Université de Neuchâtel
Faculté des Sciences
Institut d'Informatique

Text Clustering with Styles

par

Mirco Kocher

Thèse

présentée à la Faculté des Sciences
pour l'obtention du grade de Docteur ès Sciences

Acceptée sur proposition du jury:

Prof. Jacques Savoy, Directeur de thèse
Université de Neuchâtel, Suisse

Prof. Fabio Crestani, rapporteur
Università della Svizzera italiana, Suisse
Prof. Paolo Rosso, rapporteur
Universitat Politècnica de València, Espagne
Dr. Valerio Schiavoni, rapporteur
Université de Neuchâtel, Suisse

Soutenue le 27 novembre 2017

UNIVERSITÉ DE NEUCHÂTEL

Faculté des Sciences
Secrétariat-décanat de Faculté
Rue Emile-Argand 11
2000 Neuchâtel – Suisse
Tél : + 41 (0)32 718 21 00
E-mail : secretariat.sciences@unine.ch

IMPRIMATUR POUR THESE DE DOCTORAT

La Faculté des sciences de l'Université de Neuchâtel autorise l'impression de la présente thèse soutenue par

Monsieur Mirco KOCHER

Titre:

“Text Clustering with Styles”

sur le rapport des membres du jury composé comme suit:

- Prof. Jacques Savoy, directeur de thèse, Université de Neuchâtel, Suisse
- Dr Valerio Schiavoni, Université de Neuchâtel, Suisse
- Prof. Fabio Crestani, Università della Svizzera italiana, Lugano, Suisse
- Prof. Paolo Rosso, Universidad Politecnica de Valencia, Espagne

Neuchâtel, le 22 décembre 2017

Le Doyen, Prof. R. Bshary

ABSTRACT

This thesis mainly describes the author clustering problem where, based on a set of n texts, the goal is to determine the number k of distinct authors and regroup the texts into k classes according to their author. We iteratively build a stable and simple model for text clustering with styles.

We start by designing a measure reflecting the (un)certainty of the proposed decision such that every decision comes along with a confidence of correctness instead of only giving a single answer. Afterwards, we link those pairs of texts where we see an indication of a shared authorship and have enough evidence that the same person has written them. Finally, after checking every text tuple, if we can link them together, we build the final clusters based on a strategy using a distance of probability distributions. Employing a dynamic threshold, we can choose the smallest relative distance values to detect a common origin of the texts.

While in our study we mostly focus on the creation of simple methods, investigating more complex schemes leads to interesting findings. We evaluate distributed language representations and compare them to several state-of-the-art methods for authorship attribution. This comparison allows us to demonstrate that not every approach excels in every situation and that the deep learning methods might be sensitive to parameter settings.

The most similar observations (or the category with the smallest distance) to the sample in question usually determines the proposed answers. We test multiple inter-textual distance functions in theoretical and empirical tests and show that the Tanimoto and Matusita distances respect all theoretical properties. Both of them perform well in empirical tests, but the Canberra and Clark measures are even better suited even though they do not fulfill all the requirements. Overall, we can note that the popular Cosine function neither satisfies all the conditions nor works notably well. Furthermore, we see that reducing the text representation not only decreases the runtime but can also increase the performance by ignoring spurious features. Our model can choose the characteristics that are the most relevant to the text in question and can analyze the author adequately.

We apply our systems in various natural languages belonging to a variety of language families and in multiple text genres. With the flexible feature selection, our systems achieve reliable results in any of the tested settings.

Keywords: Natural language processing, inter-textual distance, text representation, digital humanities, digital libraries, text clustering, authorship attribution, unsupervised learning

RÉSUMÉ

Cette thèse présente le problème du regroupement d'auteurs formulé de la manière suivante : en partant d'un ensemble composé de n textes, le but est de déterminer le nombre k d'auteurs distincts, pour regrouper les textes en k classes. De manière itérative, nous construisons un système stable et simple qui est capable de regrouper automatiquement les documents selon leurs thèmes.

Dans notre étude, nous commençons par proposer une mesure capable d'estimer l'(in-)certitude de la décision proposée, dans le but d'obtenir un indicateur de confiance en lieu et place d'une simple réponse. Ensuite, nous combinons les paires de textes pour lesquelles une même affectation apparaît, et dont nous sommes suffisamment confiants pour affirmer qu'ils sont rédigés par le même auteur. Enfin, après avoir vérifié chaque tuple de textes, nous construisons les classes en nous basant sur une stratégie utilisant une distance entre distributions probabilistes. Grâce à l'utilisation d'une limite dynamique, nous sommes à même de choisir les plus petites distances relatives pour détecter une origine commune entre textes.

Bien que notre étude se concentre principalement sur la création de méthodes simples, des schémas plus complexes mènent à des résultats plus performants. Ainsi, nous avons opté pour une représentation distribuée et nous avons comparé son efficacité à plusieurs méthodes d'attribution d'auteurs. Cette évaluation nous permet de démontrer que toutes les approches n'excellent pas dans toutes les situations, et que des méthodes d'apprentissage profond peuvent être sensibles au choix des paramètres.

Les observations les plus proches des exemples en question (ou la catégorie ayant la plus petite distance) déterminent généralement les réponses proposées. Nous avons testé plusieurs fonctions de distance inter-textuelle sur des critères théoriques et empiriques. Nous démontrons que les distances dites de Tanimoto et de Matusita respectent toutes les propriétés théoriques. Toutes deux obtiennent également de bons résultats dans le cadre de tests empiriques. Toutefois, les mesures de Canberra et de Clark sont encore mieux adaptées, bien qu'elles ne remplissent pas toutes les caractéristiques théoriques demandées. De manière générale, l'on constate que la fonction Cosinus ne répond pas à toutes les conditions, et se comporte de façon suboptimale. Enfin, nous observons que la réduction des traits stylistiques retenues diminue le temps d'exécution et peut également améliorer les performances en ignorant les redondantes.

Nous testons nos systèmes pour différentes langues naturelles appartenant à une variété de familles de langues et pour plusieurs genres de textes. Grâce à la sélection flexible des attributs, nos systèmes sont capables de produire des résultats fiables dans toutes les conditions testées.

Mots-clés : Traitement automatique du langage naturel, distance inter-textuelle, linguistique computationnelle, attribution d'auteur, classification automatique, apprentissage non-supervisé

ACKNOWLEDGMENTS

I would like to express my gratitude first and foremost to my supervisor Prof. Jacques Savoy for his support, guidance, and encouragement to always strive for improvement during my Ph.D. thesis.

Likewise, I would like to thank the members of the jury, namely Prof. Fabio Crestani (Università della Svizzera italiana), Prof. Paolo Rosso (Universitat Politècnica de València), and Dr. Valerio Schiavoni (Université de Neuchâtel), for the time they devoted to evaluating this dissertation.

I am thankful for the many interesting discussions at the IIUN and all my colleagues at UniNE throughout these years have my thanks for maintaining a great atmosphere, both at the office and in the many activities organized that are hardly related to Computer Science. Furthermore, I am much obliged to my friends for their kind thoughts, entertainment, and all the great time we spend together. Finally, I want to address the biggest thanks to my family for their patience and endless support; I couldn't have done it alone.

This research was supported in part by the Swiss National Science Foundation under Grand #200021-149665/1.

Contents

Chapter 1

Introduction

The Internet offers a vast amount of information in written format for which we should develop and improve effective text analyzing algorithms. This publicly available material poses important questions from a security perspective, especially regarding the increased number of pseudonymous posts, chats, threatening e-mails, or anonymously written documents available on the Web [37]. Authorship analysis involves the study of textual data to deduce knowledge about its creator based on the applied writing style. The result is a "fingerprint" for anything written by an author or a particular group of people making it easy to compare it to any other writing. Some interesting problems emerging from blogs and social networks are, for instance, detecting plagiarism, recognizing stolen identities, or rectifying wrong information about the writer. To be able to determine or characterize the real author of a document automatically is an obvious concern for criminal investigations. In historical or literary studies, being able to verify the gender of a given character may open new research directions, e.g., does Juliet speak like a female figure [11]? From a marketing viewpoint, companies may be interested in knowing, based on the analysis of blogs and online product reviews, the demographics of people that like or dislike their products (e.g., gender, personality traits). Similarly, online audience identification can be used by advertisers to target specific groups with unique characteristics, e.g., to show a product to people from a limited age range, with a certain level of education, with a given political view, or in a defined geographic location. Furthermore, detecting the real person behind a pseudonymous author is a general curiosity, e.g., trying to find out who Robert Galbraith is (we now know it is J. K. Rowling) or who Elena Ferrante is (still unknown at the time of writing this thesis). Therefore, proposing suitable algorithms to those problems presents an indisputable interest.

1.1 Motivation & Objectives

Given the growing interest in the Web and the importance of multilingual applications, we aim to promote authorship analyzing models that would also be capable of correctly identifying the author of texts available from various electronic document sources (wiki-based corpus, blogosphere, e-mail) and written in a variety of languages. Today, these issues are of prime importance in a world centered on electronic information, especially given its ever-increasing volume of illegal copies, fraud

threats, and terrorism. Within the natural language processing domain, designing, implementing, and evaluating models working with languages other than English is an important and promising research field. Having a better understanding of the impact that different linguistic constructions may have on automatic processing effectiveness is an important task.

Further motivation for choosing our direction of work was the prominence of relatively complicated approaches that require the tuning of several hyper-parameters and many parameters. In the end, the "optimal" parameter setting works for a specific case (combination of language and text genre) to solve some problems. Our observations were that various parameter settings could lead to very different and sometimes unpredictable results. Furthermore, deep learning and word embedding got quite popular lately showing promising achievements because the vector representations of a word based on its context can be employed in many natural language processing applications and is an active field. We speculated that it is possible to reach a high effectiveness without tuning (and possibly over-fitting) the complex systems, but instead to shift the focus to an overall simpler framework that works independently of the underlying textual data. We feel that the user should receive a logical reasoning for the decisions, which is not possible with a black box system (e.g., Support Vector Machine (SVM), neural networks, random forest) and even harder for combinations of multiple approaches.

Our goal is to design, implement, and evaluate authorship analyzing models for writings in various natural languages from a variety of language families (e.g., members of the Romance, Germanic, or Slavic families), along with selected others (e.g., Arabic and Greek). Furthermore, we want to provide and evaluate different text representation strategies and define selection procedures to allow the authorship analyzing system to be more efficient. The final goal is to develop and implement clustering strategies to discover certain information linked to the author. Overall, based on the writing style in a text, we want to be able to determine the real author of a given document, verify a shared authorship, determine the demographics of an individual author, or group up all documents by the same person.

1.2 Stylometry

To solve these stylometric questions, we can assume that every person (or each demographic class) owns a unique, measurable style that is distinct from that of others. The measured style should be the same for all documents written by a given author (e.g., Anna) or from any category (e.g., female or young writers). Simultaneously, the measured style should be different to another author (e.g., John) or another category (e.g., male or old writers). Therefore, we search for textual features that have an intra-class similarity, and at the same time, show an inter-class dissimilarity. Looking at different aspects of linguistic style, e.g., word or sentence length, vocabulary richness, and various frequencies (e.g., of words, word lengths, word forms, characters, or combinations thereof), stylometric approaches can help to analyze the author of a text. This general problem represents different applications, and, for instance, in our case, we want to tackle the task of authorship attribution, authorship verification, author profiling, authorship linking, and finally

author clustering.

1.2.1 Authorship Attribution

Computer-assisted authorship attribution aims to determine, as accurately as possible, the true author of a document or a text excerpt [32, 37, 49]. Under this general definition, the closed-class attribution problem assumes that the real author is one of the specified candidates as visualized in the left part of Figure 1.1. One can presuppose that the writing style is stable over the author's life, or at least during a few decades (e.g., between 20 to 40 years). To determine such personal stylistic aspects, a sample of texts written by each of the possible writers must be available. Based on specific stylistic representations or author profiles, the system can compute a distance (or similarity) measure between the unknown text and each possible author. The author with the smallest distance (or the highest similarity) is considered to be the actual author in this closed-class situation. We worked on this problem in a paper which we introduce and discuss in section 3.1. In the open-set situation, the real author could be one of the proposed authors or another unknown one (right side of Figure 1.1). If all probable authors show a similar distance (or similarity) and none of them is below a given threshold, then the text in question could be written by someone else. This second case is strongly related to the author verification task as explained below and we can approach it in the same way. Authorship attribution is usually the first problem to be solved when tackling authorship analyzing issues and can be extended to handle further cases.

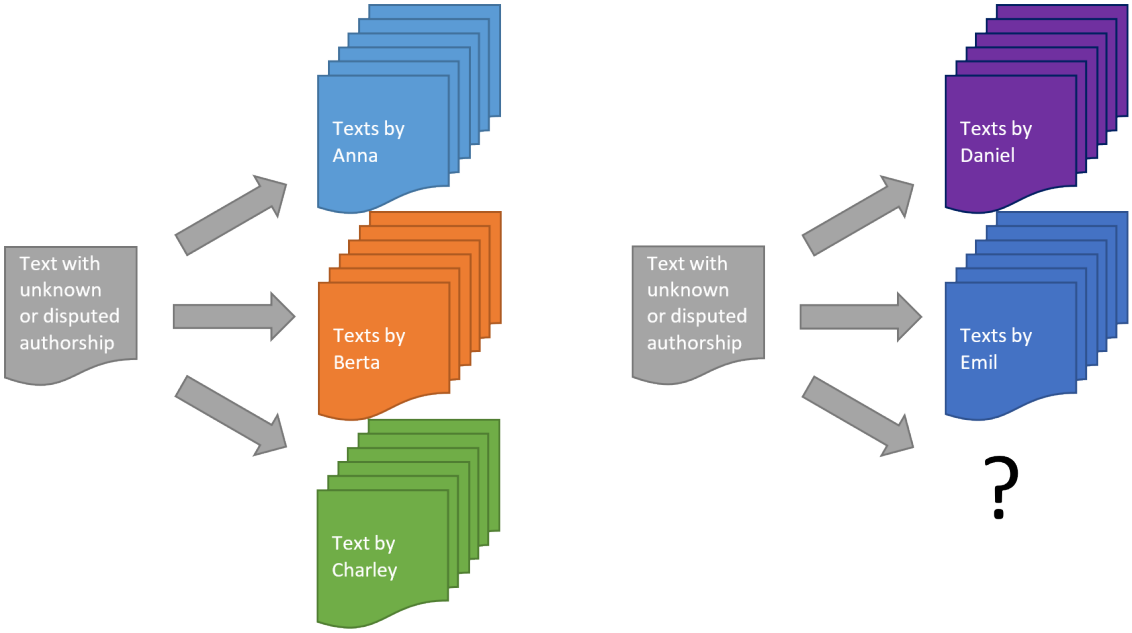


Figure 1.1 – Left: closed-class authorship attribution to decide whether the sample was written by Anna, Berta, or Charley. Right: open-class authorship attribution to decide whether the sample was written by Daniel, Emil, or someone else.

1.2.2 Authorship Verification

The authorship attribution model can be adapted so that it only determines whether a given author did, in fact, write a given document (chat, threatening e-mail, doubtful testimony) [28, 51] as visualized in Figure 1.2. At first, this question seems simpler than the classical authorship attribution problem because the answer is a simple binary answer (yes/no). However, the challenging part is to refute a shared authorship and to find a suitable threshold for the similarity when we can still verify a common origin of two texts. For example, if we want to know if a newly discovered poem was written by Shakespeare [11, 53], the system 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 because we are never sure we have included all other writers having a style similar to Shakespeare. Furthermore, instead of being limited to a single decision, the system should return a probability (or degree of credibility) that the proposed author is the real one or that the answer is correct [44], as well as some justification supporting the outcome. Finally, if the system is unsure if it should verify or contradict a shared authorship, it would be better to state that there is not enough evidence to make a decision instead of risking a wrong answer and losing the user's trust in the system. We worked on this problem in a paper which we present and discuss in section 3.2 and a working notes paper [24].

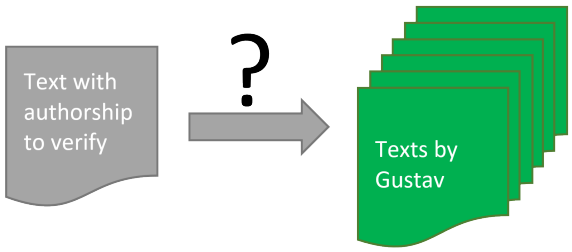


Figure 1.2 – The authorship of the text on top has to be verified or refuted to be written by the given author Gustav.

1.2.3 Author Profiling

Do men write like women, or are there significant differences in their writing styles [22]? What are the features that best discriminate typings by separate age groups [3, 40]? Is it possible to reliably detect somebody's personality traits based on a text excerpt? Similarly, can we detect the features that best discriminate writings by several language varieties? The spelling difference between British English and American English is well defined, but can we detect a variation from the US to Canada, or Ireland and Great Britain, and can we discriminate between New Zealand and Australia?

We can transform these author profiling issues to authorship attribution questions with a closed set of possible answers [25]. Determining the gender of an author can be seen as attributing the text in question to either the female or male authors. Similarly, the age group detection takes one of four or five groups to attribute the

unknown text. To uncover the Big Five personality traits (extraversion, neuroticism, agreeableness, conscientiousness, and openness on an interval scale from -5 to +5 with a step size of 1) this approach is taken even further by selecting for each factor one of eleven categories. And the language variety detection in an unknown Spanish text can take one of seven groups ("Argentina", "Chile", "Colombia", "Mexico", "Peru", "Spain", or "Venezuela"). Figure 1.3 shows a general visualization of the task. Furthermore, when we profile Twitter tweets, the spelling may not always be perfect, and more sociocultural traits could be detected. We worked on this problem in various working notes papers [22, 25, 27].

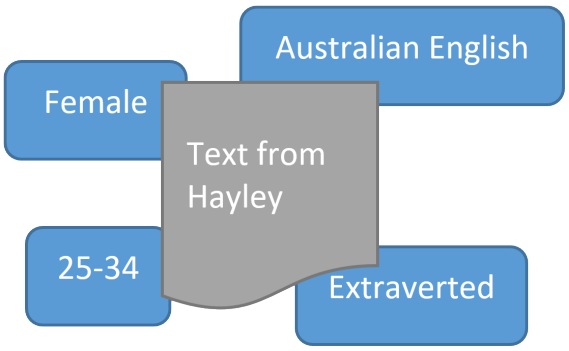


Figure 1.3 – The author can be profiled based on a text excerpt to determine, for example, her[1] gender, age, language variety, and personality traits.

1.2.4 Authorship Linking and Author Clustering

For all previous tasks, it was possible to gather training data because a set of documents written by the different possible authors (or categories of writers, such as men and women) can be collected to learn which features are pertinent, what the optimal parameters are, and which threshold works the best. Here, we have a different situation where such labeled data is not available. The first targeted question, called authorship linking, is defined as follows. Having a set of n documents (or text excerpts) created by several distinct writers, find the pairs of texts generated by the same person. Therefore, the goal is to produce a ranked list containing document pairs with a shared authorship. We worked on the linking problem in a paper which we present and discuss in section 3.4 and two working notes papers [23, 26].

In the related task called author clustering, the objective is similar and usually extends the linking task. Here, the number k of distinct authors must be determined to form k separate clusters based on a preset threshold for the ranked list of authorship links. A set of documents (or text excerpts) may be regrouped in different clusters for each author (authorship attribution), for each demographic class (e.g., gender, age, personality traits, language variety (author profiling)), or according to any other factor as visualized in Figure 1.4. One of the biggest problems is that there is no way to tell beforehand how many clusters we have to create, i.e., it could be one cluster for each document individually, or only one group with documents,

[1] There is no gender-neutral third-person singular pronoun in the English language, but we believe that using pronouns makes writing about authorship easier. Here, the author is a she and in others, she can be a he without loss of generality.

or (more probably) something in between. In this current study, there is no training data available that would allow an estimation of the cluster size distribution and an unsupervised approach must be designed and evaluated. As possible applications for both tasks, we can regroup a set of proclamations written by different terrorist groups, link the reviews written by the same author in a collection [1], or gather a set of poems (or excerpts of literary works). The clustering problem was the topic in various papers which we present and discuss mainly in section 3.5 and section 3.6 and the same two working notes papers as the linking problem mentioned above.

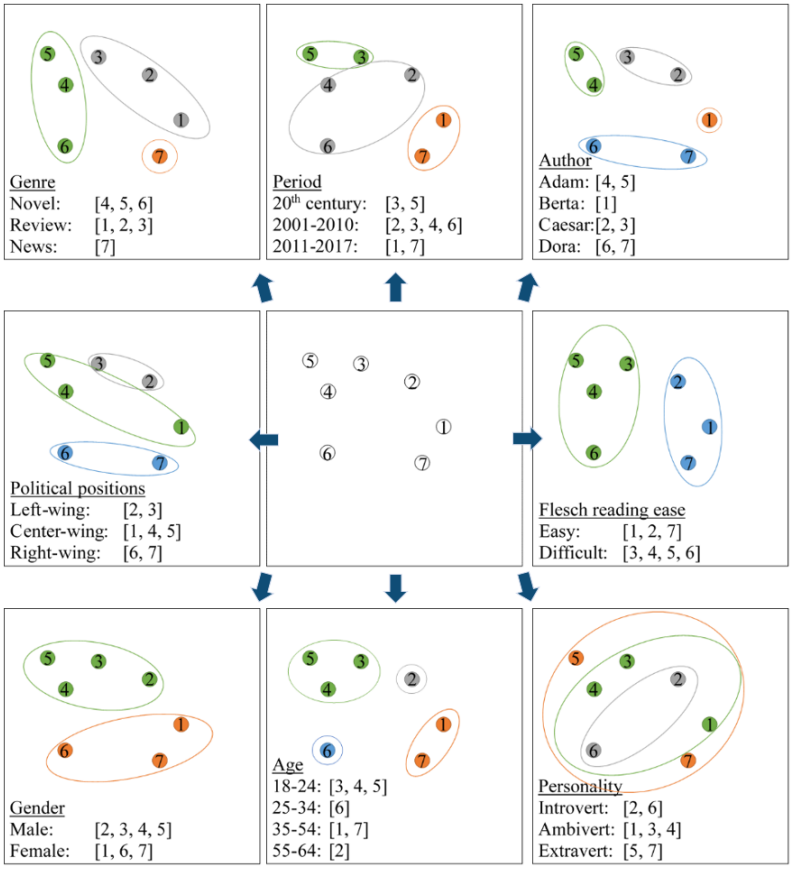


Figure 1.4 – Texts grouped in clusters according to different properties.

1.3 Achievements

In this dissertation, we examine text clustering with styles. In the work for this thesis, we achieve the following milestones:

1. We test multiple inter-textual distance functions in theoretical and empirical tests. Some of the measures work well in theory but not always in practice and vice versa.

2. We compare different text representation strategies. Reducing the size of the text not only decreases the runtime but can also increase performance by ignoring spurious features.

3. We create different feature selection procedures. The SPATIUM chooses features most important to the text in question.

4. We design a measure reflecting the (un)certainty of the proposed decision. We designed the system such that every decision comes along with an estimation of confidence of correctness.

