (between 0 and 1) that can be used to determine a degree of certainty [6]. More importantly, the proposed attribution could be clearly explained because it is based on a reduced set of features on the one hand and, on the other, those features are words or punctuation symbols. Thus the interpretation for the final user is clearer than when working with a huge number of features, when dealing with *n*-grams of letters or when combing several similarity measures. The SPATIUM-L1 decision can be explained by large differences in relative frequencies of frequent words, usually corresponding to functional terms.

To improve the current classifier, we will investigate the consequence of some smoothing techniques, the effect of other distance measures, and different feature selection strategies. In the latter case, we want to maintain a reduced number of terms. In a better feature selection scheme, we can take account of the underlying text genre, as for example, the most frequent use of personal pronouns in narrative texts. As another possible improvement, we can ignore specific topical terms or character names appearing frequently in an author profile, terms that can be selected in the feature set without being useful in discriminating between authors. We might also try to exploit PAN specific properties such as the requirement for equally distributed positive and negative problems. In case our system decides in over half the cases for (or against) a verification we could assign for the least certain part that it is a *don't know* answer.

## Acknowledgments

The author wants to thank the task coordinators for their valuable effort to promote test collections in authorship attribution. This research was supported, in part, by the NSF under Grant #200021_149665/1.

## 6 References

1. Burrows, J.F. 2002. Delta: A Measure of Stylistic Difference and a Guide to Likely Authorship. *Literary and Linguistic Computing*, 17(3), 267-287.
2. Gollub, T., Stein, B., & Burrows, T. 2012. Ousting Ivory Tower Research: Towards a Web Framework for Providing Experiments as a Service. In: Hersh, B., Callan, J., Maarek, Y., & Sanderson, M. (eds.) SIGIR. *The 35th International ACM*, 1125–1126.
3. Hastie, T., Tibshirani, R., & Friedman, J. 2009. *The Elements of Statistical Learning*. *Data Mining, Inference, and Prediction*. Springer-Verlag: New York (NY).
4. Peñas, A., & Rodrigo, A. 2011. A Single Measure to Assess Nonresponse. In *Proceedings 49th ACL*, 1415-1424.
5. Potthast, M., Gollub, T., Rangel, F., Rosso, P., Stamatatos, E., & Stein, B. 2014. Improving the Reproducibility of PAN's Shared Tasks: - Plagiarism Detection, Author Identification, and Author Profiling. In: Kanoulas, E., Lupu, M., Clough, P., Sanderson, M., Hall, M., Handbury, A., & Toms, E. (eds.) *CLEF. Lecture Notes in Computer Science*, vol. 8685, 268–299. Springer.
6. Savoy, J. 2015a. Estimating the Probability of an Authorship Attribution. *Journal of American Society for Information Science & Technology*, to appear.
7. Savoy, J. 2015b. Comparative Evaluation of Term Selection Functions for Authorship Attribution. *Digital Scholarship in the Humanities*, to appear (dx.doi.org/10.1093/llc/fqt047).
8. Sebastiani, F. 2002. Machine Learning in Automatic Text Categorization. *ACM Computing Survey*, 34(1), 1-27.
9. Stamatatos, E. 2009. A Survey of Modern Authorship Attribution Methods. *Journal of the American Society for Information Science and Technology*, 60(3), 433-214.
10. Stamatatos, E., Daelemans, W., Verhoeven, B., Juola, P., Lopez Lopez, A., Potthast, M., & Stein, B. 2015. Overview of the Author Identification Task at PAN 2015. In Working Notes Papers of the CLEF 2015 Evaluation Labs, *CEUR Workshop Proceedings*, CEUR-WS.org.
11. Thisted, R., & Efron, B. 1987. Did Shakespeare Write a Newly-Discovered Poem? Biometrika, 74(3), 445-456.
12. Witten, I.H., Frank, E., & Hall, M.A. 2011. Data Mining. *Practical Machine Learning Tools and Techniques*. 3rd ed., Morgan Kaufmann: Burlington (MA).