5. We evaluate distributed language representations as well as state-of-the-art methods for authorship attribution. Not every approach is better in every aspect, and the deep learning methods can be sensitive to parameter settings.

6. We create a simple framework to cluster texts based on the writing style. Using a dynamic threshold the relative smallest distance values can be chosen.

7. We apply our system in various natural languages belonging to a variety of language families and in multiple text genres. With a flexible feature selection, the system works in any of the tested settings.

The main takeaway message is that a simple approach can lead to solid results in a short time while more complex models do not necessarily improve the result significantly but always take longer to complete. Furthermore, the results of our simple approaches can be justified in plain English because they are not based on a black box approach but use frequent words and the difference between them.

1.4 Organization of the Thesis

The remainder of this dissertation is structured as follows.

Next, in chapter 2, we show the essential steps and challenges any authorship analysis system (in our case for authorship attribution, author profiling, authorship verification, authorship linking, or author clustering) has to face by presenting relevant related works. Furthermore, we briefly introduce an overview of the testing methods and performance measures that we will use throughout the discussion of the following chapters.

We present the various publications upon which we base this thesis in the central part in chapter 3. First, the chapter provides the necessary context to understand the role of the individual papers. The rest of that chapter then contains individual sections for each paper. In each of these sections, we will briefly summarize the main aspects of the article, particularly concerning its importance to our goal of developing a text clustering system based on the writing style.

Finally, chapter 4 summarizes the results of our contributions and highlights the significance of our findings with their relevance to the state of the art in text clustering. In the end, we look into the future, seeking ways to improve text clustering effectiveness or efficiency by looking at various interesting issues that remain more or less unresolved.

The complete articles, containing results, discussions, as well as the full list of references, can be found in Appendix A. In Appendix B, we list our publications of journal articles, conference proceedings, and the working notes in evaluation forums. Appendix C shows the Voronoi diagrams of several distance measures.

Chapter 2

Authorship Analysis

There exists a broad body of literature which covers various techniques to solve different authorship analysis problems and studies them according to numerous perspectives. We have to solve three main challenges to achieve an effective solution for any authorship analysis task (authorship attribution, authorship verification, author profiling, authorship linking, or author clustering). First, a text representation must be defined reflecting the stylistic aspects of the author, without taking account of the genre or the topics explicitly. Therefore, we must select the most pertinent features from each text document for each category. Second, a useful distance measure between two text representations must be determined. Such a function must return a small value when the same author has written the two documents and a larger one otherwise. Instead of applying a distance measure, a similarity measure can be used to state that two texts were written by the same person when the similarity value is high enough.F Finally, a classifier has to assing a sample to a category depending on preset or learned parameters.

2.1 Feature Selection

The first challenge is to represent the text suitably, and it is not clear which text representation proposes the highest effectiveness. To describe the stylistic aspects of an author, the first set of methods suggests defining an invariant stylistic measure [17] reflecting the particular style of a given author and varying from one person to another. Different lexical richness measures or word distribution indicators are possible solutions. For instance, Yule's K measure, statistics related to the type-token ratio (TTR) (e.g., Herdan's C, Guiraud's R, or Honoré's H), the proportion of word-types occurring once or twice (e.g., Sichel's S), the average word length, or the mean sentence length [4, 45]. None of these measures have proven very satisfactory due, in part, to word distributions ruled by a large number of very low probability elements (LNRE) [4].

If words seem a natural way to generate a text surrogate, other studies have suggested using the letter occurrence frequencies [21, 31] or the distribution of short or long sequences of letters (character n-grams) [38]. As demonstrated by Kešelj et al. [20] such a representation can produce good results. We can justify this approach by considering that we can detect an author employing the continuous present time form more frequently by a high frequency of the tri-gram "ing" and verbal forms

related to the verb "to be" (e.g., "am", "is", "are"). As another example, one can identify more adverbial forms with a word ending in "ly". However, it is not clear which n value for the character n-gram is needed to offer the highest performance level, and this value may depend on the collection, language (e.g., $n = 2$ for Chinese, $n = 3$ to $n = 6$ for English), as well as other factors (e.g., text genre, OCR text) [34].

On social media channels and modern messaging services, new features have been appearing in recent years. The English language went from having 92 basic printable Latin characters (small & capital letters when ignoring diacritics (e.g., résumé, naïve) plus digits and punctuations & symbols[1]) to a language with thousands of characters in just a few years because of the addition of (the language independent) emojis[2]. Informally, we now encode (some interpretation of) facial expression, gender, and skin color in the texts we write on modern media. Those new features are only marginally researched but could provide indications to profile the sender of short and informal texts [27].

The Part-Of-Speech (POS) distribution can also be used to reflect the stylistic characteristics of the different authors. One writer may prefer using more noun phrases than verb phrases leading to nouns being more frequent in the text in comparison to adjectives. For instance, when comparing Presidents Kennedy's and Obama's speeches, one can see this difference, with Obama adopting more verbal constructions, i.e., a style oriented towards action ("yes, we can") [46]. Such a text representation does not usually produce a noteworthy result. On the other hand, instead of considering only the distribution of single POS tags, short sequences of POS tags can be more useful to detect some discriminative stylistic aspects of different authors. We could, for instance, distinguish a preference for Noun-Noun constructions (e.g., Information Retrieval) over Noun-Preposition-Noun forms (e.g., Retrieval of Information).

However, a simple count based on a single feature, and primarily based on content words, cannot provide a reliable measure. The text register has an impact on those predictors, as pronouns are in general less frequent in a formal context. Nonetheless, political speeches delivered by US presidents contain more pronouns, even when the context is official [45]. Therefore, we must view a generalization based on a single experiment or the use of a single text register with caution.

Gender distinction might be perceived as the simplest profiling task since the classification is a binary choice (at least as organized by the PAN[3] lab in the CLEF[4] evaluation forum and as utilized in this thesis). Therefore, we could easily collect a relatively large amount of data. Past studies also showed that the writing style between genders or between different age groups does differ and the stylistic differences can be detected [47]. According to Pennebaker [40], women tend to employ more personal pronouns (especially more "I" and "we") than men while in comparison using fewer determiners and prepositions. The difference is not huge, but it

[1] In ascending Unicode order: ! " # $ % & ' ( ) * + , - . / : ; < = > ? @ [\ ] _ { | } ~

[2] The Unicode Standard as of June 2017 contains 2,666 emojis including some sequences for genders or skin tones, flags, and the components that are used to create keycaps and other sequences, cf. https://emojipedia.org/

[3] http://pan.webis.de/

[4] http://www.clef-initiative.eu/

does exist. Usually, there is only a small percentage change in the relative frequencies. Since those features form a closed set, we can create a simple list of words. As an alternative, the LIWC (Linguistic Inquiry and Word Count) [52] proposes a set of word lists to measure semantic-based categories. For instance, negations, hedge phrases, cognitive, and social words are more related to female authors, while swear words, and references to money or numbers are more associated with men.

Past studies indicate that common word-types or function words can reflect the personal style of each writer closely while other researchers suggest taking account of the entire vocabulary and other experiments propose to ignore terms having a low occurrence frequency (e.g., appearing once or twice). For example, Burrows [9] suggested two distinct but complementary tests. The first one is based on words regularly used by one author but sporadically by the others while the second is grounded on words infrequently used by one author and ignored by the others. The most frequent terms are suitable in many cases. According to Zipf's law, the words used only once or twice (hapax/dis legomenon) make up around 50% of written texts [54, 55]. The advantage of using only the most frequent terms is that it can be dynamically computed based on the underlying text data without predefining specific lists. It is often possible to describe even complex concepts with a limited set of terms. For instance, using just the 1,000 most common words in the English language it is possible to describe everything from the periodic table and a car engine to the tectonic plates, the US Constitution, and the solar system in a simplistic manner [36].

2.2 Distance Measures

We can find further concerns when choosing the most appropriate distance (or similarity) measure between two text extracts. For example, in the information retrieval [33] or deep learning community [15], Cosine from Equation 2.3 corresponds to the most popular measure. In the first research domain, we can explain this because we have to compare the rather long documents to a short query (possibly a few words). In such a query, usually all of the terms appear just once, therefore comparing it to texts based on the individual frequency is not useful and we prefer a distance measure based on the inner product. In the word embedding field, the Cosine is used in combination with algebraic operations to detect relationships between words in the vector space created by the neural network. However, many other distance measures [13] do exist that provide a more nuanced differentiation, and their success in other authorship analyzing problems is largely unknown.

Choosing a suitable distance measure presents an important aspect for the authorship analyzing system. The choice depends on the field, and some domains employ the distance only implicitly. There should be a smoothness prior to the classification, meaning a slight change in the input data should either result in the same conclusion or give only a small variation of the outcome. The most simple distance measure is certainly the Manhattan distance as shown in Equation 2.1 or the Euclidean norm from Equation 2.2. Lastly, we present the Jaccard distance in Equation 2.4.

$$\Delta_{Manhattan}(A,B) = \sum_{i}^{m} |a_i - b_i| \tag{2.1}$$

$$\Delta_{Euclidean}(A,B) = \sqrt{\sum_{i}^{m} |a_i - b_i|^2} \tag{2.2}$$

$$\Delta_{Cosine}(A,B) = \frac{1}{\pi} cos^{-1}\left(\frac{\sum_i^m a_i b_i}{\sqrt{\sum_i^m a_i^2}\sqrt{\sum_i^m b_i^2}}\right) \tag{2.3}$$

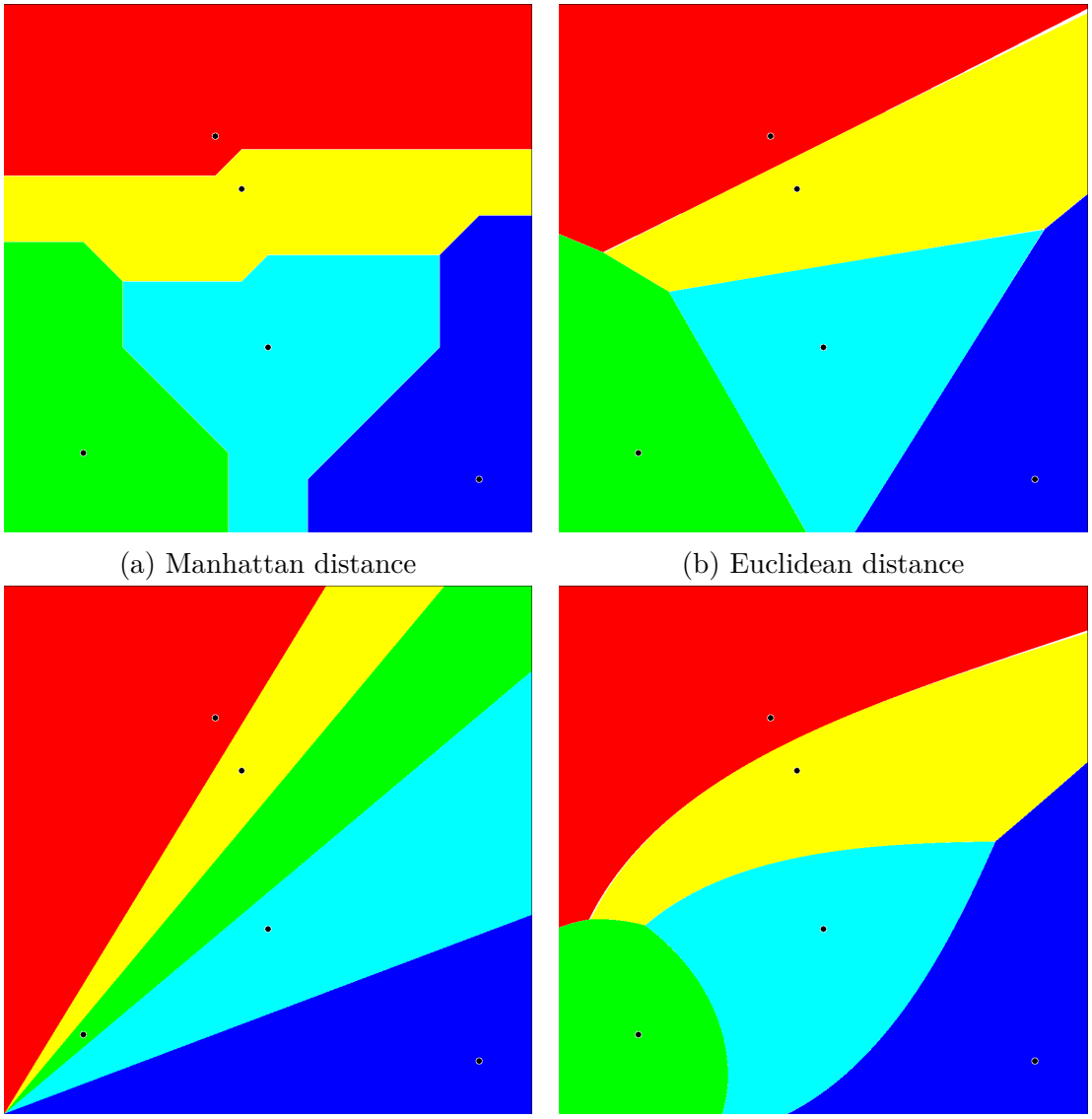
$$\Delta_{Jaccard}(A,B) = 1 - \left(\frac{\sum_i^m a_i b_i}{\sum_i^m a_i^2 + \sum_i^m b_i^2 - \sum_i^m a_i b_i}\right) \tag{2.4}$$

To visualize those distance measures, we used a heat map in the paper which we presented in section 3.3. Here, we draw Voronoi diagrams with preset dots to show the enclosed neighborhood areas. For each dot, there is a corresponding colored region consisting of all points closer to that dot than to any other. We have a vector space $[X \quad Y]$ that goes from the bottom left [0.0 0.0] to [1.0 1.0] in the top right and place the five dots at the positions [0.90 0.10] (blue), [0.40 0.75] (red), [0.15 0.15] (green), [0.50 0.35] (cyan), and [0.45 0.65] (yellow). As a simple illustration, we see in Figure 2.1a those five dots and their covering areas based on the Manhattan distance. We can observe that for this measure all borders between adjacent regions are either horizontal, vertical, or in a ± 45 degree angle. Compared to Manhattan, when using the Euclidean distance, the borders are all still straight but can be at any angle as shown in Figure 2.1b. In this case, the border is at the midpoint between two dots and is perpendicular to them. For some points, the distance to two dots differs less than 0.1%, and we left points in such regions therefore uncolored (white). The Voronoi diagram in Figure 2.1c of the Cosine distance depicts the angular measure well. Lastly, we can see in Figure 2.1d that the Jaccard distance includes a directional factor in addition to the requirement that the points have to be spatially close located.

We compare many distance measures in section 3.3 both from a theoretical standpoint as well as in various empirical test cases. In Appendix C we present the visualizations of the remaining interesting distance measures using Voronoi diagrams. Certain functions look strange at first (especially Canberra in Figure C.2c and Wave-Hedges from Figure C.2e), but they can be explained similarly as above. Some of them reward an exact match (or strong similarities) of an individual dimension more than an overall spatial proximity.

2.3 Classifier Choices

When we represent each text excerpt in a vector space with suitable features, and if we have a good distance measure, a machine learning approach can decide to which of a set of categories an unseen query text belongs. Therefore, having a training set with data containing samples whose category membership is known, a supervised learning method can be used. If no such training set containing correctly identified



(a) Manhattan distance (b) Euclidean distance

(c) Cosine distance (d) Jaccard distance

Figure 2.1 – Voronoi Diagram using different distances.

samples is available, an unsupervised procedure has to group data into clusters based only on some distance measure and a threshold value.

The k-nearest neighbors algorithm (k-NN) is a common method for classification where we analyze a test sample according to the majority of its surroundings. For instance with the 5-NN, when there are three samples labeled as category A and two as category B, the test sample would be assigned A independent of the ranking within those five representations. Therefore, the 3-NN could assign a different label if both of the B labeled samples are among the closest three samples. It is also possible that the k-NN has to solve a tie. For instance, if there are two of category A and category B each and one from category C, there's no clear decision. One possibility would be to iteratively decrease k until there is no longer a tie in the nearest neighbors and then take this decision. To avoid a tie verdict altogether, it is also possible to not represent each sample individually in the vector space but to average the samples of each category to a single point for the whole category and

check which is the closest. Another way would be to include the distance value and calculate the weighted average of the k-NN.

Linear regressions (or ordinal linear regressions) are well suited to predict scalar categories, like the level of extraversion as one of the psychological traits. Next to continuous and discrete quantitative variables, a linear regression could also be used to determine ordinal categories (age groups).

The naïve Bayes classifier only requires a feature set, but no distance measure has to be defined. In this case, the probability for each category to contain a given sample can be directly computed. Even though it assumes complete independence of the features without any possible correlation between them (which is clearly not satisfied for natural languages), the classifier can work well and requires only a small training collection.

Some proposed text categorization solutions employ a black box approach (e.g., deep learning, SVM, neural nets) that have been shown to be effective. However, such models have more difficulty justifying or explaining the proposed solution to the final user than other simpler approaches. In some cases, this is of no concern (e.g., as a character recognition system) but for authorship attribution or gender discrimination, a solution that can be justified in plain English seems to be a better approach (even if a reduction of the effectiveness has to be accepted). We prefer approaches like Burrows Delta [8], naïve Bayes, or especially an instance based k-nearest neighbor approach.

2.4 Evaluation Methodology

In this section, we will present testing methods to evaluate the quality of different aspects of authorship analysis systems, from author attribution and author profiling to author verification and author clustering. Machine learning models have parameters that can be fitted and optimized. We can not expect that our labeled training set covers every possible combination of features or that it is free of any noise. Therefore, future data probably won't match exactly with the data that we have to tune the system today. To assess the predicting performance of authorship analysis systems on unseen data we need solid testing methods. Furthermore, to compare multiple systems between each other, we need to convincingly show that one works better than the other by including a measurement of certainty.

The amount of labeled data is usually limited in practice. With the holdout method, we split the corpus into two parts for separate training (generally $^2/_3$ of the samples are randomly selected) and testing (the remaining $^1/_3$ of the samples) of the system. Optimally, we represent each class with approximately equal proportions in both parts, and this process can be repeated to achieve a more reliable estimate.

A better approach is cross-validation (CV) which avoids overlapping test sets. In the k-fold CV the data is split into k subsets of (roughly) equal size (usually $k = 10$). Then, in k rounds, each subset is used for testing, while the rest is used to train the system. For instance, in the 3^{rd} round of a 5-fold CV, the system trains with the subsets 1, 2, 4, and 5 to predict the classes in subset 3. The process can also be repeated multiple times to shuffle the samples in the subsets. Finally, the

overall error estimation or performance measure takes the average of the scores in all subsets.

Leaving-one-out (LOO) is a particular form of CV where the number of folds k is the number of training instances. This method makes the best use of the data, involves no random subsampling, but is computationally very expensive.

Today, the best classifiers are trained with supervision to determine the demographics of unknown writers and to study how they use the language. However, to achieve high accuracy, the training corpus needs to be large, diverse, and accurately annotated, which may be difficult to obtain. Furthermore, to estimate the standard error of the system's performance, a large number of corpora is required. An alternative to labeling huge amounts of data is to use artificial texts from a generator based on a preexisting corpus. The bootstrap is such a generator using a labeled dataset. In this approach, the system generates S new random bootstrap samples. A bootstrap copy has the same length, but the probability of choosing one given term (word or punctuation symbol) depends on its relative frequency in the original text. We draw the words with replacement; thus, the underlying probabilities are fixed. As the syntax is not respected, each bootstrap sample is not readable but reflects the stylistic aspects when analyzed as a bag-of-words.

With $S = 200$ copies for each sample, we can compare multiple schemes or parameter settings by comparing their respective average performance and confidence interval. Overall, this is a cheap method as there is no labeling cost and could further be used to simulate improper spelling and language usage. However, the synthetic texts may not be realistic, resulting in a lower performance on real test documents and making it only comparable to other bootstrap corpora.

2.5 Performance Measures

In this section, we will present measures to assess the quality of different aspects of authorship analysis systems, from author attribution and author profiling to author verification and author clustering.

For categorization tasks, such as in author profiling or author attribution, the most frequently used measure is the accuracy rate (or the percentage of correct assignments). We have two distinct schemes to compute this value. When we have the same number of texts for each category, both measures return the same value.

As a first method, the micro-averaging principle assumes that one decision corresponds to one vote. When the system can correctly identify, for example, the right class for 80 texts out of a total of 100 documents, the resulting accuracy rate (micro-average) is ${}^{80}/_{100} = 0.8$ or 80%.

As a second method, the accuracy rate is first computed for each of the c categories, under the assumption that the same importance can be attached to each category. In this case, one class corresponds to one vote (macro-average), and thus the overall accuracy rate is the mean of all classes. For example, with $c = 3$ possible categories, and with an accuracy rate of 0.8 for the first class, 0.7 for the second, and 0.6 for the third, the macro-averaging accuracy rate is ${}^{(0.8+0.7+0.6)}/_{3} = 0.7$, or 70%.

In authorship attribution studies, the micro-averaging technique is most frequently used to compute a mean performance. The argument in favor of this method is that categories should count proportionally to their frequency, and since most corpora are well balanced, the difference in the number of texts between categories is not too large, which results mostly in the same accuracy score.

Additionally, if the task requires categorizing multiple factors (e.g., age, gender, personality traits) for one document, the joint accuracy can be used. In this case, we divide the number of problems where we correctly predicted all factors for the same sample by the total number of problems in the corpus.

Finally, if we allow the system to not answer in case of uncertainty, we can use the $c@1$ [39]. Having n questions with n_c correctly classified and n_u unsolved answers, the $c@1$ is defined as follows:

$$c@1 = \frac{1}{n} * (n_c + \frac{n_u * n_c}{n}) \tag{2.5}$$

If the system provides an answer for all open problems, then the $c@1$ is the same as the traditional accuracy measure. However, $c@1$ rewards approaches that maintain the same number of correct answers and decrease the number of incorrect answers by leaving some questions unanswered (indicating that the provided evidence is not strong enough to make a final decision). For example, with $n = 100$ and $n_c = 80$ ($n_u = 0$), the accuracy rate is 0.8, and $c@1$ gives the same value. However, when the system leaves 10 of the incorrect decisions without an answer ($n_u = 10$), the $c@1$ does not view them as fully wrong and considers them to be correct with a probability of 0.8, so we get $c@1 = 0.88$. Therefore, open problems add value to the $c@1$ as if they are answered with the accuracy already achieved.

Additionally, we can have different weights for correct, omitted, and incorrect answers to measure the "merit" of the system. Of course, the ultimate goal is to reach a zero-mistake rate. When an error-free system is unlikely, we should penalize the wrong decisions. To reflect this aspect, a right answer counts as +1 point, the decision *don't know* adds +0.5, while we attribute -2 for an incorrect decision. Providing wrong answers surely hurts the credibility of an automatic system.

For example, having again $n = 100$ problems with $n_c = 80$ correct answers, $n_u = 10$ unsolved cases, and $n_i = 10$ incorrect decisions, the merit score would be $1 * 80 + 0.5 * 10 - 2 * 10 = 65$. To maximize the merit rating, we could increase the thresholds to take a decision. If this can reduce the incorrect decisions to $n_i = 5$ and the correct answers to $n_c = 75$, on account of some correct answers becoming unsolved, and end up with $n_u = 20$ unsolved cases, the merit score would rise to $1 * 75 + 0.5 * 20 - 2 * 5 = 75$.

Faced with stupid or incorrect answers, the end user will lose his confidence in the approach. Such an attribution scheme cannot be used, for example, to support court decisions. We actively prefer a system able to know when "it does not know" and provide an answer when there is enough evidence to make a choice.

To evaluate the clustering output, we can measure the purity and completeness of the generated clusters. In this perspective, a perfect system must create only k clusters, each containing all the documents written by the same person. Moreover, each document must belong to exactly one cluster; thus, the clusters must be non-overlapping. The evaluation measures are for each document the precision, the

recall, and the harmonic mean of the two previous values (denoted *BCubed* F_1) [2]. To evaluate this, we first compute the cluster correctness function $cc(d_i, d_j)$ between two documents d_i and d_j as follows:

$$cc(d_i, d_j) = \begin{cases} 1 & if\ A(d_i) = A(d_j) \wedge C(d_i) = C(d_j) \\ 0 & \text{otherwise} \end{cases} \tag{2.6}$$

where $A(d_i)$ indicates the real author of d_i, and $C(d_i)$ is the cluster in which d_i appears. Based on this notation, the document precision $pr(d_i)$ and document recall $re(d_i)$ are defined as:

$$pr(d_i) = \frac{\sum_{d_j \in C(d_i)} cc(d_i, d_j)}{|C(d_i)|} \tag{2.7}$$

$$re(d_i) = \frac{\sum_{d_j \in A(d_i)} cc(d_i, d_j)}{|A(d_i)|} \tag{2.8}$$

The document precision represents how many texts in the same cluster are written by the same author. Therefore, this measures the purity of the cluster (the absence of noise). Symmetrically, the recall associated with one document represents how many documents from that author appear in its cluster. Therefore, this measures the completeness of the cluster. We could generate one cluster per document to achieve a perfect precision, which is a solution that produces a strong baseline if there are numerous small clusters. Therefore, the purity of each group is maximal, and the resulting precision is 1.0. Similarly, to achieve a recall of 1.0, all documents can be regrouped into a single cluster. Thus, the two measurements are in opposition. Having n texts in a given collection, the *BCubed precision* and *recall* for the whole corpus is defined as:

$$precision = \frac{1}{n} \sum_{i=1}^{n} pr(d_i) \tag{2.9}$$

$$recall = \frac{1}{n} \sum_{i=1}^{n} re(d_i) \tag{2.10}$$

from which the well-known F_1 performance measure can be computed, as

$$F_1 = \frac{2 * precision * recall}{precision + recall} \tag{2.11}$$

with a higher value meaning a better performance and a better distribution of the clusters. The *BCubed* F_1 will serve as the main effectiveness measure in our final research.

In Figure 2.2 we see an example of a suboptimal clustering output on the left side, and how it should have been on the right side. There are two authors with one having written only a single document while the other has written three texts and the goal is to create the clusters accordingly. However, in this example, two clusters of equal size were created. The document precision for both text A and B is 0.0 because none of the texts in their cluster should be in that cluster. At the same time, for text C and D we have 100% document precision as they are in

a pure cluster. This example gives us a *BCubed precision* of 0.5 for this output. The document recall of text B is 1.0 (nothing is missing), while for A we have $^1/_3$ document recall because it is isolated from both C and D who in turn have a $^2/_3$ document recall. Therefore, this results in a *BCubed recall* of 0.666 and the *BCubed* F_1 is 0.571 overall for this output.

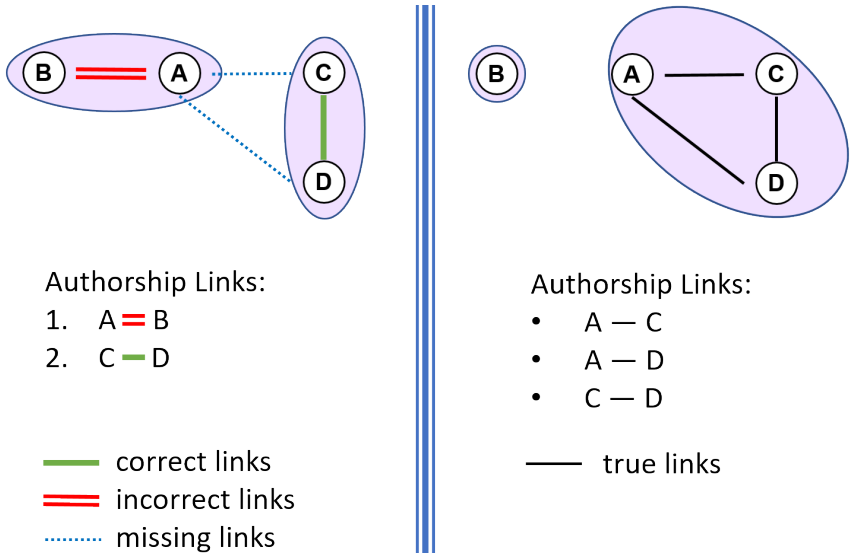


Figure 2.2 – Example of author clustering and authorship linking output (left) and the corresponding truth (right).

As a second measure, one can ask the clustering algorithm to return a list of links between text pairs, ordered by an estimated probability of having the same author for the two cited documents. To evaluate such an ordered list, one can apply the average precision (AP) [50]. If we assess the AP of a single system in multiple collections, we use the mean of the AP, called mean average precision (MAP). For this, we first compute the link correctness function $lc(i)$ at the i-th position of the ranked link list containing document pairs as follows:

$$lc(i) = \begin{cases} 1 & if\ A(d_{i1}) = A(d_{i2}) \wedge d_{i1} \neq d_{i2} \\ 0 & \text{otherwise} \end{cases} \tag{2.12}$$

where $A(d_{i1})$ and $A(d_{i2})$ indicate the real author of the first and second document in the link at the i-th rank. The *precision* at a rank i and the final AP of the whole list are then defined as:

$$precision(i) = \frac{\sum_{j=1}^{i} lc(j)}{i} \tag{2.13}$$

$$AP = \frac{\sum_{i=1}^{|L|} precision(i) * lc(i)}{|R|} \tag{2.14}$$

where $|R|$ specifies the number of true (relevant) links and $|L|$ the number of created links. Using Equation 2.13, a common performance value is provided at rank 10 (denoted $Prec@10$) or 20 ($Prec@20$). These two limits are frequently adopted in the information retrieval domain because they correspond to the first two pages of results returned by a commercial search engine. The number of true links in the test collection $|R|$ defines another interesting limit. This limit varies from one test

collection to another, and we denote it as $RPrec$. Finally, to measure the capability of a system to return only good results (or links in this context), one can measure its high precision (denoted $HPrec$) by indicating the $rank - 1$ of the first incorrect answer appearing on the top of the returned list.

For instance, taking the example in Figure 2.2 from the clustering output with the created links again, we can calculate the AP. The goal is to create three links to connect all documents in the bigger cluster together while leaving text B unlinked. However, there are two missing and one incorrect link in the output. Additionally, we see that the correctly created link from text C to text D is ranked lower than the incorrect link from document A to B. We therefore have $precision(1) = 0.0$ and $precision(2) = 0.5$ with $|R| = 3$ which gives us an AP of 0.166. As the number of created links $|L|$ is smaller than the number of relevant links $|R|$ in this example, the $RPrec$ is the same as the AP. Since the highest ranked pair is an incorrect link, the $HPrec$ is 0.

AP is a standard evaluation measure in the IR domain [33]. The AP is roughly the average area under the precision-recall curve for a set of problems. This measure is sensitive to the first rank(s), and providing an incorrect answer in the top ranks hurts the AP value intensively. Nonetheless, AP does not punish verbosity, i.e., every true link counts even when appearing near to the end of the ranked list. Therefore, by providing all possible authorship links, one can attempt to maximize AP [41].

2.6 From Analyzing to Clustering

Our goal is to build a text clustering system based on the writing style that can reason its decisions back to the end-user. We explain our procedure for this in detail in chapter 3.

For all authorship analyzing tasks, a text collection covering individuals with diverse demographics is needed. A varied corpus is better for the system to learn or deduce the writing style of particular groups of people or a specific author. The sentence structure and lexical usage can be unique and serve as a "fingerprint" in a text and can be better determined if there is more data available. Our corpora are rather short (e.g., we have 1,200 short samples of male Arabic authors), but organizations (in particular on the national level) can collect an enormous amount of data. For instance, the NSA, through PRISM[5] (the program to gather data from Internet communications that matches court-approved search terms from US Internet companies) and MUSCULAR[6] (where the NSA copied the unencrypted data flows from the main communications links between data centers of Google and Yahoo) are theoretically able to compare billions of author profiles with a certain author's writings to find her true identity.

From the beginning, we have evaluated our work in the CLEF evaluation campaign by participating in multiple PAN labs (namely in the years 2014, 2015, 2016, and 2017). Working in the context of an evaluation campaign allows performance as-

[5] https://en.wikipedia.org/wiki/PRISM_(surveillance_program)
[6] https://en.wikipedia.org/wiki/MUSCULAR_(surveillance_program)

sessments across systems by numerous participants, and helps to avoid comparisons to weak baselines. The corpora used in our analysis were collected from newspaper articles, crawled from publicly available Twitter profiles, written by students as an essay or a review, or extracted from various novels.

It is a valuable endeavor to explore the full space of authorship analysis techniques and iteratively build a reliable and stable system. When having a suitable text collection, we can first create a system to attribute an unknown text to one out of a closed group of possible authors (section 3.1). The extension to this is if one from the group of potential authors or another unknown author may have written the sample. For this, we need to be able to verify or controvert a shared authorship for a pair of texts (section 3.2). Similarly, when profiling a text (e.g., male or female; 10s, 20s, or older), the sample has to be classified as one out of a limited number of categories depending on the most similar observations or the category with the smallest distance (section 3.3). Then, we can link those pairs of texts where we see an indication of a shared authorship and have enough evidence that the same person has written them (section 3.4). Finally, we check for every text tuple to determine if we can link them together and build the final clusters based on different strategies (section 3.5 and section 3.6).

Chapter 3

Presentation of the Publications

The core of this dissertation is based on the following six publications.

- Mirco Kocher, Jacques Savoy.
  Distributed Language Representation for Authorship Attribution
  In *Digital Scholarship in the Humanities*, to appear.

- Mirco Kocher, Jacques Savoy.
  A Simple and Efficient Algorithm for Authorship Verification
  In *Journal of the American Society for Information Science and Technology*, 68(1), 259-269, 2017.

- Mirco Kocher, Jacques Savoy.
  Distance Measures in Author Profiling
  In *Information Processing and Management*, 53(5), 1103-1119, 2017.

- Mirco Kocher, Jacques Savoy.
  Evaluation of Text Representation Schemes and Distance Measures for Authorship Linking
  In *Scientometrics (Special Issue Proposal on Scieno-Network-Mining)*, submitted.

- Mirco Kocher, Jacques Savoy.
  Author Clustering Using Spatium
  Short Paper JCDL 2017, Toronto, Canada, June 19-23, 2017, ACM/IEEE, 265-268.

- Mirco Kocher, Jacques Savoy.
  Author Clustering with an Adaptive Threshold
  In *Jones, G. J. F., Lawless, S., Gonzalo, J., Kelly, L., Goeuriot, L., Thomas M., Cappellato, L., & Ferro, N. (Eds)*, Experimental IR Meets Multilinguality, Multimodality, and Interaction, Proceedings of the 8th International Conference of the CLEF Association, CLEF 2017, Dublin, Ireland, September 11-14, 2017, 186-198.

These contributions address various questions concerning motivations and objectives regarding the goal of successively building an effective and robust text

clustering system with styles. The following sections give an overview of the content and contributions for each of these publications by first exposing the questions they raise and eventually presenting the approach taken to provide answers. Each section contains the central insights of the contribution while all the articles, containing results, discussions, as well as the full list of references, can be found in Appendix A.

We first start in section 3.1 by comparing different methods to solve the authorship attribution problem. Then, in section 3.2, we extend our gained knowledge by verifying the authorship of a given text. In section 3.3 we have a more fundamental look at the core of authorship analyzing approaches by evaluating numerous distance measures in author profiling tasks. We evaluate different text representation schemes as well as distance measures in section 3.4, specifically for the authorship linking task. Afterwards, in section 3.5, we present our first approach for solving the author clustering problem and conclude in section 3.6 with the final assessment for reaching the goal of text clustering with styles.

3.1 Distributed Language Representation for Authorship Attribution

Deep learning approaches have become quite popular lately by producing effective results, particularly in image recognition. Similarly, word embedding now appears in many natural language processing proposals and we started investigating the use of the underlying word vector representation in the work of this thesis. A solution to the authorship attribution problem is the first step for our authorship analyzing system with the final goal to solve the task of text clustering with styles.

As stylistic features able to discriminate between different authors, numerous studies propose to use the most frequent words or function words (determiners, pronouns, conjunctions, prepositions, and auxiliary verb forms) [8, 16, 43]. Those features are usually extracted from the text without considering their context. Only a few authorship experiments proposed generating document representations based on words in their context. However, those were mainly limited to bigrams of terms used directly [19] or employed to produce word networks instead [35]. In this study, the authorship of a text will be determined based on the relationships between terms using deep learning [5, 14].

We propose and implement two new classifiers based on such a distributed language representation. In this perspective, a vector-space representation can be generated either for each author or each disputed text according to words and their nearby context. Every word is nested somewhere in a high dimensional vector space such that nearby words share the same meaning, e.g., synonyms would be grouped close together. With simple linear algebraic operations, the semantic relationship between words can be explored, e.g., we see that the result of vector(France) - vector(Paris) + vector(Italy) is very close to vector(Rome).

For the first model, we sum up the vectors of each word from a bag-of-words to create the complete text representation in this vector space. Additionally, the probability of each word appearing in the text scales each vector, meaning the weights correspond to the relative frequencies. Therefore, a document vector is the concatenation of its word vectors. In the second model, we embed the full document in the vector space without the intermediate step of word vector combinations. This representation allows the model to capture the meaning of each word better and thus, to be similar, two texts must not only share the same words, but those words must appear in similar contexts (and with similar frequencies).

To determine the authorship of a disputed text representation, the Cosine similarity between the vectors is usually applied, and then the k highest Cosine similarities are inspected. In our experiments, we had four diverse test collections available (the *Federalist Papers*, the *State of the Union* addresses, the *Glasgow Herald*, and *La Stampa* newspapers). Therefore, we show that the proposed strategies can be adapted without any difficulty to different languages (such as English and Italian) or genres (essays, political speeches, and newspaper articles). Furthermore, we compare its effectiveness with baseline methods where we treat each text as a bag-of-words, such as in the k-NN (traditionally, without deep learning), NSC (nearest shrunken centroid), chi-square, Delta, LDA (latent Dirichlet allocation), and multi-layer perceptron (MLP) classifiers.

The deep learning schemes offer a broad range of parameters that can be tuned and optimized. We show that the proposed default values for the learning rate, vector size, and context window size tend to produce a high-performance level where small variations around them do not modify the obtained effectiveness significantly. Nonetheless, adjusting the topology of neural networks with a different number of hidden layers or a different number of nodes in each of them has a notable influence on the achieved accuracy.

Our two proposed models and the evaluations described in the paper from section A.1 indicate that such a distributed language representation performs well, providing sometimes better effectiveness than state-of-the-art methods. More specifically, we observe that the second model (document vector) shows better results than the first model (concatenation of word vectors), and the second model also tends to achieve high accuracy rates compared to the selected baselines.

However, the proposed decision of the authorship attribution cannot be traced back to easily understandable reasons due to the system being a black box. Therefore, the final user does not get any explanations justifying the proposed attribution and estimating some degree of support or belief that the proposed author is the right one is harder to specify concretely.

The effectiveness of all methods depends mostly on the size of the corpora. Moreover, the results indicate that none of the proposed authorship attribution methods dominates all the others in every case. For instance, k-NN can outperform many approaches in the smallest data set while NSC works especially well in the biggest collections. Different strategies can provide the best performance for individual corpora (e.g., MLP or LDA for the *Federalist Papers*, NSC with *La Stampa* corpus, the second distributed learning model for the *Glasgow Herald* collection). However, for the two biggest corpora (*Glasgow Herald* and *La Stampa*) having a large number of possible authors for texts published during the same year, the deep learning approaches provide the highest accuracy rates.

To build our system for text clustering with styles, we learned that a simple authorship analyzing model could compete with more complex ones.

3.2 A Simple and Efficient Algorithm for Authorship Verification

Simply attributing a text sample to a set of authors is plausible as long as the list of possible authors is limited and complete. Otherwise, we have to have a method to verify or contradict the authorship of a particular writer for an individual text sample. If the writing style used in a query text Q is similar enough to the one employed by the author A_v that we want to verify, then we can confirm the shared authorship. Otherwise, we can contradict it. Nonetheless, deciding if A_v wrote Q should not just result in a binary answer, but ideally, a degree of certainty is obtained. The problem is to define what "similar enough" writing style means, if there exists a space between verification and contradiction, how big that range is, and finally a way of computing the degree of certainty.

As features, we extract the 200 most frequent tokens (MFT, words and punctuation symbols) solely from Q. Therefore, we do not use a general feature list but collect the features dynamically depending on the text in question. This selection means the classifier does not require a learning stage to perform the feature selection. Using the MFT from Q assures that we apply features that are the most pertinent for the text in question. This representation has the additional benefit that the features are independent of the language (only a tokenizer that can handle the script is required) or genre used. In our case, we successfully tested it with the Dutch, English, Spanish, and Greek languages and the essay, novel, review, and newspaper article genres.

Although intuitively the concept of a *word* seems simple, there is no absolutely "correct" number of words in a text. The number of words depends on a series of decisions about what counts as a word. For instance in "Ann's reading in John's book." the tokenizer could split the sentence based on different rules. If the ten tokens

ann ' s reading in john ' s book .

are produced, the genitive possessive case "'s" from John is treated the same as the contracted third person singular form of "to be" from Ann. However, if we leave them as the five tokens

ann's reading in john's book.

then it is probable that the strings "ann's", "john's", and "book." are hapax legomenon and won't be useful in the authorship analysis. Different text-analysis programs and different analysts produce various tokenizations. There is no harm in this, but in every analysis, we have to treat the words in a consistent way across all of the texts [18].

If A_v has similar usage of the MFT, we can verify the shared authorship. As a distance measure, we use the Manhattan distance (L^1 norm), meaning the sum of the absolute differences of all features between Q and A_v, which we denote $\Delta(Q, A_v)$, as presented in Equation 2.1. This feature selection and distance calculation model, we called it Spatium-L^1 (a Latin word meaning distance), builds the foundation of our targeted system.

For example, assume that Text A corresponds to "The fox, the moose, and the deer jump over a wolf." After ignoring the letter case and the hapax legomenon, the resulting vector is $[the \quad ,] = [3 \quad 2]$ giving the final relative representation $[the \quad ,] = [0.6 \quad 0.4]$ based on the term frequency. Assuming Text B contains the following sentence: "The quick fox and the brown deer jump over the lazy dog and a cat." When computing the distance $\Delta(A, B)$, we use the terms {the ,} because they are extracted from the representation of Text A. The representation of Text B is therefore $[the \quad ,] = [1.0 \quad 0.0]$ and calculating the Manhattan distance between these two vectors gives us $\Delta(A, B) = 0.8$. Analogously, when estimating the distance $\Delta(B, A)$, only terms belonging to B's representation are considered, namely {the and}, giving us the representation $[the \quad and] = [0.6 \quad 0.4]$ for Text B and $[the \quad and] = [0.75 \quad 0.25]$ for Text A, resulting in a distance $\Delta(B, A) = 0.3$. This measure is not symmetric due to the choice of the terms.

To verify whether the resulting distance is small rather than large, we randomly select m impostors (e.g., $m = 3$) from a text collection (in the same language and same genre) and only retain the distance of the closest sample. We repeat this step r times (e.g., $r = 5$) and then compute the mean over those smallest distances in the r rounds to accommodate the probabilistic influence. The mean smallest distance from the text in question to the impostors $\Delta(Q, A_{mr})$ is then compared to $\Delta(Q, A_v)$. If $\Delta(Q, A_v)$ is more than 2.5% below (or above) $\Delta(Q, A_{mr})$ we can confirm (or contradict) that Q was written by A_v. Finally, if the value of $\Delta(Q, A_v)$ lies within $\pm 2.5\%$ of $\Delta(Q, A_{mr})$ then we can neither confirm nor contradict the shared authorship. The limit of this 5% window was chosen arbitrarily but corresponds to a typical threshold value in statistical tests. Based on this simple rule, we can define when there is enough evidence to propose an answer or when the attribution scheme is unable to decide with a high degree of certainty.

Compared to the results of the PAN CLEF 2014 evaluation campaign, our proposed attribution scheme usually achieved a performance among the three best systems within the six different test collections. When computing an overall mean over the six test collections, this approach shows the best performance level. The 5% decision window gives us a significantly higher $c@1$ than when always taking a decision and only considering the raw accuracy.

In the PR domain (Public Relations) it is known that a happy customer will talk to only 4 to 6 friends about a good event but a dissatisfied user will tell 9-15 people if they had a bad experience [7]. This phenomenon is relatively unknown in the academic world where the traditional performance measures tend to underestimate the real "cost" of wrong classifications. Inspecting the merit score, we can see that the SPATIUM-L[1] system generates a good overall performance and is more cautious about the danger of incorrect decisions. We can explain the high result by the fact that this verification scheme tends to opt more often for a *don't know* answer when the decision is uncertain than risking a wrong answer. Having enough evidence (surpassing the 2.5% threshold), SPATIUM-L[1] is then able to propose either an affirmative answer or not. Therefore, we have a method to decide for each text pair if the same person wrote them, and the degree of certainty with which we can give the answer to this question.

To build our text clustering system with styles, we learned that focusing on

those words that are most pertinent for the writer of a given document can capture the most relevant information. Furthermore, we noticed that a relative approach to detect the notably small distances is useful, and high performance can be achieved to verify a shared authorship with a computationally inexpensive approach. Finally, we showed that we could give a degree of certainty with the decision relatively easily to increase confidence in the system.

3.3 Distance Measures in Author Profiling

The SPATIUM model we introduced in section 3.2 can be used for various authorship analyzing tasks. Determining some demographics about the author of a document (e.g., gender, age) has attracted many studies during the last decade. To determine the targeted category, studies have suggested different distance measures without one approach dominating all others. In this paper, we evaluate the model on author profiling problems using different distance measures. More specifically, we study 24 measures from five general distance families of functions, like the L^1 family (e.g., Manhattan, Canberra), L^2 family (e.g., Euclidean, Clark), inner product family (e.g., Cosine, Jaccard), Entropy family (e.g., KLD, Topsoe), and combinations thereof (e.g., Average, Taneja).

We check if each function satisfies six theoretical properties we find useful to solve any categorization task in a vector space. Four of them are the conditions required for a distance to be a mathematical distance metric. That is, the distance from a vector to itself has to be zero (identity of indiscernibles), and if the two vectors differ, then a positive distance has to be returned (positivity). Furthermore, the ordering of the vectors in a distance function must not matter (symmetry). The last mathematical property of a metric is that the insertion of an intermediate vector cannot shorten the total distance (triangle inequality). This attribute should make the distance measure stable and robust (small variations or small errors in the input result in small distance changes). We also proposed two properties that are especially useful for authorship analysis and incorporated well with the SPATIUM system. The first one is that missing a frequent feature is worse than missing an occasional feature. This rule means that if one text shows a high usage of a particular word while the other document does not use it at all, it is less likely that the same person wrote them. Similarly, if one author sporadically writes something that the other one does not, then this is less striking. The second requirement is that we think it is better to have any feature than to miss any feature. This means if a particular word has a 2% probability of showing up from one author, then it is better if the other author uses the same word with double the likelihood (4%) than completely missing out on it (0%) even though the difference between them ($\pm$2%) is the same.

We show that the Tanimoto and Matusita distance measures respect all those properties with most of the remaining ones satisfying five of them. Looking at their definition, the difference between these 24 distance measures is usually rather small. We could also require that the measure should have a clear interpretation for the end user (no black-box system). Looking at the heat map in the paper or the Voronoi Diagrams in Appendix C, we see that not all measures result in a clear visualization. While this criterion is subjective to the observer, some measures are easier to interpret than others.

Using the SPATIUM model, we empirically evaluate the performance of the distances with a k-NN approach to predict the author's gender (male or female) and age (four or five groups) in 13 test collections from the previous CLEF campaigns. This test set covers four languages (Dutch, English, Italian, and Spanish) and four text genres (blogs, reviews, social media, and tweets) created for PAN 2014, 2015, and 2016. The empirical evaluations indicate that the Matusita, Canberra, Tani-

moto, and Clark distance measures tend to produce better effectiveness than the rest, at least in the context of an author profiling task. However, the Cosine, which is well-known in various distributed language models and the IR domain, tends to produce rather low-performance levels and never appears in the best five measures in any of the categorization tasks.

To assess the performance, we measure the accuracy of each category separately as well as the joint accuracy to determine the proportion of samples where we correctly predict both the gender and age group at the same time. Using the leaving-one-out approach, the mean accuracy in the 13 test collections is 37% for the joint score with Wave-Hedges resulting in the best average gender prediction (64.4%) and Matusita giving the highest score for the age groups with 48.7%.

Using a training collection from a different year (but in the same genre and language) has a negative influence on the effectiveness in the test set. For instance, when classifying Spanish Tweets from the PAN 2016 corpus, using the labeled Spanish Tweets from the PAN 2014 collection reduces the accuracy in all categories. More specifically, the absolute difference is on average -4% for the gender, -10% for the age groups, and -8% when predicting both at the same time in our corpora.

When the training and test sets are built from different genres, the effect is even worse on the performance. For instance, when classifying Spanish Tweets from the PAN 2016 corpus using the labeled Spanish blog entries from the PAN 2014 collection, the accuracy is further reduced in all categories. More specifically, the absolute difference over all collections is -7% for the gender, -11% for the age groups, and -9% for the joint accuracy in cross-genre categorization. Therefore, our experiments confirm that having a training set related closely to the test set (e.g., a collection both in the same genre and the same period) is of clear benefit for the overall performance.

To build our text clustering system with styles, we learned that distance measures such as Canberra or Clark are more useful than those based on the inner product or from the Entropy family.

3.4 Evaluation of Text Representation Schemes and Distance Measures for Authorship Linking

Before building the final author clustering system, we first link text pairs together that are most likely written by the same person. While there are similarities to the authorship attribution task, in this case, there is no training information provided, and the solution must be unsupervised. Therefore, based on n text excerpts, the authorship linking task is to determine pairs of documents for which we can verify a common origin. In section 3.3 we saw that the Matusita, Canberra, Tanimoto, and Clark distance measures tend to produce the best performances. Now we combine this knowledge with various text representation strategies that can be applied, such as character, punctuation symbols, or letter n-grams as well as words, lemmas, Part-Of-Speech (POS) tags, and sequences of them. From all those possible combinations of implementations, it is not clear which text representation and distance functions produce the best performance, and this section provides an answer to this question.

Three corpora, extracted from the French and English literature, have been evaluated using the AP (average precision as presented in Equation 2.14) as well as the $RPrec$. Moreover, we used the additional performance measure called high precision ($HPrec$) capable of judging the quality of a ranked list of links to provide only correct answers.

As distance measures, this study found that the Tanimoto, Matusita, or Clark distance measure perform better than the often-used Cosine function. We had the same observation in section 3.3 where we tested the measures in various author profiling tasks. This conclusion is a validation of the previous findings and shows that they also work well in the authorship linking task.

Comparing the token- and lemma-based text representation we found no systematic difference. While two out of the three corpora show higher AP for the lemma-based representation than for the tokens, the third collection indicates the contrary. On average, the AP difference is within $3-8\%$, relatively small, and also the $HPrec$ differs only $1-3$ ranks. When analyzing the variations related to the distance measures based on the two representations, none of them performed the best in all cases.

As suspected in section 2.1, simple POS tags do not provide practical features because they are rather limited compared to the vocabulary size. Nonetheless, short sequences of them form a good text representation where sequences of two or three tags improve the result significantly. The best performance with POS tags (and sequences of them) is usually below those achieved based on word-based representation. In some cases, however, the difference is rather small and can almost achieve the performance of token-based representations.

As a final text representation, we evaluate short sequences of letters, denoted n-grams, extracted from the document. Comparing the token-based representation with a combination of uni- and bigrams there are no notable differences. However, when using longer n-grams from $n = 5$ to $n = 7$, we see a significant increase in all performance measures. For example, the AP in the English corpus is increased by

more than 40% and the $HPrec$ raises from less than 60 to over 90.

We considered the entire vocabulary and all possible n-grams in the previous experiments. With n being a higher number (5 or more), the number of generated n-grams becomes huge, and most of them have a very low occurrence frequency. As we can assume a Zipfian distribution for the occurrence frequencies, the terms appearing only once or twice tend to correspond to 50% of all word-types. However, categorizing a text based on such sparse features can over-fit a system, or the system can treat a certain document as an outlier. Therefore, we apply a pruning procedure to reduce the representation complexity and possibly improve the effectiveness of the attribution scheme. We measure both the vocabulary size and the achieved performance using a subset of the text as its representation. We can observe a notable size reduction when filtering out the n-grams that appear twice or fewer in a document. In all of the collections, the vocabulary size can be reduced by over 80% by culling terms appearing once or twice. As a general trend, we observe that removing very low-frequency word-types might increase the performance. For example, only considering the 500 most frequent words, the $RPrec$ is increased by around 10% in the English collection and even more in the French corpora.

To build our text clustering system with styles, we learned that we do not have to consider the whole vocabulary. Furthermore, we have confirmed that Cosine is a suboptimal distance measure and one based on the L^1 or L^2 norm (e.g., Clark) is more useful.

3.5 Author Clustering Using Spatium

In this paper, we present the author clustering problem and compare it to related authorship attribution questions. The task is when having a set of n documents (or text excerpts) written by several authors, to determine the number k of distinct authors, and to regroup the documents into separate clusters written by the same person. To develop our system for text clustering with styles, we combined the results of our previous research. We base the proposed approach on the feature selection model Spatium that we developed earlier in section 3.2 and use the Canberra distance (weighted version of L^1 norm) which we determined to be one of the optimal measures in section 3.3 for a related task. We present a method estimating the pairwise dissimilarities of documents and use a distance of probability distributions followed by a single link hierarchical clustering.

From PAN CLEF 2016 we have 18 test collections covering two text genres (newspaper articles and reviews) and written in either English, Spanish, or Greek (each genre-language tuple is present three times). Those corpora contain rather short samples, meaning from a bit more than 100 words up to fewer than 1,500 words on average per text. Also, many authors have written just a single text in the collection while a few authors have written up to nine documents, meaning we have to create clusters of any size from containing only one text up to a significant fraction of the available documents. To increase the diversity on the data, we then added two test corpora extracted from literary novels, namely one in English called *Oxquarry1* [30], and the second in French called *Brunet* [29]. Those corpora provide longer texts where we have in mean over 10,000 and 8,000 words respectively. In the *Brunet* corpus, each author is represented with exactly four excerpts extracted from two novels each and in the *Oxquarry1* corpus, we have at least two texts for each writer, meaning there are no clusters of size 1.

Our proposed method first computes the Canberra distance from each text to every other text in a given collection. From the $n \times n$ distance, matrix we have to select the document pairs that have been written by the same person and have to be merged to form a cluster. To do so, we cannot simply define a global threshold because this would vary from one collection to another. We can neither select the lowest x distance values because it is possible that more or fewer than x clusters should exist. Our strategy instead is to look at each row in the $n \times n$ distance matrix, calculate its mean and standard deviation, and highlight a given cell in this row if its value is below the average with δ times the standard deviation taken away. The limit value of δ can be chosen depending on how many and how big we expect the clusters to be, where a lower value tends to create more and larger clusters. Those selected cells show up a distance value to another text which is especially low compared to the distance values from the text in this row to all others in the collection. The same computation is also done using the columns of the $n \times n$ distance matrix to highlight cells that show up a distance value from a text which is especially low compared to the distance values to the text in this column from all others in the collection.

Each highlighted cell serves as a hint for a shared authorship. Since the feature selection with Spatium does not give a symmetric distance matrix, we can get a total of four hints for each text tuple A and B, namely $dist_{row}(A, B)$, $dist_{row}(B, A)$,

$dist_{column}(A, B)$, and $dist_{column}(B, A)$. If two out of those four hints are satisfied, then we consider this to be enough evidence that the same author wrote the two texts and we link the two texts together. We generate the final clusters by grouping all texts using the single link clustering. This approach means that having a connection between A and B, and another one between B and C, the final cluster $\{A, B, C\}$ is formed to have each document in exactly one cluster. This strategy might lead to a chaining of texts, but by using a rather restrictive threshold (high δ value), we do not expect the clusters to grow unusually long.

The performance of our clustering system is evaluated using the *BCubed* F_1 score as presented in Equation 2.11. In the PAN corpora, some documents are wrongly clustered together, which decreases the document precision part of the *BCubed* F_1, but overall, we cluster most documents correctly together (increasing document recall). To put our results in perspective, we compare them with the results of the other participants from PAN CLEF 2016. Our clustering performance is in the second rank out of eight, with the best approach being 0.05% better than our system. However, we noticed that a naïve approach of clustering each document in a single cluster would give only a slightly worse outcome (better than the third best-ranked participant) because of the many authors who only wrote a single text.

This bias is not present for our two literary corpora *Oxquarry1* and *Brunet* where we can significantly outperform such a naïve baseline method. More specifically, when inspecting the *Oxquarry1* collection, our model was able to correctly cluster the 12 texts written by Hardy, seven by Stevenson, six by Morris, six by Orzcy, four by Butler, and three by Chesterton. However, the proposed clustering was not perfect because the system splits the eight excerpts from Conrad into two clusters with four documents apiece, and we can find each of the three texts written by Tressel and Forster in a single cluster. Figure 3.1 presents a visualization of the results, where a green line represents a detected shared authorship between the document pair, a blue dashed line is an undetected link between the two texts, and the purple bubbles represent the final created clusters. We can see that there are many missing links in the biggest cluster with the 12 texts from Hardy, but since we used a single link clustering strategy, they are not needed to group all documents together correctly. For those texts that we cannot link directly, the computed distances between them, even though the same author wrote them, are too high for our δ limit. The *BCubed precision* is 1.0 (100%) in this example, meaning we never cluster documents together that have been drafted by different persons. Because of the incomplete and split clusters as described above, the *BCubed recall* is reduced to (38*1.0 + 8*0.5 + 6*0.333)/52 = 0.8462 according to Equation 2.10.

In the *Brunet* corpus, the overall performance is lower compared to the *Oxquarry1*. Here, we have exactly four texts per author (and 11 distinct writers), but our system was not specifically adjusted to account for the requirement of a minimal or maximal cluster size. Our approach can form a cluster with the four texts written by Voltaire or Proust. The four excerpts of Maupassant and Flaubert were also detected, but Spatium adds a link between the two clusters. Similarly, the four texts of Marivaux form a cluster, but we add a link to a cluster of two works written by Sand. These two-incorrect links are the only false positive ones. We generate six small clusters composed of two texts written by the same author each and leave the last ten excerpts in individual clusters. Figure 3.2 presents a visualization of

the results where, in addition to the previous representation, a red double line represents a pair of texts that was incorrectly assigned the same author. Again, we can see that we do not need to detect a shared authorship for all document pairs, but at least three links for each cluster of four documents should be selected.

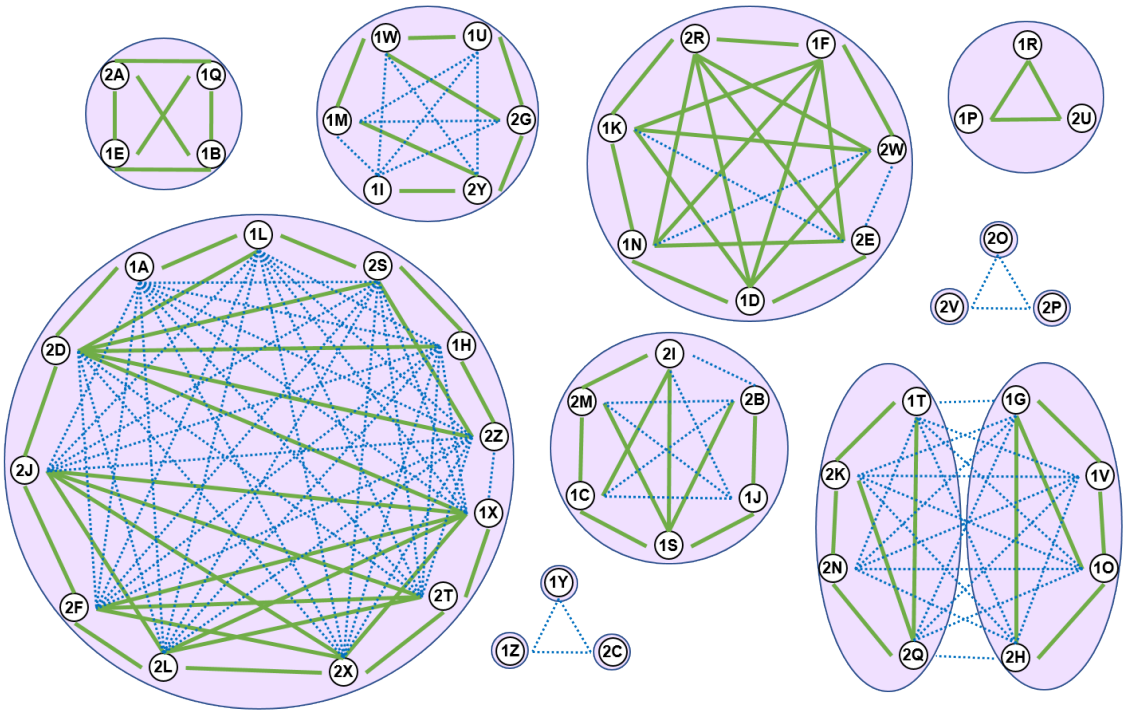


Figure 3.1 – Clustering results in the *Oxquarry1* corpus.

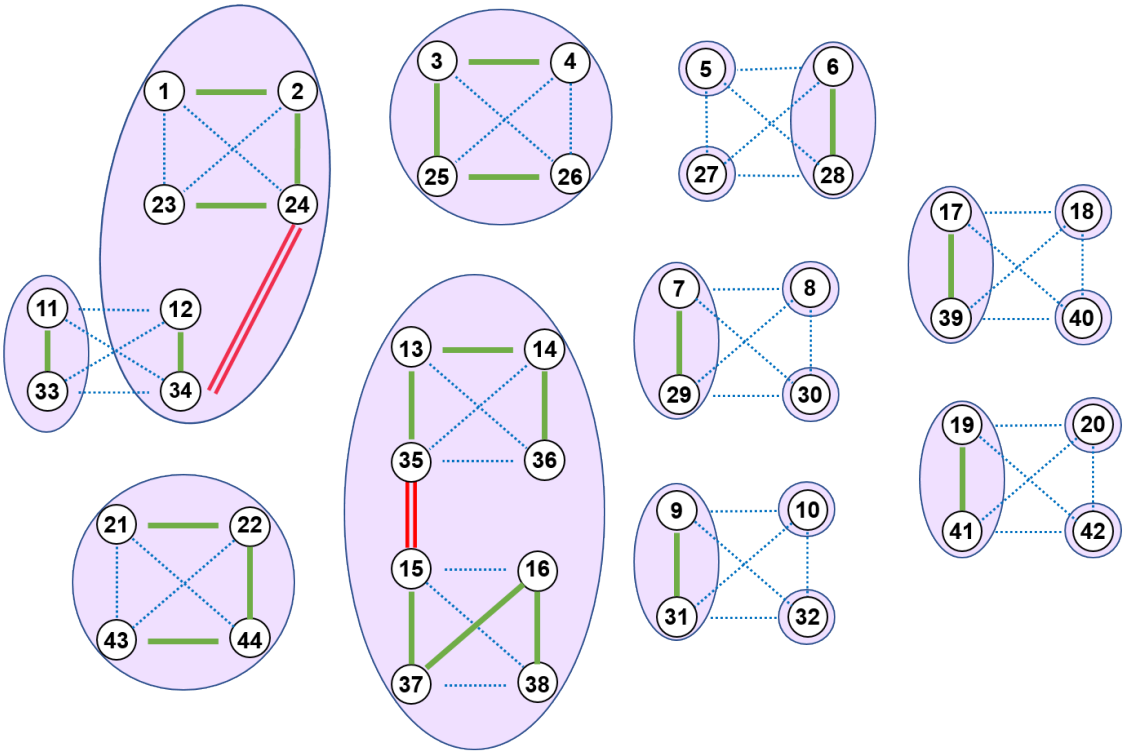


Figure 3.2 – Clustering results in the *Brunet* corpus.

3.6 Author Clustering with an Adaptive Threshold

While in section 3.5 we presented the author clustering problem and our approach to solving it, in the current section, we explain the applied methods, decisions, and reasons for them more deeply. Furthermore, we provide a more detailed analysis showing the strengths of our system, but we also indicate the potential problems and provide reasons for some possible failures of the model and explain how the model can be adapted.

The main challenge we had was that the PAN corpora contain vastly diverse sorts of texts and the clusters are rather small with many authors having written just a single document. When calculating the inter-textual distances, we expected the pairs of texts that are written by the same person to have the smallest distance values. Fig. 1 in section A.6 visualizes our expectations where we see the correct document links in blue mostly in the left part of the histogram having a small distance between them while the incorrect document links (red) are on the right side of the plot with a large distance. What we observed for instance in one of the Dutch corpora containing newspaper articles was a complete interleaving of correct and incorrect links, as visualized analogously in Fig. 2 in section A.6. This observation made us decide that an approach based simply on the overall smallest distance values does not work. A different linking method has to be chosen which is a confirmation of what we found in section 3.2 where we stated that a relative approach might be effective to detect a shared authorship between documents and that focusing on what is most pertinent for the writer of a specific document can capture the most relevant information.

Therefore, we looked at each row and column individually and defined the threshold to be the mean (μ) of the row (or column) minus δ times the standard deviation (σ) of that same row (or column). The result is that the threshold adapts itself dynamically based on what "small" means in the current context for the text in question. If any of the values in that row or column is below the threshold, we indicate a shared authorship. The rationale behind this approach is that when we expect the distances to be normally (Gaussian) distributed, the area under the curve outside of $\mu \pm \delta * \sigma$ corresponds to a small portion. Setting $\delta = 1.96$ means that this area is only 5%, and since only the lower part (the notably short distances) are of interest to us, the fraction in question is 2.5% of all distances. Other common limits are $\delta = 1.64$ and $\delta = 1.28$, that would shrink the possibly relevant data to 5% and 10% of the values. Accordingly, the δ value can be adapted to allow more or fewer links.

To create the ranked list of links, we check for each document pair the number of available indications ν to link them. At most, there can be $\nu = 4$ indications, and a link gets a probability between $(\nu + 1) * 0.2$ and $\nu * 0.2$. This means if for instance $dist_{row}(A, B)$ and $dist_{row}(B, A)$ are satisfied, but the requirements for $dist_{column}(A, B)$ and $dist_{column}(B, A)$ are not fulfilled, then the probability range is between $(2 + 1) * 0.2 = 0.6$ and $2 * 0.2 = 0.4$. To define the position in this range, we sort the links according to the sum of the two distance values (i.e., $dist(A, B) + dist(B, A)$) from the smallest to the largest. This method extends our system

described in section 3.2 to return a degree of belief with the decision to increase the confidence in the system.

To test the stability and sensitivity of our system, we used the bootstrap evaluation approach. Each document vector is therefore randomly changed such that each coefficient gets a value around its expected value. This way, we can test if the system overfits to specific features and if it is sensitive to slight variations. By repeating the probabilistic vector modification multiple times, we can test how stable the system performs. We noticed that the clustering performance is almost unaffected by the bootstrap approach as the mean *BCubed* F_1 is decreased by only 2%. The standard deviation of the average *BCubed* F_1 is slightly more than 1%. Therefore, the system can cluster the documents such that it is not sensitive to variations in the data and can provide stable results. Nonetheless, the MAP shows a relative change of -30% in all collections, and the standard deviation is almost 3%. These results mean the MAP is rather sensitive to the presented corpus and doesn't perform as stable as the *BCubed* F_1.

As presented in the workshop papers from PAN CLEF 2016 and 2017 [23, 26], this approach can not only easily compete with the best systems regarding performance value (*BCubed* F_1 and MAP), but also regarding runtime. There were major differences between the two years. The size per text is substantially smaller (a few paragraphs with around 100 words instead of full documents with around 1,000 words) in the second year compared to the first year. Furthermore, the average size of the clusters was increased, and the number of authors that have written only a single text was decreased. In the first year, we achieved with our system the second best result with an approach that took less than two minutes to finish, while the one better performing system (0.0005 higher *BCubed* F_1 and 0.1149 higher MAP) took over two days to complete. Similarly, in the second year, we had the fastest system out of the best-performing ones with again the second best result (average of *BCubed* F_1 and MAP). These results confirm our initial assumptions from section 3.1 that a simple model can match the performance of much more complex strategies in a variety of situations.

Chapter 4

Conclusion

In this thesis, we have presented our works related to text clustering with styles. We felt that it is a valuable endeavor to explore the full space of authorship analysis techniques and iteratively build a reliable and stable system to approach our goal. With a suitable text collection, we first created a system to attribute an unknown text to one out of a closed group of possible authors (authorship attribution task). Similarly, to profile a sample (e.g., male or female; 10s, 20s, or older), it had to be classified as one out of a limited number of categories depending on the most similar observations or the category with the smallest distance (author profiling). The extension to this is if one person can write the sample from the group of potential authors or another unknown author. For this, we had to verify or controvert a shared authorship for a pair of texts (authorship verification). Afterwards, we linked those pairs of texts where we saw an indication of a shared authorship and had enough evidence that the same person has written them (authorship linking). Finally, after checking every text tuple for possible links, we built the final clusters based on various strategies (author clustering).

We evaluated our methods and systems continuously through the participation in the CLEF evaluation campaigns from 2014 to 2017. Our clustering approach was shown to be effective, and our systems obtained results that were competitive with the top performing systems. Clustering effectiveness was not our only goal and reason for choosing our methods. Rather, we wanted to develop approaches that generally perform well beyond specific test collections.

4.1 Summary of Contributions

In this dissertation, we examined text clustering with styles. Overall, we can make the following conclusions.

We tested multiple inter-textual distance functions in theoretical and empirical tests. In this research, we discovered that some of the measures work well in theory but not always in practice and vice versa. In fact, the Tanimoto and Matusita distances respect all theoretical properties and belong to the best performing measures. The empirical tests indicate that the Canberra and Clark measures are even better suited even though they do not fulfill all of our requirements. Overall, we noted that the Cosine function neither satisfies all conditions nor works notably well.

Furthermore, we designed a measure reflecting the (un)certainty of the proposed decision in authorship verification. By designing the system such that every decision comes along with a confidence of correctness, the user trusts the system more than when given a binary answer only.

Then, we compared different text representation strategies to both determine suitable feature sets as well as keeping them computationally inexpensive. Reducing the size of the text not only decreases the runtime but can also increase performance by ignoring spurious features. Moreover, we created different feature selection procedures. Our model, called SPATIUM, chooses features most relevant to the text in question and can characterize the author adequately.

We also evaluated distributed language representations and compared them to state-of-the-art methods for authorship attribution. While in our work we were mostly focused on the creation of simple methods, investigating more complex schemes led to interesting findings. We showed that not every approach is better in every aspect and that the deep learning methods might be sensitive to parameter settings.

Finally, we created a simple framework to cluster texts based on the writing style. Using a dynamic threshold, we can choose the relative smallest distance values. Our system was applied in various natural languages belonging to a variety of language families and in multiple text genres. With the flexible feature selection and the distance of probability distribution, the system works in any of the tested settings.

The main takeaway message is that a simple approach can lead to solid results in a short time while more complex models do not necessarily improve the result significantly but always take longer to complete. Besides, the results of our simple approaches can be justified in plain English because they are not based on a black box approach but use word frequencies and the differences between them.

4.2 Future Directions

The presented achievements of this thesis motivate additional ideas in the same direction. While we feel that from the standpoint of laboratory testing, the field has reached a high degree of maturity regarding clustering performance, it is evident that much remains to be done concerning the uptake of author clustering technologies in the "real" world.

We could systematically look at authorship analyzing methods under different conditions. Unlike in classical approaches, in which we expect clear values from the input data, we could test how much we can change the ground truth and the available texts (e.g., adding noise) while still getting useful results. Instead of fixing all parameters, some sources of variation in the corpora should be allowed and even enforced. Similarly, we could further investigate the performance degradation when learning on one text genre and applying the learning model on another text genre (e.g., learning on blogs and evaluating on a sequence of tweets). The difference of text size and sample size could be compared to determine if it is better to have many samples per category even if they are short or if it is beneficial to have a few extended

examples. The influence of noise could be evaluated to discover whether we can still learn the profile of a new author even when the training data contains errors. More specifically what degree of noise can be tolerated and still produce results with high reliability. The effect of combining texts from different genres could be examined to decide whether in the absence of enough training data it is helpful to add texts from a different genre. Furthermore, the best textual representation of a class could be extracted to get a deeper understanding of the writing style of a given class. Finally, the applicability of state-of-the-art approaches should be tested in languages from different families that received less attention in research (e.g., Swahili, Chinese). In a long-term sense, we should evaluate the special importance of the author, text genre, topics, and type in the authorship analysis.

The development of authorship analyzing technologies has reached a point at which it can be carefully applied in practice to resolve cases of unknown or disputed authorship [10]. Such systems are used on a regular basis to support a testimony of forensic linguists in court as expert witnesses in cases where the authenticity of a piece of writing is important [48]. Despite their successful application, none of the existing approaches have been shown to work flawlessly, partially because of the overall complexity and also because the problems do not have to be well-posed [42]. All approaches have a likelihood of returning false decisions under certain circumstances, but we barely understand the conditions under which they fail. Therefore, it is particularly interesting to analyze whether and how these conditions can be regulated because any form of control over the outcome of an author identification software bears the risk of misuse [41].

There is a large increased interest in digital humanity studies that work with machine learning approaches on diverse datasets. In computational social science, the dominant data forms are texts originating from the Web. Different new research perspectives are related to the current research such as the discrimination between fake and real news, or the spread of misinformation in social networks [6, 12]. This sort of text tends to contain noise, can be from different genres, and may have an arbitrary length. Therefore, evaluating the robustness and reliability of systems handling such data will be of importance with the ever-growing Web. Fortunately, authorship analyzing remains a very active field, as is attested by the continuous high interest in the PAN CLEF lab.

Appendix A

Papers

A.1 Distributed Language Representation for Authorship Attribution

Mirco Kocher, Jacques Savoy.
In *Digital Scholarship in the Humanities*, to appear.

Distributed language representation for authorship attribution

Mirco Kocher and Jacques Savoy
University of Neuchatel, Switzerland

Abstract

Distributed language representation (deep learning) has been applied successfully in different applications in natural language processing. Using this model, we propose and implement two new authorship attribution classifiers. In this perspective, a vector-space representation can be generated for each author or disputed text according to words and their nearby context. To determine the authorship of a disputed text, the cosine similarity between vector representations can be applied. The proposed strategies can be adapted without any difficulty to different languages (such as English and Italian) or genres (essays, political speeches, and newspaper articles). Evaluations using the *k*-nearest neighbors (*k*-NNs))and based on four test collections (the *Federalist Papers*, the *State of the Union* addresses, the *Glasgow Herald*, and *La Stampa* newspapers) indicate that the distributed language representation preforms well, providing sometimes better effectiveness than state-of-the-art methods such as *k*-NN, nearest shrunken centroids, chi-square, Delta, latent Dirichlet allocation, or multi-layer perceptron classifier.

Correspondence:
Jacques Savoy, University of Neuchatel, Rue Emile-Argand 11, 2000 Neuchatel, Switzerland.
E-mail:
Jacques.Savoy@unine.ch

1 Introduction

Computer-assisted authorship attribution aims to determine, as accurately as possible, the true author of a document or a text excerpt (Love, 2002; Olsson, 2008; Stamatatos, 2009). Under this general definition, the closed-class attribution problem assumes that the real author is one of the specified candidates. In the open-set situation, the real author could be one of the proposed authors or another unknown one. The authorship attribution can however be limited to determine demographic or psychological traits of the author (profiling) (Argamon *et al.*, 2009; Pennebaker, 2011) or simply to determine whether a given author did in fact write a given text (chat, threatening e-mail, doubtful testimony) (verification) (Koppel *et al.*, 2007; Stover *et al.*, 2016).

To solve these attribution questions, it is assumed that every author owns a unique, measurable style that is distinct from that of other writers. Moreover, one can presuppose that such personal style is stable over the author's life, or at least during a few decades (e.g. between 20 and 40 years). To determine such personal stylistic aspects, a sample of texts written by each of the possible writers must be available. Using specific stylistic representations or author profiles, the system can compute a distance (or similarity) measure between the disputed text and each possible author. The author with the smallest distance (or the highest similarity) is considered to be the true author. Instead of having an answer limited to a single name (black box system), an authorship system might return a probability that the proposed author is the real one (Savoy, 2016), as well as some evidence supporting the proposed decision (transparent box model).

As stylistic features able to discriminate between different authors, numerous studies propose

Digital Scholarship in the Humanities © The Author 2017. Published by Oxford University Press on behalf of EADH.

doi:10.1093/llc/fqx046

(Burrows, 2002; Grieve, 2007; Savoy, 2015b) to use the most frequent words or functional words (determiners, pronouns, conjunctions, prepositions, and auxiliary verb forms). Mostly those features are extracted from the text without considering their context. Only a few authorship experiments proposed generating document representations based on words in their context; however, those were mainly limited to bigrams of words used directly (Jockers and Witten, 2010) or employed to generate word networks (Mehri *et al.*, 2012).

In this study, the authorship will be determined grounded on the relationships between words using a distributed language representation (deep learning) (Bengio, 2009; Goldberg, 2017). With the help of the back-propagation algorithm (Rumelhart *et al.*, 1986), and using various neural network architectures, such approaches have shown effective solutions in several tasks such as image categorization (Krizhevsky *et al.*, 2012). They have also proven beneficial in different natural language processing issues such as word prediction (Mikolov *et al.*, 2013a), word similarities (Levy and Goldberg, 2014), named entity recognition, morphological and syntactical relationships (Mikolov *et al.*, 2013b), translation (Sutskever *et al.*, 2014), speech recognition (Mohamed *et al.*, 2012), information retrieval (Vulić and Moens, 2015), or query expansion (Ramrakhiyani *et al.*, 2015), as well as for some text categorization tasks (Goldberg, 2017) such as sentiment analysis or political orientation detection (Iyyer *et al.*, 2014).

The success of this language representation can be related to the distributional hypothesis (Harris, 1954) specifying that words occurring in similar contexts tend to have similar meaning. Such a hypothesis was empirically verified by Miller and Charles (1991), and used to design information retrieval (IR) systems (Blair, 1990), or to determine the meaning of words (Labbé and Labbé, 2005).

In authorship attribution, this contextual representation can detect differences in the context of words such as *love* or *power* in Shakespeare's or in Bacon's works (Michell, 1996; Craig and Kinney, 2009). In political speeches, the term *tax* or the pronoun *we* are usually related to distinct contexts when a speech is written by a Republican as opposed to a Democrat (Lakoff and Wehling, 2012; Sylwester and Purver, 2015). Thus, considering the word context using a distributed language representation provides a new perspective in authorship attribution and can offer high accuracy rates.

The rest of this article is organized as follows. The next section presents the state of the art in authorship attribution, while the third section describes the four test collections used in our experiments. Section 4 exposes the evaluation methodology applied in this study. The different authorship attribution approaches used as baseline are presented in Section 5, while Section 6 explains the two new proposed authorship attribution models. Afterward, Section 7 evaluates them and compares their performances with the selected baselines. A conclusion draws the main findings of this study.

2 Related Work

As strategies to solve the authorship attribution questions, a first set of methods suggests defining an invariant stylistic measure (Holmes, 1998). Such invariant values must reflect the particular style of a given author, on the one hand, and, on the other, they should vary from one person to another. As possible solutions, different lexical richness measures or word distribution indicators have been proposed such as Yule's K measure, statistics related to the type-token ratio (e.g. Herdan's C, Guiraud's R or Honoré's H) (Baayen, 2008), the proportion of word types occurring once or twice (e.g. Sichel's S), as well as the average word length and mean sentence length. None of these measures has proven very satisfactory due in part to word distributions (including word bigrams or trigrams) ruled by a large number of very low probability elements (Large Number of Rare Events) (Baayen, 2008).

As a second framework, a multivariate analysis can be applied to capture each author's discriminative stylistic features. Some of the main approaches applicable here are principal component analysis (Binonga and Smith, 1999; Craig and Kinney, 2009; Holmes and Crofts, 2010), clustering (Labbé,

2007), or discriminant analysis (Jockers and Witten, 2010). As stylistic features, these approaches tend to employ the top 50–200 most frequent word types, as well as some part-of-speech information.

As a third set of techniques, various effective machine-learning classifiers have been proposed, such as *k*-nearest neighbors (*k*-NN), naïve Bayes (Manning *et al.*, 2008), nearest shrunken centroids (NSC) (Tibshirani *et al.*, 2003), decision tree, support vector machine, etc. (Stamatatos, 2009; Jockers and Witten, 2010). These different classification strategies share the following common procedure (Sebastiani, 2002). First, text samples are collected for each possible author. With these samples, a feature selection scheme is applied to choose the most appropriate features able to discriminate between the different writers. Then the classifier learns the discriminative stylistic aspects of each possible author based on text samples. Finally, the disputed text is given to the learning system to determine the most probable author.

As a fourth framework, different distance-based measures have been suggested. Using the differences in word distribution between authors, this paradigm proposes to define a distance between the disputed text and either author profiles or different texts for which the authorship is known. As well-known strategies following this perspective, one can mention Burrows' Delta (2002) or the chi-square method (Grieve, 2007), both using the top m most frequent word types (with $m = 40$–$1{,}000$), the Kullback–Leibler divergence (Zhao and Zobel, 2007) using a predefined set of 363 English words, the use of specific vocabulary (Savoy, 2012), or Labbé's method (2007) using the whole vocabulary. Such distance measures can also be applied with less frequent words. For example, Burrows (2007) proposed two distinct but complementary tests. The first one is grounded on words used regularly by one author but sporadically by the others, while the second is grounded on words used infrequently by one author and ignored by the others.

In the current study, a distributed language representation is proposed to determine the real author of a disputed text. The underlying classifier is grounded on a neural network architecture, an approach that is not fully new in authorship attribution. Tweedie *et al.* (1996) have proposed such a strategy using eleven functional words (e.g. 'an', 'upon', 'can', 'his', 'there', etc.) to solve the twelve *Federalist Papers* problem. Using also on a multi-layer perceptron (MLP) network, Kjell (1994) suggests using the letter pair frequencies as features to solve the authorship problem. Using a deep learning approach (Bengio, 2009; Goldberg, 2017), the current attribution scheme will consider a larger number of words (however, word types having an absolute frequency smaller than five or used by a single author will be ignored).

Finally, the latent Dirichlet allocation (LDA) (Blei *et al.*, 2003; Blei, 2012) approach has some relationship with the proposed strategy. LDA views documents as composed of a mixture of topics (in this case, topics does not mean subjects but probabilistic distributions of word occurrences). From such a generative model, Rosen-Zvi *et al.* (2010) proposed to include the authors' information by the mean of a distribution over the topics. Such a view can be useful to see topic variations for a given author, or relationships between authors based on shared topics. The use of LDA as an authorship attribution scheme was described and evaluated recently (Savoy, 2013a).

3 Test Collections

To evaluate the different authorship attribution models, we used four test collections written in two languages and belonging to different text genres. The first and smallest corpus is the *Federalist Papers* containing eighty-five newspaper articles written in 1787–88 to convince New York citizens to ratify the US constitution (Rossiter, 2003). Appearing under the pseudonym *Publius*, these essays were in fact written by Alexander Hamilton, James Madison, and John Jay. More precisely, five articles have been written by Jay, fourteen by Madison, fifty-one by Hamilton, three are jointly written by Madison and Hamilton, and twelve are disputed papers. Due to its historical and political importance, the authorship of these papers was investigated by several studies (Holmes and Forsyth, 1995; Jockers and Witten, 2010; Mosteller

and Wallace, 1964; Savoy, 2013b; Tweedie *et al.*, 1996). Currently, a large consensus admits that Madison wrote all the twelve disputed papers. Table 1 provides some statistics about this corpus. To obtain a larger number of texts, one can evaluate the assignment approaches with the seventy papers (5 + 14 + 51) for which the authorship is certain. As an additional test, the twelve disputed articles will form an additional small test collection.

Our second evaluation corpus is formed by the 226 *State of the Union* addresses (SOTU) delivered by forty-one US presidents from Washington (1790) to Obama (2016). This address is required by the US Constitution, where it is mentioned that the president must provide information to the Congress about the SOTU and 'measures as he shall judge necessary and expedient'. Such an address provides both an analysis of the current situation, indicates the president's priorities, and presents the legislative agenda for the coming year. In this list, Harrison (1841) and Garfield (1881) do not appear (term in office too short), while Taylor (1849) and Trump (2017) were excluded because they wrote only one speech.

Table A1 depicts the number of addresses per president together with the average length of the speeches. To create this corpus, all addresses from the Web site *www.presidency.ucsb.edu* have been downloaded. The longest speech was written by Taft in 1910 (30,773 tokens), and the shortest by Washington in January 1790 (1,180 tokens). Since these speeches are without copyright and spelling errors, belonging to the same text genre, and are relatively easy to understand, they form an interesting evaluation corpus (Savoy, 2015a).

Table 1. Distribution of *Federalist Papers* by author, number of articles, and the mean length (in number of word tokens) and standard deviation

Author	Number	Length (standard deviation)
John Jay	5	1,692 (333)
James Madison	14	2,781 (481)
Alexander Hamilton	51	2,189 (801)
Disputed papers	12	2,009 (449)

The third test collection is extracted from newspaper articles appearing in 1995 in the *Glasgow Herald*. This corpus is part of the Cross-Lingual Evaluation Forum (CLEF) 2003 test collection (Peters *et al.*, 2004) which is available publicly through the European Language Resources Association (ELRA) Web site. The entire collection is composed of 5,420 articles written in English by twenty different authors. Table A2 depicts some information about the distribution over the twenty authors.

As an authorship test collection, this corpus contains articles written in a similar register, targeting the same audience, during the same period of time (1995), and by authors sharing a common background and culture. Thus, various factors having an impact on the author style are kept constant. This corpus was also already used in a previous evaluation study (Savoy, 2012).

The last test collection is extracted from newspaper articles appearing in 1994 in the *La Stampa* newspaper. It contains 4,346 articles written in the Italian language by twenty columnists. It was also part of the CLEF 2003 test collection (Peters *et al.*, 2004). The distribution of the number of articles per author is given in Table A3. In selecting this corpus, our intention was to verify the quality of the different attribution methods using another language but one having linguistic relationships with English.

Finally, for all experiments, each document is preprocessed to determine the different word tokens. All uppercase letters are transformed to their lowercase equivalents, and only the period and comma are retained as punctuation symbols. All other punctuations are ignored as well as digits. Thus, the string 'Paul's book' is viewed as 'paul s book' and the expression 'IBM-360' as 'ibm'. As an additional preprocessing step, all word types appearing less than five times are removed, largely reducing the vocabulary length. For example, applying this constraint to the SOTU corpus, the vocabulary decreases from 19,577 entries to 8,470 (−56.7%). Such a large reduction will speed up the learning stage of the distributed language representation. Moreover, words used only by a single author have been removed because they can easily be exploited to mask the real author. In the *Federalist Papers*, for example the form 'whilst' is used only by Madison, whereas

Hamilton and Jay prefer the term 'while'. So, impostors could use 'whilst' to easily imitate the authorship of Madison.

4 Evaluation Methodology

A single corpus could erroneously favor one attribution scheme over the others. Therefore, using multiple test collections is the norm in an empirical analysis. This avoids incorrect decisions generated by an unknown and hidden characteristic of a single corpus. To evaluate the attribution effectiveness, the most frequently used measure is the accuracy rate (or the percentage of correct assignments). This value can be computed according to two distinct schemes. As a first method, the micro-averaging principle assumes that one decision corresponds to one vote. When the system can correctly identify, for example the right author for 80 articles of 100 articles, the resulting accuracy rate (micro-average) is $80/100 = 0.8$, or 80%. In authorship attribution studies, this technique is most frequently used to compute a mean performance.

As a second method, the accuracy rate is first computed for each of the c authors (or categories), under the assumption that the same importance can be attached to each writer (or category). In this case, one author corresponds to one vote (macro-average), and thus the overall accuracy rate is the mean of all categories. For example, with $c = 3$ possible authors, and with an accuracy rate of 0.8 for the first author, 0.7 for the second, and 0.6 for the third, then the macro-averaging accuracy rate is $(0.8 + 0.7 + 0.6)/3 = 0.7$, or 70%. When we have the same number of texts for each author, both measures return the same value. As depicted in Table 1 and Tables A1–A3, the different evaluation corpora do not have this characteristic. We prefer applying the micro-averaging principle arguing that authors should count proportionally to their frequency, and the difference in the number of texts between writers is not too large.

To determine statistically whether an attribution method should be viewed as better than another scheme, we apply the test proposed by Eberhardt and Flinger, (1977) in which the null hypothesis H_0 states that both attribution models result in similar performance levels (significance level $\alpha = 5\%$). Finally, in our experiments, the accuracy rate is computed based on the leaving-one-out (for the smallest test collections) or ten-fold cross-validation method (for the two largest test collections) (Witten *et al.*, 2011). Using this evaluation methodology, any instance appearing in the training stage never occurs in the test phase, and vice versa.

5 Baseline Attribution Schemes

To evaluate the effectiveness of a new authorship attribution scheme, the performance of the proposed system must be compared to a baseline. However, a single solution cannot play this role. According to the *no free lunch theorem* (Wolpert, 1996, 2001), averaged over all possible problems, every classification algorithm has a similar accuracy rate when classifying new unseen data. No learning scheme is universally better than all the others. Therefore, this study proposes a set of six distinct authorship attribution algorithms to be used as possible baselines reflecting the state of the art. For each of them, we try to respect the proposed implementation in selecting the stylistic features and in computing the similarity or distance between the different profile or text representations.

To promote an attribution scheme, the choice of the feature selection is the first question to be solved. In this view, one can start by considering the solution proposed by Zhao and Zobel (2007). In this case, each text will be represented by a predefined list of very frequent words. Zhao and Zobel (2007) propose a list containing 363 terms belonging mainly to function words (e.g. 'the', 'of', '*is*', '*has*', '*you*', '*can*'). These items correspond mainly to entries in a stopword list (SL) in an IR system (Manning *et al.*, 2008) regrouping words having no precise or important meaning. For the Italian language, an SL (399 words) provided by a search system achieving high retrieval performance in CLEF evaluation campaigns for that language (Savoy, 2001) has been selected.

After defining this feature list, the relative frequency of each word appearing in this list is

computed for each text or each author profile (generated by concatenating all his/her writings). Each text (or author profile) can thus be viewed as a vector in an m dimensional space where m indicates the number of stylistic features (or word types in the current study) (e.g. $m = 363$ for the English corpora). In this vector-space, each text with known authorship corresponds to a point with a label, while the disputed text is viewed as a point without any label.

To assign an author to a disputed text, an intertextual distance is required to measure the gap between points. Frequently used in IR systems (Manning *et al.*, 2008) as well as in distributed language representation (Mikolov *et al.*, 2013b), the cosine similarity metric is selected. As a possible variant, the L_1 distance can be used which has proven effective in previous studies (Kocher and Savoy, 2017).

To define the most probable author of a disputed text, we can assign the label of the closest or the label of the majority of the k-NNs. With a higher value for the parameter k, the classifier results tend to be more robust (less sensitive to noise or, in our context, to variations in word frequencies in the various texts written by the same author). In case of a tie, the closest point is selected to determine the proposed author.

In our test collections, the *SOTU* corpus contains only a few texts per author. In such cases, it is better to fix a small value for k (e.g. $k = 3$). With the two newspaper corpora (*Glasgow Herald* and *La Stampa*), where each possible author wrote numerous articles, a larger value for k (e.g. $k = 7$–11) makes more sense to prevent attribution to outliers.

To reflect more closely the state of the art in authorship attribution, the NSC method (Tibshirani *et al.*, 2003) has been selected. This strategy can be viewed as a variant of the k-NN method in which the less discriminative features are ignored (small feature weights are shrunken toward zero). In authorship attribution, this classifier tends to offer high accuracy rates and thus can be viewed as a highly competitive model (Jockers and Witten, 2010).

Moreover, to also consider authorship models derived from the distance-based paradigm, two additional approaches will also be evaluated, namely, the Delta model (Burrows, 2002) and the chi-square approach (Grieve, 2007). With these additional approaches, the training data are used to generate an author profile (or centroid) for each possible writer. Then the classifier computes a distance from the disputed text representation to each author profile, returning as probable author the one with the smallest distance.

Finally, as another possible attribution scheme, we have selected an approach based on the LDA approach (Savoy, 2013a). In this framework, documents are viewed as composed of a mixture of *topics*. Of course, a given document may cover only a single topic, but this is more the exception than the norm. Therefore, each *topic* does not correspond to a symbolic subject heading such as 'Politics' or 'Sports', but it is defined as a specific word distribution. To define each author profile, we concatenated all topic distributions representing each text written by the same writer. Using the trained data, the system can infer the topic distributions of a new and unseen document (the disputed text in our case). To define the possible author of this query text, we suggest computing the cosine similarity between the topic distribution of the query text and those corresponding to the author profiles. The highest similarity defines the most probable author of the query text.

6 Distributed Language Representation

The distributed language representation (or deep learning) approach consists of representing each word type as a point in a vector-space R^q, where q indicates the dimensions of the representation space. Such a representation is learned for each word according to a training corpus and an objective function (e.g. maximizing the likelihood of word context occurrences). The implementation can be done on different neural network architectures and effective learning procedures (Bengio, 2009), (Goldberg, 2017).

The main aim of such representation is either to predict the surrounding words of a given one (skip-gram model) or to determine the target word given a context (cbow, continuous bag-of-words). Denoting by w_t the word type in the position t, the skip-gram

model needs as input w_t and returns its most probable nearby context (w_{t-b}, w_{t-b-1}, ... w_{t-1}, w_{t+1}, ..., w_{t+b}). The context is not always fixed and some terms can be skipped (e.g. the model may predict only w_{t-b}, ..., w_{t-b-3}, w_{t+3}, ... w_{t+b}). The word order is however always preserved. The cbow model is used to predict the occurrence of the w_t, given a context w_{t-b}, w_{t-b-1}, ... w_{t-1}, w_{t+1}, ... w_{t+b}. When choosing the cbow model, the training corpus must be larger than for the skip-gram model due to the larger number of possible contexts needed to be learned.

To propose an authorship attribution scheme, the skip-gram model was selected. In this case, the representation is learned to predict a word according to its context, with the following objective (maximizing the likelihood):

$$\text{Max } \Pi_{j\neq t, j=t-b}^{t+b} p_v(w_{sj}|w_{st}) \text{ or } \sum_{j\neq t, j=t-b}^{t+b} \log\, p_v(w_{sj}|w_{st}), \tag{1}$$

where w_{sj} indicates the word type occurring in the sentence s at position j, w_{st} the input word type, b is the skip-gram window size, and $p_v()$ is the neural network classifier (probability) that the word w_{sj} appears in the context of word w_{st}, based on the representation v. The learning process will maximize this probability for all possible words w_{st} (in the context of size b), and all possible sentences s.

Different implementations of this framework are freely available such as `word2vec` (Mikolov *et al.*, 2013a) supported by Google, `GloVe` model (Pennington *et al.*, 2014), or the `gensim` Python library (Rehurek and Sojka, 2010). In these implementations, the parameter b is set by default to 5. Thus, the generated representation is considering the five words before and after the target word w_{st}. Increasing this value tends to produce better results but increases the computational complexity. Moreover, a larger distance between words usually implies that the relationship between them tends to be weaker (Harris, 1954).

Predicting the surrounding words of a given one (skip-gram) is the first step, but we need to adapt this representation to be able to solve a document categorization task (Sebastiani, 2002), namely, authorship attribution. Recently, Le and Mikolov (2014) showed how we can adapt the `word2vec` model to obtain a `doc2vec` model. The proposed model can then be applied to predict the polarity or sentiment in each sentence (Pang and Lee, 2005). To achieve a similar objective, Taddy (2015) suggests a simpler model by modifying the learning stage and adopting a Bayesian classifier model (available with the `gensim` library). In this case, a document $\boldsymbol{d}$ is viewed as an ordered set of sentences $\boldsymbol{d} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_s\}$ where $\mathbf{w}_i$ indicates the ith sentence in the underlying document (a variable shown in bold corresponds to a list or a vector). The learning objective given in Equation 2 is viewed as:

$$\log\, p_v(\boldsymbol{d}) = \sum_s \sum_t \sum_{j\neq t, j=t-b}^{t+b} \log\, p_v(w_{sj}|w_{st}), \tag{2}$$

in which the probability model for a document d is based on the representation v trained to maximize the likelihood (Equation (2)). This estimation is summed over the context, all words, and all sentences belonging to the document.

In a document categorization task, we can assign to each text a class label (or author name) $y \in \{1, \ldots, c\}$ where c indicates the number of possible classes (or authors). After regrouping all documents according to the class label, we can train each class separately and obtain c different distributed language representations. We can then estimate the probability $p_{vy}(d)$ that the document d belongs to class y given the representation v. Using this notation, we can determine the most probable class y for a document d as (Bayes rule):

$$p(y|d) = p_{vy}(d) \cdot \pi_y \Big/ \sum_{i=1}^{c} p_{vi}(d) \cdot \pi_i, \tag{3}$$

where π_y is the prior probability of class y.

Such a language representation requires a large corpus of (high quality) texts, having the same genre, time period, and topics as the target applications. It is known that with time, the spelling, and meaning of words may change over long periods (e.g. more than 50 years) (Rule *et al.*, 2015) as well as the style (Crystal, 2003) (e.g. the mean sentence length tends to decrease with time). The text genre might also have an important impact of the vocabulary choice (Biber and Conrad, 2009).

Moreover, various parameters must be given such as the window size ($b = 5$ by default), the number of

dimensions used in the representation ($q = 100$), the minimum number of occurrences of five (or more) terms to be taken into consideration, the number of iterations in the learning stage (or number of epochs), the learning rate (parameter $\alpha = 0.025$), and other parameters related to the underlying neural network structure. To minimize the training time (number of epochs, number of word tokens, vocabulary length, window size, and size of the hidden layer), efficient learning solutions have been proposed (Mikolov *et al.*, 2013a; Rong, 2016).

With the library `word2vec` (or `gensim`) and selecting the skip-gram model, the output is not the probability for each possible word in the neighbor of the input word (see Equation (1)) but the representation of the input word as a vector (of size q). This vector can then be compared to other word vectors to generate clusters of related terms. This is the fundamental property of the distributed language modeling; words appearing close together are related in meaning. The absolute position is not useful to infer additional information and the distance to the origin is meaningless. As another application, the relationships between words can be explored by applying simple algebraic operations (e.g. the result of vector(France)—vector(Paris) + vector(Italy) is very close to the vector(Rome)) (Mikolov *et al.*, 2013b).

To propose our first attribution model, we represent a document d by a vector composed by m selected words (or stylistic features). Each of them also corresponds to a vector (of size q) and the document d is simply the weighted average of these components as shown in Equation (5).

$$\text{Model1} : d = r_1 \begin{bmatrix} w_{1_1} \\ w_{1_2} \\ \vdots \\ w_{1_q} \end{bmatrix} + r_2 \begin{bmatrix} w_{2_1} \\ w_{2_2} \\ \vdots \\ w_{2_q} \end{bmatrix} + \cdots r_i \begin{bmatrix} w_{i_1} \\ w_{i_2} \\ \vdots \\ w_{i_q} \end{bmatrix} + \ldots r_m \begin{bmatrix} w_{m_1} \\ w_{m_2} \\ \vdots \\ w_{m_q} \end{bmatrix}, \quad (4)$$

where r_i indicates the weight associated with the word type w_i. As a first implementation, each r_i corresponds to the relative frequency of the corresponding w_i in the document d (or author profile).

With this first model, the specification of the m word types included in each document representation must be specified. To define these terms, previous stylistic studies have shown that most frequent words (Savoy, 2015b) or functional words (Zhao and Zobel, 2007) provide effective stylistic features. We will follow this good practice. Within Model 1, the importance of those terms is specified by their respective weight r_i corresponding to the relative frequency. Using the distributional language representation framework, the context of those terms is also considered. In this case, when an author tends to use constructions such as 'of the' or 'we need to do more' more frequently, his/her profile will reflect such lexical phenomena.

Instead of adopting a compositional view of a document proposed by Equation (4), our Model 2 represents each document based on Equation (2) (Taddy, 2015). In this case, the underlying idea is to represent a document with all words and their context. Thus, to be similar, two documents must not only share the same words, but those words must appear in similar contexts (and with similar frequencies).

7 Evaluation

To define our first baseline, the k-NNs attribution procedure has been selected. Table 2 reports the accuracy rate achieved by this approach in the second column. Within each cell, the corresponding value of the parameter k is given. As additional baselines, Table 2 depicts the accuracy rate achieved by the chi-square, the Delta, and the nearest shrunken centroid (NSC) methods (with a shrunken parameter set to 2.0). For these methods, each cell indicates the number of words used to generate each author profile (or centroid). Moreover, Zhao's list containing 344 words or the Italian SL with 399 entries were also used. For the LDA scheme (Savoy, 2013a), the main parameter is the number of topics, specified with the variable t in Table 2.

Table 2. Accuracy evaluation for the four test collections and eight authorship attribution models

Test corpus	*k*-NN	Chi-square	Delta	NSC	LDA	MLP	Model 1	Model 2
Federalist	94.3% *	**82.9%** *	92.9% *	41.4%	71.4%	72.9%	87.1% *	**88.6%**
Papers 70	($k=1$)	($m=200$)	($m=200$)	($m=200$)	($t=3$)	(default)	($k=1$)	($k=1$)
articles	**95.7%**	81.4% *	**94.3%** *	**42.8%**	**98.6%** *	97.1%	**88.6%** *	84.3%
	($k=3$)	($m=300$)	($m=300$)	(Zhao)	($t=6$)	(Web)	($k=3$)	($k=3$)
	95.7%	77.9% *	92.9% *	44.3%	95.7% *	**98.6%**	**88.6%** *	85.7%
	($k=5$)	($m=500$)	($m=500$)	($m=500$)	($t=10$)	($h=1/100$)	($k=5$)	($k=5$)
SOTU 226	**61.6%**	**77.4%**	84.1% *	73.9%	**54.9%**	5.2%	68.1%	**87.8%**
speeches	($k=1$)	($m=200$)	($m=200$)	($m=200$)	($t=41$)	(default)	($k=1$)	($k=1$)
	58.1%	69.5%	86.3% *	61.9%	53.5%	8.7%	**69.0%**	84.7%
	($k=3$)	($m=300$)	($m=300$)	(Zhao)	($t=50$)	(Web)	($k=3$)	($k=3$)
	54.6%	56.7%	**87.6%** *	**86.3%** *	52.2%	**42.8%**	63.8%	84.3%
	($k=5$)	($m=500$)	($m=500$)	($m=500$)	($t=60$)	($h=1/100$)	($k=5$)	($k=5$)
Glasgow	42.3%	60.8%	68.3%	65.5%	49.6%	**69.2%**	65.4%	73.7%
Herald 5,420	($k=1$)	($m=200$)	($m=200$)	($m=200$)	($t=20$)	(default)	($k=1$)	($k=1$)
article	45.2%	69.2%	**72.4%**	69.8%	50.4%	8.0%	68.2%	75.9%
	($k=5$)	($m=300$)	($m=300$)	($m=300$)	($t=30$)	(Web)	($k=5$)	($k=5$)
	46.8%	49.5%	53.0%	58.9%	52.3%	50.0%	**68.4%**	**76.1%**
	($k=9$)	(Zhao)	(Zhao)	(Zhao)	($t=40$)	($h=1/100$)	($k=9$)	($k=9$)
	47.9%	**70.5%**	**72.4%**	**73.2%**	**57.2%**	49.3%	**68.4%**	75.7%
	($k=15$)	($m=500$)	($m=500$)	($m=500$)	($t=50$)	($h=1/100$)	($k=11$)	($k=11$)
La Stampa	44.3%	75.4%	80.0% *	77.2%	54.1%	79.5% *	69.4%	80.7%
4,346 articles	($k=1$)	($m=200$)	($m=200$)	($m=200$)	($t=20$)	(default)	($k=1$)	($k=1$)
	47.0%	71.1%	83.2% *	81.2% *	**60.3%**	28.3%	**72.2%**	82.5%
	($k=5$)	($m=300$)	($m=300$)	($m=300$)	($t=30$)	(Web)	($k=5$)	($k=5$)
	48.1%	46.9%	67.8%	69.6%	54.1%	57.3%	72.1%	82.4%
	($k=9$)	(SL)	(SL)	(SL)	($t=40$)	($h=1/100$)	($k=9$)	($k=9$)
	48.8%	73.4%	**83.9%** *	**84.8%**	55.7%	**57.5%**	71.7%	**82.7%**
	($k=17$)	($m=500$)	($m=500$)	($m=500$)	($t=60$)	($h=1/100$)	($k=13$)	($k=11$)

For each combination of test corpus and authorship attribution model, the best performance is indicated in bold

One possible value for this parameter t is the number of authors (e.g. three for the *Federalist Papers*, forty-one for the SOTU addresses, twenty for both newspaper collections). However, the value for t could be larger reflecting the fact that two or more distributions of words (topics) are needed to thoroughly describe the various styles of a given author. Finally, as features, the list of 273 English function words (determiners, prepositions, conjunctions, pronouns, and auxiliary verb forms) has been used for the *Federalist Papers* and SOTU corpora. For the newspaper collections, the top 500 most frequent words and punctuation symbols were used as features (because they tend to produce higher accuracy rates than the set of functional terms).

Under the label 'MLP', the performance measure was obtained using a neural network similar to those proposed in previous authorship attribution studies (Kjell, 1994; Tweedie *et al.*, 1996). The general package used for our experiments was downloaded from the Web site `scikit-learn.org`. To generate the input layer, the words appearing in Zhao's list (or the Italian SL) have been chosen. The set of possible authors forms the output layer. As parameter, the number and size of the hidden layers must be given, as well as the selected solver (e.g. stochastic gradient descent), the value for the parameter α ($=0.001$, learning rate), and the number of epochs ($=200$, duration of the training phase). We have adopted different parameter settings; under 'default' (one hidden layer, 100 nodes, Adam solver), the values suggested by the implementation are selected, while under 'web' (two hidden layers, with five and two nodes, lbfgs solver), the values are fixed according to

recommendations given on the Web site. For the other parameter settings, each cell indicates the number of hidden layers and their respective size (in number of nodes). For example, the notation 'h = 1/100' indicates the presence of one hidden layer with 100 nodes. Finally, the last two columns reported the effectiveness of the two proposed models derived from distributed language representation.

As shown in Table 2, the performance levels achieved with the *Federalist papers* are usually higher than for the other test collections. For this corpus, the highest accuracy rate is achieved with the LDA and MLP approaches. This can be explained by considering that the decision must be taken over only three possible authors. Moreover, the articles are relatively long, and the training data reflect closely the conditions of the test data (same year, genre, register, topics, and objectives).

With the SOTU collection, the number of possible authors is larger (41) which raises the difficulty of finding the right author. However, a large temporal gap between presidents (e.g. more than one century) implies larger style differences (Crystal, 2003; Biber and Conrad, 2009). This phenomenon tends to slightly facilitate a correct assignment with this corpus. With the two newspaper corpora, the number of authors (20) is smaller than with the SOTU, but the articles are, in mean, shorter than the SOTU addresses.

Overall, for the last two test collections (*Glasgow Herald* and *La Stampa*), the accuracy rate usually tends to be lower. Applying the test proposed by Eberhardt and Flinger (1977), the performance differences are usually statistically significant compared to the Model 2 approach. When the performance difference is not significant, an asterisk (*) is added in the corresponding cell in Table 2.

Moreover, the results depicted in Table 2 indicate that none of the proposed authorship attribution methods dominates all the others. For one collection, one strategy can provide the best performance (e.g. MLP or LDA for the *Federalist papers*, NSC with *La Stampa* corpus, Model 2 for the *Glasgow Herald*). However, for the two more difficult corpora (*Glasgow Herald* and *La Stampa*) having a large number of possible authors for texts published during the same year, Model 2 offers high accuracy rates, regardless of the parameter values, compared to the different baselines that tend to be more sensitive to the choice of the parameter values.

Focusing on the neural network models, the accuracy rate performance achieved by Model 2 is usually better than Model 1 performance. On the other hand, Model 1 tends to offer accuracy rates similar to the baselines, except for the SOTU. For both the *Federalist Papers* and SOTU corpus, a small value for the parameter k (between 1 and 3) allows us to obtain the best performance. Such a result was expected due to the small number of texts written by each author for the first two collections. For the two newspaper collections, a higher k value is needed (between 9 and 17). Finally, the MLP approach does not provide high accuracy rates, and the choice of the parameter values is crucial for an effective classification.

Having a test collection written in the Italian language does not present any particular difficulty, and the overall performance achieved with *La Stampa* newspaper is similar to the effectiveness obtained with the *Glasgow Herald*.

The performance levels reported in Table 2 are achieved using the default parameter setting for both Model 1 and Model 2. When analyzing Model 2 in-depth, we found that the window value (parameter b), set by default at 8, returns very effective accuracy rates. Moreover, changing this value in the range from 4 to 12 does not modify the overall performance (Fig. A1 depicting this performance variation is presented). Second, we have conducted a sensitivity analysis with the parameter q defining the size of the vector representing the document in Model 2. By default, this parameter is fixed at 300 for Model 2. When considering values between 150 and 375, we can observe similar performance levels as depicted in Fig. A2. Representing documents with a smaller number of dimensions tends to reduce the accuracy rate. Third, the learning rate (denoted α) is fixed by default at 0.025. Values between 0.1 and 0.95 tend to produce similar levels of performance as shown in Fig. A3. A value smaller than 0.1 or higher than 0.95 tends to decrease the accuracy rate of the attribution.

Table 3 Accuracy rate evaluation for the twelve disputed articles of the *Federalist Papers*

Test collection	*k*-NN	Chi-square	Delta	NSC	LDA	MLP	Model 1	Model 2
Federalist Papers 12 disputed articles	83.3% * ($k=1$)	**91.7%** * ($m=200$)	91.7% * ($m=200$)	91.7% * ($m=200$)	**91.7%** * ($t=3$)	0.0% (default)	50.0% * ($k=1$)	66.7% ($k=1$)
	91.7% * ($k=3$)	83.3% * (Zhao)	83.3% * (Zhao)	**100%** (Zhao)	**91.7%** * ($t=6$)	33.3% (Web)	41.7% * ($k=3$)	**75.0%** ($k=3$)
	83.3% * ($k=5$)	75.0% * ($m=500$)	**100%** ($m=500$)	91.7% * ($m=500$)	**91.7%** * ($t=10$)	**91.7%** * ($h=1/100$)	**66.7%** * ($k=5$)	66.7% ($k=5$)

For each authorship attribution model, the best performance is indicated in bold

Even if the figures appearing in the Appendix are based on the *Glasgow Herald* corpus and Model 2, using the *La Stampa* newspaper or Model 1, similar overall conclusions can be obtained. The proposed default parameter setting tends to produce high-performance levels. Small variations around these default values do not change the overall system effectiveness. According to our data, we can deduce that some robustness has been achieved by the two proposed distributed models and their underlying learning schemes.

Table 3 depicts the accuracy rate obtained when defining the right author for the twelve disputed article of the *Federalist Papers* (Paper #49 to #58, #62, and #63). A large consensus assumes that all these articles have been written by Madison. When inspecting the results more closely, we observed that Jay was never selected as a possible author by Model 1 or Model 2, confirming the fact that Hamilton and Madison's styles are more closely aligned. For the *k*-NN, chi-square, or Delta methods, the recurrent assignment error is with Paper #56 and #55. This last article was also incorrectly classified by the LDA model as well as with our two models. In addition, Model 2 encounters problems with Paper #54 and #57, while Model 1 performs relatively poorly with this test collection (incorrect classification for Paper #49, #55, #56, #62, and #63).

To have a better understanding of the reasons explaining the difficulty of achieving a correct assignment, some SOTU addresses can be analyzed with greater details. First, for some presidents such as Obama, Clinton, or Kennedy, the right attribution does not present any real difficulty for all attribution models. On the contrary, correctly defining the right author of some speeches could be hard, using only on a textual representation. For example, the 1964 SOTU address was uttered by Johnson, but for all authorship attribution models, the most probable author is Kennedy. However, Kennedy was assassinated 22 November 1963, and the SOTU address was delivered 8 January 1964. Clearly the time gap was too short to have a new team of ghostwriters writing a completely new speech reflecting more closely Johnson's views and rhetoric.

Another difficult attribution is the 2001 SOTU address delivered by G. W. Bush. In this case, attribution systems indicate 'Reagan' or 'Clinton' as the most probable author, reflecting the fact that the new presidency was faced with similar issues and difficulties as previous presidents. As a second explanation, we must recall that this speech was the single one given before the attacks of 11 September 2001. After this tragic event, the Bush's administration focused more on the terrorist questions and homeland security and less on issues presented in the 2001 SOTU speech. For the system, the first speech is therefore distant from the other Bush speeches, and closer to either Clinton's or Reagan's style.

As another example, the first speech uttered by G. H. Bush, 9 February 1989 can be studied. This address was attributed to 'Reagan' by the system. First, this was not really an SOTU address, but this speech was delivered to Congress and gives the objectives for the new administration. Thus, like an SOTU address, this speech is delivered in front of the Senate and the House of Representatives. Both the form and the content correspond clearly to an SOTU address (and listed as it in the Web site *www.presidency.ucsb.edu*). In this

case, we see the influence of the previous administration (leaving 20 January 1989) during the very first months of the new one.

More difficult attribution assignments can be found in the first twelve speeches, eight delivered by Washington (1790–96) and four by Adams (1797–1800). In those cases, the system assigns correctly only two addresses to Washington, and none to Adams. The attribution scheme indicates Jackson (1829–36) as the most probable author, a president sharing similar political views with Adams and Washington, and like the latter he was also an Army General. Moreover, behind Washington one can find different writers such as Hamilton, Madison, or even a joint work of two or more writers. Therefore, Washington's style is not stable or reflecting the style of a single person. Finally, we must mention that the automatic attribution is less reliable when the disputed text is short (Potthast *et al.* 2014). The mean length of these first twelve speeches is 1,891 word-tokens, while the mean over the 226 speeches is 8,727 word-tokens. Clearly, these first addresses are shorter than the mean, and thus more problematic to attribute with a high degree of certainty.

8 Conclusion

Recently, different distributed language representations have demonstrated very effective solutions, particularly when faced with continuous, dense data (e.g. image recognition) or with sequential datasets. In this article, we suggest to apply such a language model to propose two new authorship attribution schemes. In this perspective, the document representation is grounded on a combination of word vectors (Model 1) or directly as a document vector (Model 2). In both cases, the attribution procedure considers not only of isolated words but also their relatively large context (usually five to eight words before and after). This second aspect is a new source of evidence, as previous studies tend to consider only word bigrams or trigrams.

To evaluate both proposed models, four test collections have been used, namely, the *Federalist Papers* (seventy documents, three possible authors, or twelve disputed articles, three authors), the SOTU addresses (226 documents, 41 authors), and two newspaper corpora (*Glasgow Herald*, 5,420 articles, 20 authors; *La Stampa*, 4,346 articles, 20 authors). The first three were written in English, the last one in Italian. To compute the distance between the document representation or author profile, we selected the cosine similarity proposed by previous studies in distributed language representations (Mikolov *et al.*, 2013a) or various IR models (Manning *et al.*, 2008).

In this study, various existing authorship attribution models have been selected to represent the state of the art, namely, *k*-NN, Burrows' Delta (2002), Grieve's chi-square (2007), NSC (Tibshirani *et al.*, 2003), and MLP. As performance measure, the accuracy rate has been computed. This performance measure indicates that Model 2 (document vector) performs better than Model 1 (combination of word vectors), and Model 2 tends to achieve high accuracy rates compared to the selected baselines.

The proposed attribution scheme owns however some problems and drawbacks. As with many other machine learning-based approaches, the system needs training data having similar characteristics to that of the test data (e.g. extracting from the same time period, written in the same text genre and register, and having similar topics). Even if the deep learning model requires the specification of different parameters, the proposed default values tend to produce a high-performance level. Small variations around them do not modify significantly the obtained effectiveness. This robustness around the possible parameter values was not achieved with the MLP, where the achieved accuracy rate clearly depends on the selection of the most appropriate values, especially for the number of hidden layers, their size, and the solver procedure (softmax, stochastic gradient descent).

The final user usually needs some explanations justifying the proposed attribution and some degree of support or belief that the proposed author is the true one. These two aspects are more difficult to specify concretely and could be a subject of future work. Moreover, the evaluation using the accuracy rate or the F_1 value does not allow us to have different costs for the false-positive and false-negative

cases. Returning an incorrect assignment to the final user generates a sentiment of insecurity with respect to the system, causes a lack of confidence, or engenders a percept that the computer is stupid. This phenomenon is relatively unknown in the academic world where the traditional performance measures tend to underestimate the real 'cost' of incorrect classifications. Finally, the proposed attribution scheme must be viewed more as a complementary solution that other authorship attribution models can use as confirmation to achieve a higher degree of confidence about the final attribution.

Funding

This research was supported, in part, by the NSF under Grant #200021_149665/1.

References

Argamon, S., Koppel, M., Pennebaker, J. W. and Schler, J. (2009). Automatically profiling the author of an anonymous text. *Communications of the ACM*, **52**(2): 119–23.

Baayen, H. R. (2008). *Analysis Linguistic Data: A Practical Introduction to Statistics using R*. Cambridge: Cambridge University Press.

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, **2**(1): 1–127.

Biber, C. and Conrad, S. (2009). *Register, Genre, and Style*. Cambridge: Cambridge University Press.

Binonga, J. N. G. and Smith, M. W. (1999). The application of principal component analysis to stylometry. *Literary and Linguistic Computing*, **14**(4): 445–65.

Blair, D. C. (1990). *Language and Representation in Information Retrieval*. Amsterdam: Elsevier.

Blei, D. M. (2012). Probabilistic topic models. *Communication of the ACM*, **55**(4): 77–84.

Blei, D. M., Ng, A. Y. and Jordan, M. I. (2003). Latent Dirichlet allocation. *Machine Learning Research*, **3**(3): 993–1022.

Burrows, J. F. (2007). All the way through: Testing for authorship in different frequency strata. *Literary and Linguistic Computing*, **22**(1): 27–47.

Burrows, J. F. (2002). Delta: A measure of stylistic difference and a guide to likely author-ship. *Literary and Linguistic Computing*, **17**(3): 267–87.

Craig, H. and Kinney, A. F. (2009). *Shakespeare, Computers, and the Mystery of Authorship*. Cambridge: Cambridge University Press.

Crystal, D. (2003). *The Cambridge Encyclopedia of the English Language*. Cambridge: Cambridge University Press.

Eberhardt, K. R. and Flinger, M. A. (1977). A comparison of two tests for equality of two proportions. *The American Statistician*, **31**(4): 151–5.

Goldberg, Y. (2017). *Neural Network Methods for natural Language Processing*. San Rafael: Morgan & Claypool Publishers.

Grieve, J. (2007). Quantitative authorship attribution: An evaluation of techniques. *Literary and Linguistic Computing*, **22**(3): 251–70.

Harris, Z. (1954). Distributional structure. *Word*, **10**(23): 146–62.

Holmes, D. I. (1998). The evolution of stylometry in humanities scholarship. *Literary and Linguistic Computing*, **13**(3): 111–17.

Holmes, D. I. and Crofts, D. W. (2010). The diary of a public man: A case study in traditional and non-traditional authorship attribution. *Literary and Linguistic Computing*, **25**(2): 179–97.

Holmes, D. I. and Forsyth, R. S. (1995). The *Federalist* revisited: New directions in authorship attribution. *Literary and Linguistic Computing*, **10**(2): 111–27.

Iyyer, M., A., Enns, P., Boyd-Graber, J. and Resnik, P. (2014). Political ideology detection using recursive neural networks. In *Proceedings Association for Computational Linguistics*. Stroudsburg, PA, USA.

Jockers, M. L. and Witten, D. M. (2010). A comparative study of machine learning methods for authorship attribution. *Literary and Linguistic Computing*, **25**(2): 215–23.

Kjell, B. (1994). Authorship determination using letter pair frequencies features with neural networks classifiers. *Literary and Linguistic Computing*, **9**(2): 119–124.

Kocher, M. and Savoy, J. (2017). A simple and efficient algorithm for authorship verification. *Journal of the American Society for Information Science and Technology*, **68**(1): 259–69.

Koppel, M., Schler, J. and Bonchek-Dokow, E. (2007). Measuring differentiability: Unmasking pseudonymous

authors. *Journal of Machine Learning Research*, **8**(6): 1261–76.

Krizhevsky, A., Sutskever, I. and Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Proceedings Advanced in Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates, Inc. pp. 1106–14.

Labbé, C. and Labbé, D. (2005). How to measure the meaning of words? Amour in Corneille's work. *Language Resource Evaluation*, **29**(2): 335–51.

Labbé, D. (2007). Experiments on authorship attribution by intertextual distance in English. *Journal of Quantitative Linguistics*, **14**(1): 33–80.

Lakoff, G. and Wehling, E. (2012). *The Little Blue Book: The Essential Guide to Thinking and Talking Democratic*. New York: Free Press.

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. *Proceedings International Conference on Machine Learning*. Stroudsburg, PA, USA: Journal of Machine Learning Research''. Levy and Goldberg.

Levy, O. and Goldberg, Y. (2014). Linguistic regularities in sparse and explicit word representations. In Proceedings Computational Language Learning, pp, 171–80.

Love, H. (2002). *Attributing Authorship: An Introduction*. Cambridge: Cambridge University Press.

Manning, C. D., Raghavan, P. and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge: Cambridge University Press.

Mehri, A., Darooneh, A. H. and Shariati, A. (2012). The complex networks approach for authorship attribution of books. *Physica A*, **239**: 2429–37.

Michell, J. (1996). *Who Wrote Shakespeare*? London: Thames and Hudson.

Mikolov, T., Chen, K., Corrado, G. and Dean, J. (2013a). Efficient estimation of word representations in vector space. *Proceedings of* Workshop at ICLR 2013.

Mikolov, T., Yih, W. - T. and Zweig, G. (2013b). Linguistic regularities in continuous space word representations. *Proceedings of* NAACL HLT 2013, pp. 746–51. Stroudsburg, PA, USA: The Association for Computational Linguistics.

Miller, G. A. and Charles, W. G. (1991). Contextual correlates of semantic similarity. *Language and Cognitive Processes*, **6**(1): 1–28.

Mohamed, A. - R., Dahl, G. and Hinton, G. E. 2012. Acoustic modeling using deep belief networks. *IEEE Transactions on Audio, Speech and Language Processing*, **20**(1): 14–22.

Mosteller F. and Wallace D. L. (1964). *Applied Bayesian and Classical Inference: The Case of the Federalist Papers*. Reading: Addison-Wesley.

Olsson, J. (2008). *Forensic Linguistics*. London: Continuum.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings Association for Computational Linguistics*, pp. 115–24.

Pennebaker, J. W. (2011). *The Secret Life of Pronouns. What our Words Say about us*. New York: Bloomsbury Press.

Pennington, J., Socher, R. and Manning, C. D. (2014). Glove: Global vectors for word representations. In *Proceedings of the Empirical Methods in Natural Language Processing*, pp. 1532–43.

Peters, C., Gonzalo, J., Braschler, M. and Kluck, M. (2004). *Comparative Evaluation of Multilingual Information Access Systems*. LNCS #3237. Heidelberg: Springer.

Potthast, M., Gollub, T., Rangel, F., Rosso, P., Stamatatos, E. and 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., Snaderson, M., Hall, M., Hanbury, A. and Toms, E. (eds), *Information Access Evaluation meets Multilinguality, Multimodality, and Visualization. 5th International Conference of the CLEF Initiative (CLEF 2014)*. Berlin: Springer, pp. 268–99.

Ramrakhiyani, N., Pawar, S. and Palshikar, G. (2015). Word2vec or JoBimText? A comparison for lexical expansion of Hindi words. In ACM- Proceedings of the 7th Forum for Information Retrieval Evaluation. New York, NY, USA: Association for Computing Machinery, pp, 39–42.

Rehurek, R. and Sojka, P. (2010). Software framework for topic modelling with large corpora. In *Proceedings LREC* Valletta, Malta: University of Malta, pp. 45–50.

Rong, X. (2016). Word2vec parameter learning explained. *arXiv.org*, arXiv:1411.2738 [cs.CL], 1–21.

Rosen-Zvi, M., Chemudugunta, C., Griffiths, T., Smyth, P. and Steyvers, M. (2010). Learning author-topic

models from text corpora. *ACM–Transactions on Information Systems*, **28**(1): 1–38.

Rossiter, C. (2003). *The Federalist Papers*. New York: Signet Classic.

Rule, A., Cointet, J. - P. and Bearman, P. S. (2015). Lexical shifts, substantive changes, and continuity in *State of the Union* discourse, 1790-2014. *Proceedings National Academy of Sciences United States of America*, **112**(35), 10837–44.

Rumelhart, D., Hinton, G. and Williams, R. (1986). Learning representations by back-propagating errors. *Nature*, **323**(6088): 533–6.

Savoy, J. (2001). Report on CLEF-2001 Experiments. In Peters, C., Braschler, M., Gonzalo, J., and Kluck, M. (eds), *Cross-Language Information Retrieval and Evaluation*. Berlin: Springer, pp. 27–43. Lectures Notes in Computer Science #2069.

Savoy, J. (2012). Authorship attribution based on specific vocabulary. *ACM – Transactions on Information Systems*, **30**(2): 170–99.

Savoy, J. (2013a). Authorship attribution based on a probabilistic topic model. *Information Processing and Management*, **49**(1): 341–54.

Savoy, J. (2013b). The *Federalist Papers* revisited: A collaborative attribution scheme. *Proceedings of the American Society for Information Science and Technology*, **50**(1): 1–8.

Savoy, J. (2015a). *Authorship Attribution Using Political Speeches*. Recent Contributions to Quantitative Linguistics. Berlin: De Gruyter Mouton, pp. 153–164.

Savoy, J. (2015b). Comparative evaluation of term selection functions for authorship attribution. *Digital Scholarship in the Humanities*, **30**(2): 246–61.

Savoy, J. (2016). Estimating the probability of an authorship attribution. *Journal of the Association for Information Science and Technology*, **67**(6): 1462–72.

Sebastiani, F. (2002). Machine learning in automatic text categorization. *ACM Computing Survey*, **34**(1): 1–27.

Stamatatos, E. (2009). A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, **60**(3): 538–56.

Stover, J. A., Winter, Y., Koppel, M. and Kestemont, M. (2016). Computational authorship verification method attributes a new work to a major 2nd century African verification. *Journal of the American Society for Information Science and Technology*, **67**(1): 239–42.

Sutskever, I., Vinayls, O. and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Proceedings* Advanced in Neural Information Processing Systems, Montreal, pp. 3104–3112.

Sylwester, K. and Purver, M. (2015). Twitter language use to reflects psychological differences between democrats and republicans. *PLoS One*, **10**(9): e0137422. doi:10.1371/journal. pone.0137422.

Taddy, M. (2015). Document classification by inversion of distributed language representations. In *Proceedings Association for Computational Linguistics* Stroudsburg, PA, USA: The Association for Computer Linguistics. pp. 45–9.

Tibshirani R., Hastie T., Narasimhan B. and Chu G. (2003). Class prediction by neareast shrunken centroids, with applications to DNA microarrays. *Statistical Science*, **18**(1): 104–17.

Tweedie, F. J., Singh, S. and Holmes, D. I. (1996). Neural network applications in stylometry: the *Federalist papers*. *Computer and the Humanities*, **30**(1): 1–10.

Vulić, I. and Moens, J.-F. (2015). Monolingual and cross-lingual information retrieval models based on (Bilingual) word embedding. ACM - Special Interest Group on Information Retrieval 2015. New York, NY, USA: Association for Computing Machinery, pp. 363–372.

Witten, I. H., Frank, E. and Hall, M. A. (2011). *Data Mining. Practical Machine Learning Tools and Techniques*. Burlington: Morgan Kaufmann.

Wolpert, D. H. (1996). The lack of a priori distinctions between learning algorithms. *Neural Computation*, **8**: 1341–90.

Wolpert, D. H. (2001). The supervised learning no-free-lunch theorems. In *Proceedings of the 6th Online World Conference on Soft Computing in Industrial Applications*. London: Springer, pp. 25–42.

Zhao, Y. and Zobel, J. (2007). Searching with style: authorship attribution in classic literature. Proceedings of the Thirtieth Australasian Computer Science Conference (ACSC2007), Ballarat, pp. 59–68.

Appendix

Table A1. Distribution of 226 SOTU addresses by presidency, number of speeches, and their average length (in number of word tokens) and standard deviation

Author	Number	Length (standard deviation)	Author	Number	Length (standard deviation)
George Washington	8	2,079 (596)	William McKinley	4	16,648 (3,230)
John Adams	4	1,790 (357)	Theodore Roosevelt	8	19,627 (5,361)
Thomas Jefferson	8	2,589 (377)	William H. Taft	4	17,378 (7,505)
James Madison	8	2,712 (570)	Woodrow Wilson	8	4,342 (1,624)
James Monroe	8	5,291 (1,454)	Warren Harding	2	5,690 (103)
John Quincy Adams	4	7,764 (765)	Calvin Coolidge	6	8,610 (1,557)
Andrew Jackson	8	10,648 (2,682)	Herbert Hoover	4	6,360 (2,731)
Martin van Buren	4	11,330 (1,575)	Franklin D. Roosevelt	12	3921 (,1424)
John Tyler	4	8,496 (489)	Harry S. Truman	7	8,288 (7,549)
James Polk	4	18,010 (2,059)	Dwight D. Eisenhower	9	6,015 (1,229)
Millard Fillmore	3	10,496 (2,051)	John F. Kennedy	3	5,644 (565)
Franklin Pierce	4	10,453 (740)	Lyndon B. Johnson	6	4,820 (1,211)
James Buchanan	4	14,091 (1,446)	Richard Nixon	5	3,943 (1,204)
Abraham Lincoln	4	6,869 (959)	Gerald R. Ford	3	4,566 (350)
Andrew Johnson	4	9,537 (1,729)	Jimmy Carter	3	3,781 (728)
Ulysses S. Grant	8	8,139 (2,334)	Ronald Reagan	7	4,595 (703)
Rutherford B. Hayes	4	8,558 (1,848)	George H.W. Bush	4	4,270 (502)
Chester A. Arthur	4	4,907 (2,353)	William J. Clinton	8	7,341 (780)
Grover Cleveland	4	12,299 (5,554)	George W. Bush	8	4,829 (870)
Benjamin Harrison	4	13,625 (1,727)	Barack Obama	8	6,555 (532)
Grover Cleveland	4	14,574 (1,392)			

Table A2. Distribution of *Glasgow Herald* articles by author, number of articles, and their average length (in number of word tokens) with standard deviation

Name	Number	Length (standard deviation)	Author	Number	Length (standard deviation)
Julie Davidson	58	1,119 (317)	William Russell	292	884 (393)
Derek Douglas	411	694 (318)	Tom Shields	174	873 (136)
John Fowler	31	764 (417)	Christopher Sims	391	428 (227)
Ken Gallacher	409	637 (246)	Graeme Smith	322	465 (270)
Doug Gillon	369	582 (322)	Ken Smith	213	544 (342)
Anne Johnstone	73	1,099 (496)	James Traynor	340	858 (322)
Ian McConnell	375	397 (173)	Stuart Trotter	337	586 (218)
Jack McLean	119	882 (207)	Andrew Wilson	434	416 (187)
Ian Paul	419	738 (328)	Ruth Wishart	73	1,026 (298)
Nicola Reeves	371	478 (199)	Alf Young	209	885 (381)

Table A3. Distribution of *La Stampa* articles by author, number of articles, and their average length (in number of word tokens) and standard deviation

Author	Number	Length (standard deviation)	Author	Number	Length (standard deviation)
Marco Ansaldo	288	684 (182)	Maria Teresa Meli	216	711 (115)
Pierluigi Battista	232	697 (253)	Stefania Miretti	64	641 (152)
Roberto Beccantini	365	638 (161)	Fiamma Nirenstein	53	905 (284)
Gabriele Beccaria	72	575 (189)	Emanuele Novazio	250	624 (221)
Enrico Benedetto	253	589 (211)	Gian Paolo Ormezzano	233	606 (251)
Oreste Del Buono	435	665 (587)	Franco Pantarelli	203	594 (141)
Alessandra Comazzi	224	501 (56)	Paolo Passarini	304	608 (149)
Angelo Conti	199	508 (99)	Valeria Sacchi	204	638 (128)
Fabio Galvano	348	616 (186)	Barbara Spinelli	58	1,189 (273)
Massimo Gramellini	119	772 (241)	Lietta Tornabuoni	226	632 (271)

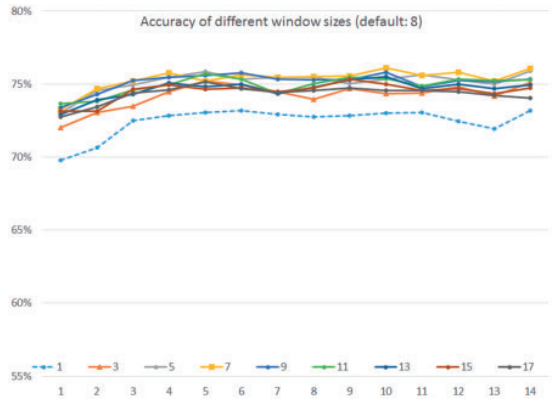


Fig. A1. Accuracy rates obtained with different window size (parameter b) and k values (Model 2, *Glasgow Herald* corpus)

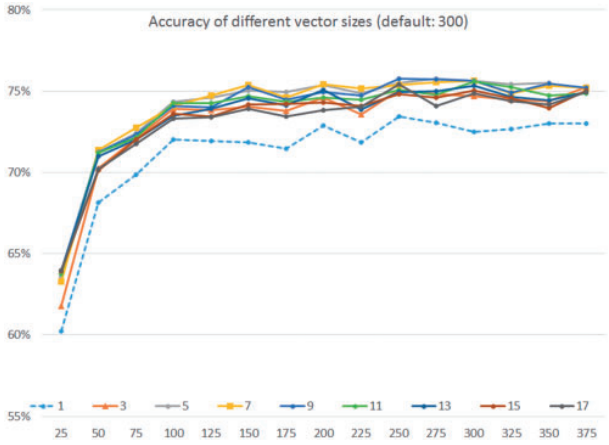


Fig. A2. Accuracy rates obtained with different vector size (parameter q) and k values (Model 2, *Glasgow Herald* corpus)

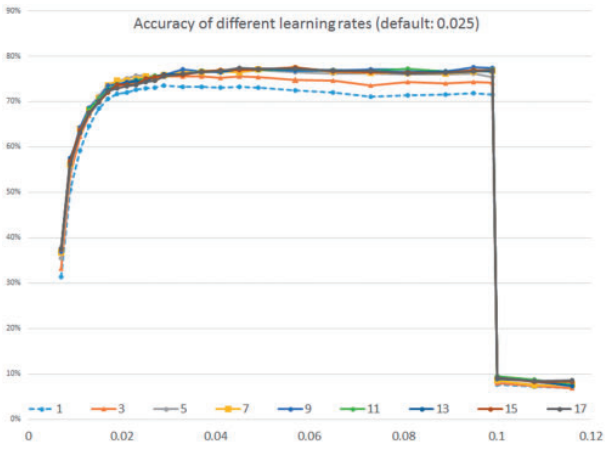


Fig. A3. Accuracy rates obtained with different learning rate (parameter α) and k values (Model 2, *Glasgow Herald* corpus)

A.2 A Simple and Efficient Algorithm for Authorship Verification

Mirco Kocher, Jacques Savoy.
In *Journal of the American Society for Information Science and Technology*, 68(1), 259-269, 2017.

A Simple and Efficient Algorithm for Authorship Verification

Mirco Kocher
University of Neuchatel, rue Emile Argand 11, 2000 Neuchatel, Switzerland. E-mail: Mirco.Kocher@unine.ch

Jacques Savoy
University of Neuchatel, rue Emile Argand 11, 2000 Neuchatel, Switzerland. E-mail: Jacques.Savoy@unine.ch

This paper describes and evaluates an unsupervised and effective authorship verification model called SPATIUM-L1. As features, we suggest using the 200 most frequent terms of the disputed text (isolated words and punctuation symbols). Applying a simple distance measure and a set of impostors, we can 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 or when the attribution scheme is unable to make a decision with a high degree of certainty. Evaluations based on 6 test collections (PAN CLEF 2014 evaluation campaign) indicate that SPATIUM-L1 usually appears in the top 3 best verification systems, and on an aggregate measure, presents the best performance. The suggested strategy can be adapted without any problem to different Indo-European languages (such as English, Dutch, Spanish, and Greek) or genres (essay, novel, review, and newspaper article).

Introduction

Automatic authorship attribution aims to determine, as accurately as possible, the true author of a whole document or a text excerpt (Stamatatos, 2009). To achieve this, a sample of texts written by each of the possible authors is needed. From this common starting point, different contexts can be encountered. In the closed-class attribution problem, the real author is one of several given possible candidates. Within the open-class problem, the real author might be one of the specified writers or another unknown one. In the verification question, the system must be able to determine whether or not a given author did in fact write a given text (e.g., a testimony, a letter, a threatening e-mail, etc.). Finally, authorship attribution can be limited to a profiling view (Pennebaker, 2011), where the system must mine demographic or psychological information about the author (e.g., gender, age, social status, personality traits, etc.).

In this paper we are using some well known historical questions such as "are the *Commentarii de Bello Gallico* (*The Gallic Wars*) really written by Julius Caesar?" or "Which parts of the *Book of the Mormon* are 'translated' by Joseph Smith?" (Jockers, Witten, & Criddle, 2010). With the Internet, the number of anonymous or pseudonymous texts is increasing. Therefore, proposing an effective algorithm for the verification problem represents an indisputable interest. Even though the answer to this verification process can be limited to a binary value (yes/no), a better output is to include a justification supporting the proposed answer. Moreover, an estimated degree of belief (or probability) that the given answer is correct will improve the confidence attached to the system response (Savoy, 2016).

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 (Craig & Kinney, 2009; Thisted & Efron, 1987), 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? Moreover, we might take into account the fact that personal style might evolve during an author's life.

This paper is organized as follows. The next section describes the state of the art in authorship attribution and verification. We then go on to explain our proposed algorithm, called SPATIUM-L1. In the section that follows, we

Received May 1, 2015; revised August 18, 2015; accepted August 19, 2015

 DOI: 10.1002/asi.23648

present our test collections and the evaluation methods used in our experiments. Afterwards, we evaluate the proposed scheme and compare it to the best-performing schemes using six different test collections written in four distinct languages and genres. In the last section, an analysis of the results explains why the proposed algorithm works correctly or sometimes may fail to provide the correct answer. A conclusion summarizes the main findings of this study.

State of the Art

To solve the authorship attribution problem, a first set of approaches is based on unitary invariant values (Holmes, 1998). These invariant measures must reflect the particular style of a given author, but they should vary from one author to another. Following this perspective, we can find the use of lexical richness measures or word distribution factors, including average word length and mean sentence length, as well as Yule's *K* measure and statistics on type-token ratios (e.g., Herdan's *C*, Guiraud's *R*, or Honoré's *H*), and also the proportion of word types occurring once or twice (e.g., Sichel's *S*). None of these measures has proven very satisfactory, due in part to word distributions (including word bigrams or trigrams) dominated by a large number of very low probability elements (Large Number of Rare Events) (Baayen, 2008).

As a second family of approaches, we could apply multivariate analysis to capture each author's discriminative stylistic features. Some of the main approaches applicable here are principal component analysis (PCA) (Binonga & Smith, 1999; Craig & Kinney, 2009; Holmes & Crofts, 2010), cluster analysis (Labbé, 2007), and discriminant analysis (Jockers & Witten, 2010).

As a third set of approaches, various effective machine-learning classifiers have been proposed, such as *k*-nearest neighbors, naïve Bayes, decision tree, support vector machine, etc. (Stamatatos, 2009). Even if various classification strategies have been proposed, the general common procedure is the following (Juola, 2006). First, text samples are collected for each possible author. Based on these samples, a feature selection scheme might be applied to choose the most appropriate features able to discriminate between the possible authors. Then the classifier learns the discriminative stylistic aspects of each possible author based on those text samples. Finally, the disputed text is given to the learning system to determine the most probable author.

As a fourth type of approach, different distance-based measures have been suggested. Based on the differences in word distribution between authors, this strategy proposes to define a distance between the disputed text and either the author profile (concatenation of all texts written by the corresponding person) or the different texts for which the authorship is known. Well-known examples of this include the Burrows's Delta (2002) based on the top *k* most frequent word types (with $k = 40$ to 1,000), the Kullback-Leibler divergence (Zhao & Zobel, 2007) using a predefined set of 363 English word types, and the use of specific vocabulary (Savoy, 2012), or Labbé's method (2007) using the whole vocabulary.

Various modifications of these attribution strategies can be applied in the more specific verification question. First, as for other authorship attribution problems, we need to extract style markers, and different feature sets can be used (e.g., *k* most frequent word types, functional words, frequencies of selected letters or *n*-grams of characters, part-of-speech [POS] *n*-grams, etc.) (Sebastiani, 2002; Juola, 2006; Stamatatos, 2009). The second step is to select a binary classifier able to discriminate between the proposed author (let's say, A) and all others (not-A). During the classification investigation, we can consider the disputed text (denoted Q) as a whole or we can extract from it a sequence of *c* chunks (e.g., each composed of 500 word tokens) and consider the result obtained by these *c* subparts of Q (Koppel, Schler, & Bonchek-Dokow, 2007).

A classical solution is to consider the proposed author A with a set of other possible writers called *impostors* (with a text sample for each of them). We then train a set of binary classifiers to learn models for A versus not-A, B versus not-B, etc. The *c* chunks of the doubtful text are then classified according to our learned models, and, if a preponderance of chunks is classified as A, then we conclude that A is the real author. Otherwise, we can infer that another unknown person wrote the text (Koppel & Winter, 2014). This strategy may fail if we do not consider all writers having a style similar to A. For example, we might have ignored author D depicting a style very similar to A. As soon as a classifier proposes A for a given chunk, we are never sure whether the author is really A or D. When applying such an attribution strategy, it is important to have imposters' texts written in the same period, genre, and on the same topics in order to keep constant other stylistic source variations than the author himself.

Another solution proposed by Koppel et al. (2007) is based on the unmasking technique. For each of the possible authors (let's say we have *m* candidates), we build a learning model with the *k* most frequent word types. We then determine the accuracy of the *m* models. From that point, we iterate a given number of times. After each iteration we remove a few strongly weighted positive and a few strongly weighted negative features. Finally, we plot the degradation of the performance achieved by the *m* models.

Using this approach, the performance graph will depict similar curves for all writers except the real author. To be more precise, when removing features strongly related to the true writer, the performance corresponding to him will clearly drop. Doing the same with another person, who is not the real author, the performance will only slightly decrease because the removed features do not present a strong association between the disputed text and this non-author. Of course, if no clear difference appears, with one author performing clearly worse than the rest, we may conclude that none of the proposed writers is the real one. However, the decision is somehow arbitrary; a decreased performance could be interpreted as marginal or substantial.

DOI: 10.1002/asi

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 (Thisted & Efron, 1987). To evaluate the effectiveness of a verification algorithm, the number of tests should, however, be larger. To create such benchmarks, and to promote studies in this domain, the PAN CLEF 2014 evaluation campaign was launched (Stamatatos et al., 2014). Thirteen research groups with different backgrounds from around the world participated in the PAN CLEF 2014 campaign. Each team has proposed a verification strategy that has been evaluated using the same method.

During the PAN CLEF 2014 campaign, various representations and classifiers were proposed. The best-performing system was based on the impostors' strategy in which each document is represented by numerous *n*-grams of letters and word types, as well as part-of-speech tags, with the number of features ranging from 3,300 to 73,000 (Khonji & Iraqi, 2014). A distance measure is applied to determine whether the query text is written or not by the proposed author. Moreover, to generate more possible impostors, texts have been downloaded from the web. Finally, the processing time of this solution was clearly more expensive (around 21 hours for around 800 verifications) than the others (around 2 hours).

The second-best performance was achieved using a decision tree model (CART algorithm) based on 17 distinct similarity measures, each of them based on numerous features (e.g., character 3-grams weighted by *tf idf*, correlation similarity, bigrams of word types) (Fréry, Largeron, & Juganaru-Mathieu, 2014). The third-best effectiveness was achieved by representing documents by three indexing schemes: all words, LSA (latent semantic indexing) (Deerwester, Dumais, Furnas, Landauer, & Harshman, 1990) using all words, and a combined surrogate based on prefixes, suffixes, *n*-grams (with $n = 1, 2, \ldots, 5$), punctuation symbols, stopwords, vowel combinations, and permutations (Castillo, Cervantes, Vilriño, Pinto, & León, 2014). The similarity between documents is defined as the maximum when considering four different similarity measures (cosine, Jaccard, Euclidian distance, Chebyshev). If the resulting similarity is higher than a given learned threshold, the system assumes that the same author has written the two texts.

As a general trend, we can see that text representation strategies are based on both *n*-grams of letters and other complementary schemes (e.g., POS tags, word types, LSA). The number of features therefore tends to be high, and larger than 1,000. The most effective solutions are based on machine-learning classifiers and the different research groups use distinct learning schemes. During the PAN CLEF 2014 evaluation campaign, the most effective approaches have chosen the impostors' strategy.

Simple Verification Algorithm

To solve the verification problem, we suggest an unsupervised approach based on a simple feature extraction and distance measure called SPATIUM-L1 (a Latin word meaning distance). The selected stylistic features correspond to the top *k* most frequent terms (isolated word types without stemming but with the punctuation symbols). Those terms are selected for the disputed text. For determining the value of *k*, previous studies have shown that a value between 200 and 300 tends to provide the best performance (Burrows, 2002; Savoy, 2015). This reduced number represents a huge difference compared to the 100,000 features used by Koppel and Winter (2014) or compared to the features set size employed in the best systems employed in PAN CLEF 2014. Moreover, the justification of the decision will be simpler to understand because it will be based on word types 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 5) 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:

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

where k indicates the number of term types (word types 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 or in the author profile A, respectively. To estimate these probabilities, we divide the term occurrence frequency (denoted tf_i) by the length in tokens of the corresponding text (n), $\text{Prob}[t_i] = tf_i / n$.

To verify whether the resulting Δ_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. 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 fixing the value $r = 5$. After this last step, we have r values denoted $\Delta_{m1}, \ldots, \Delta_{mr}$, each of them corresponding to the minimum value of a set of m impostors. Instead of working with r values, we compute the mean, denoted Δ_m, of the sample $\Delta_{m1}, \ldots, \Delta_{mr}$.

Finally, the decision rule is based on the value of the ratio Δ_0 / Δ_m as follows:

$$\begin{cases} \text{if } \Delta_0/\Delta_m < 0.975 & \textit{same author} \\ \text{if } \Delta_0/\Delta_m > 1.025 & \textit{different authors} \\ \text{otherwise} & \textit{don't know} \end{cases} \quad (2)$$

Thus, when the Δ_0 value is similar to Δ_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 Δ_0 is

DOI: 10.1002/asi

small compared to Δ_m, the evidence is in favor of assuming that the author of profile A is the real author. Finally, when Δ_m is small compared to Δ_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.

Instead of considering complex text representations, we opt for simpler ones based on the most frequent word types. This strategy has the drawback of ignoring some stylistic features such as POS distribution, complex sentence construction measures, or other type-token ratios. On the other hand, simpler text representation approaches have the advantage of simplicity, have proven to be efficient (Burrows, 2002; Hoover, 2004; Savoy, 2015), and can be understood by the final user. After an attribution has been proposed by the system, the final user may require a justification (e.g., in a court decision). To achieve this, working with frequent words the generation of such an explanation is simpler than having to extract information in a huge space of features (e.g., more than 2,000) or in complex text representation models.

As an attribution method, we propose a simple distance measure (Equation [1]) instead of a complex learning scheme usually based on a "black box" strategy (e.g., neural network, support vector machine [SVM], combination of multiple attribution models). Even if the current computer technology allows us to deploy such complex approaches, the resulting effectiveness depends on large and representative training data sets. Moreover, simpler attribution schemes may provide a high or very high level of effectiveness (Holte, 1993). For example, Hand (2006) shows that for 10 well-known data sets, the difference in performance between the best method and a simple linear approach varies from 15% to 0% (in three cases, the simple linear model produces the best possible answer).

Test Collections and Evaluation Method

During PAN CLEF 2014, six test collections were built, each containing between 100 to 200 problems. In this context, a problem is defined as: *given a set of documents (between one and five) written by the same author, is the new document also written by that author?* In each collection, all the texts matched the same language, genre, and time period. Thus, important factors related to the style are kept constant, and the main remaining stylistic variations can be related to the author. The topics of the text are recognized as having a clear impact on the vocabulary but this factor varies from one document to the other. In fact, it is usually impossible to keep this parameter constant in a test collection.

This test collection includes texts written in four different languages: English, Dutch, Spanish, and Greek. More precisely, we can find two benchmarks for the English and Dutch languages, and only one is written in Spanish or Greek. These last two corpora contain newspaper opinion articles extracted from the newspapers *El Pais* and *To Bhma*. The Dutch collections were written by students, either as an essay or a review. Authors of the English essay corpus were Finnish students having English as their second language. The second English corpus is composed of short novels (horror fiction). In total, we count four different genres in these six benchmarks.

An overview of these test collections is depicted in Table 1 in which the column "Training" indicates the number of problems in the training set. We will ignore the training set in order to be able to compare our results with those of the PAN CLEF 2014 campaign. For the test set, the number of problems is given under the label "# Problems." The mean number of documents for each problem in the test set is indicated in the column "Mean document," and the mean number of word tokens per document under the label "Mean words." For example, with the English novel corpus the style of the proposed author can be analyzed as having, on average, one document containing 6,104 word tokens.

When inspecting the Dutch collections, the number of words available is rather small (mean 116 word tokens for each review, and $2 \times 398 = 796$ mean per essay). When studying the relation between the size of text samples and the accuracy of authorship attribution methods, Eder (2015) found that a minimum length of 5,000 word tokens is required to provide stable results. To obtain reliable attributions, Labbé (2007) suggests working with disputed texts having at least 10,000 word tokens. Therefore, we can expect the mean performance for this language to be lower than that for the other languages. For the Spanish corpus, Table 1 indicates that we have, on average, five documents to learn the stylistic features of the proposed author. A

TABLE 1. PAN CLEF 2014 corpora statistics.

		Training	Test		
Language	Genre	# Problems	# Problems	Mean documents per problem	Mean words per problem
English	essay (EE)	200	200	2.6	833
English	novel (EN)	100	200	1.0	6,104
Dutch	essay (DE)	96	96	2.0	398
Dutch	review (DR)	100	100	1.0	116
Spanish	article (SA)	100	100	5.0	1,537
Greek	article (EA)	100	100	2.7	1,121

DOI: 10.1002/asi

TABLE 2. Evaluation over all six test collections (micro-averaging).

Rank	Run	c@1	Merit-0	Merit-1	Merit-2	Success
1	*Meta-classifier*	**0.713**	**568**	340	112	**0.714**
2	SPATIUM-L1	0.687	535*	**344**	**153**	0.709
3	Fréry et al. (2014)	0.684	540	298	56	0.685
4	Khonji and Iraqi (2014)	0.683	543	291	39	0.683
5	Castillo et al. (2014)	0.676	529*	301	73	0.682
6	Baseline (yes)	0.5	398*	0	–398	0.500

relatively higher performance can be assumed with this benchmark. A similar conclusion can be expected with the English novels collection consisting of longer documents (mean, 6,104 word tokens).

When considering the six benchmarks as a whole, we have 796 problems to solve. When inspecting the distribution of the correct answers, we can find the same number (398) 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 PAN CLEF 2014, a system must return a value between 0.0 and 1.0 for each problem. A value larger than 0.5 indicates that the query text was written by the proposed author and a value lower than 0.5 the opposite. Returning the value 0.5 indicates that the system is unable to make a decision based on the given information. Of course, a value closer to 1.0 (or to 0.0) can be viewed as stronger evidence in favor of (or against) the authorship.

As a performance measure, two evaluation measures were used during the PAN CLEF campaign. The first performance measure is the area under the curve (AUC) of the receiver operating characteristic (ROC) curve (Witten, Frank, & Hall, 2011). 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 the final user.

As another measure, the PAN CLEF campaign adopts the c@1 measure (Peñas & Rodrigo, 2011). This evaluation measure takes into account both the number of correct answers and the number of problems left unsolved in the whole test set. The exact formulation is given in Equation (3), with a minimum value of 0 and an optimum value of 1.

$$c@1 = \frac{1}{np} \cdot \left(nc + \frac{nc}{np} \cdot nu \right) = \frac{nc}{np} \cdot \left(1 + \frac{nu}{np} \right) \quad (3)$$

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 make a definitive decision) (Stamatatos et al., 2014). For example, with $np = 100$ and $nc = 80$ ($nu = 0$), the accuracy rate is $nc/np = 0.8$, and c@1 gives the same value. But when 10 of the "incorrect" decisions are left without an answer ($nu = 10$), the c@1 measure does not view them as wrong, and the c@1 = 0.88.

As additional performance measures that can take account of the answer *don't know*, we can attribute 1 point when the decision is correct, 0 when it is incorrect, and 0.5 when the system decision is *don't know*. To determine the quality of an attribution scheme, we can sum these values (or compute a relative value) to define a merit score. Of course, we can also specify that an incorrect decision must be penalized more strongly and attribute a value of –1 or –2 for such wrong attributions. We will report this performance measure in our evaluations.

Finally, to statistically determine whether or not a given verification strategy would be better than another, we applied the sign test (Conover, 1980). This test is rather conservative and requires strong evidence to detect a statistically significant performance difference. More precisely, when comparing two attribution schemes, the sign test considers only the direction of the difference, denoted by a + or – sign. When the two schemes return the same decision, this observation is ignored. When the decision differs, we assign the sign + if the first scheme returns a better answer than the second one. In the reverse case, this observation receives the negative sign. As the null hypothesis H_0, we assume that both verification schemes produce similar performances. Such a null hypothesis would be accepted if two verification schemes returned statistically similar decisions, otherwise it must be rejected. Thus, when H_0 is true, the number of + must be similar to the number of –. On the other hand, when the number of the two signs diverges, there is a small probability that H_0 is true. In the experiments presented in this paper we limit this probability to 5%. In other words, statistically significant differences are detected by a two-sided sign test (significance level 5%).

Evaluation

Based on the described evaluation method, we achieved the overall results depicted in Table 2 corresponding to the 796 problems present in the six test collections. These means are computed using the micro-averaging principle in which each decision has the same importance. In this table we have reported one performance measure applied during the PAN CLEF campaign, namely, the c@1. These values will be used to rank the different attribution strategies.

DOI: 10.1002/asi

As additional information, Table 2 shows three additional measures. Under the label "Merit-0," we assume that a good answer counts as 1 point, the decision *don't know* 0.5, while an incorrect answer returns 0. As a more complete answer, the attribution system may provide a degree of belief that the proposed attribution is correct (Savoy, 2016). Of course the ultimate goal is to reach a zero-mistake rate. When an error-free system is unlikely, we should penalize the wrong decisions. We clearly prefer a system able to know when "it doesn't know" and provide an answer when the evidence is strong enough to make a decision. Providing wrong answers clearly hurts the credibility of an automatic system. Faced with stupid or incorrect answers, the end user will lose his confidence in the system. Such an attribution scheme cannot be used, for example, to support court decisions.

To reflect this perspective, we attribute −1 point for an incorrect decision under the label "Merit-1," and −2 points under the column "Merit-2." As we can see, SPATIUM-L1 proposes the highest performance with these measures. Finally, the last column "Success" indicates the proportion of correct decisions when ignoring the answers *don't know*.

In Table 2, we have added the system Meta-classifier corresponding to the combination of all 13 systems submitted at the PAN CLEF 2014 evaluation campaign (but without the SPATIUM-L1 system). The underlying decision is based on an aggregation of the answers obtained by the 13 systems. We have also added a baseline corresponding to a system that always produces the answer *yes* (trivial acceptor). For each evaluation measure, the best performance is indicted in bold.

The last line of Table 2 corresponds to the trivial acceptor, and this baseline achieves a value of 0.5 under the performance c@1. The score under the "Merit-0" column is 398 and reflects the fact that this baseline answered correctly 398 problems over 796. With the "Merit-1" measure, the performance drops to zero because the number of correct and incorrect decisions is the same. Using the "Merit-2" measure, the performance is negative (−398) since the weight of an incorrect decision is −2. Ignoring the decisions *don't know*, the proportion of correct answers is 0.5, as indicated in the last column.

When comparing the different strategies using the c@1 values, Table 2 indicates that the performance differences are usually small, except with the trivial acceptor. The Meta-classifier tends, however, to present a slightly better performance (0.713). It is, however, difficult to clearly understand the differences in the system behaviors with this measure. Inspecting the three merit measures, we can see that the SPATIUM-L1 system provides good overall performance. These high values can be explained by the fact that this verification scheme tends to opt more often for a *don't know* answer when the decision is uncertain. Having enough evidence (see Equation [2]), SPATIUM-L1 is then able to propose either a positive or a negative answer.

Using the best performance as a baseline (the first row in Table 2), we compared its effectiveness with other verification models. Statistically significant differences detected by the sign test (two-sided, significance level 5%) are indicated by an asterisk (*) after the corresponding "Merit-0" value. The Meta-classifier tends to propose a statistically better performance than the other attribution schemes, except with Frery's or Khonji and Iraqi's classifier, where the performance difference cannot be viewed as significant.

To have an overview of the individual test collections, we report in the Appendix the performance across the six benchmarks and for the three best verification schemes.

Finally, to gain a better understanding of the choice of the two different parameters within the STATIUM-L1 classifier, we performed various experiments. We can modify the number of rounds r (fixed at 5) and the number k of the most frequent word types (fixed at 200). Varying the value of r from 1 to 7, and the value of k from 40 to 400, the highest c@1 value obtained was 0.691, with a Merit-0 score of 541. From a statistical point of view, this difference is not significant compared to the performance reported in Table 2.

The last possible parameter is the value of 2.5% used in Equation (2) to define when the STATIUM-L1 classifier is able to make a decision with some certainty. Increasing this percentage to 4% or decreasing it to 1.5% does not significantly modify the overall performance. For some languages and genres, such a modification could improve the effectiveness, while for others the same change will hurt the performance. Figure 1 illustrates the performance change in the six corpora when varying the threshold around the proposed 2.5% value. Reducing this threshold to 1% or below tends to force the system to always make a decision without enough evidence. The overall performance (depicted in Figure 1 with the line labeled "Mean") therefore decreases. On the other hand, selecting a value larger than 5% encourages the classifier not to make a decision. Answering more often *don't know* will reduce the performance over a correct decision and the overall performance tends to be clearly reduced.

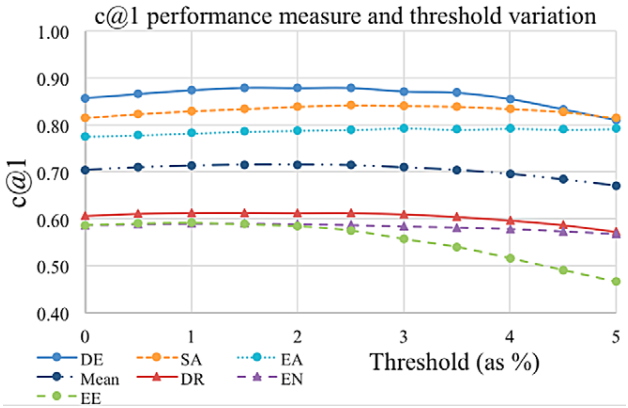


FIG. 1. Relation between the performance and the threshold variation (proposed value 2.5%). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

DOI: 10.1002/asi

TABLE 3. Evaluation over the two English collections (micro-averaging, 400 problems).

Run	c@1	Merit-0	Merit-1	Merit-2	Success
SPATIUM-L1	0.58	233.5	107.5	−18.5	0.61
SPATIUM-L1 (Zhao & Zobel, 2007)	0.50	206	46	−114	0.52
SPATIUM-L1 (Hughes et al., 2012)	0.53	214	57	−100	0.54
Baseline (yes)	0.5	200	0	−200	0.5

As the number of features, we suggest taking the 200 most frequent word types and punctuation marks in the disputed text. Instead of having a list varying from one text to another, we can opt for a fixed prior list of word types. In authorship attribution studies, Zhao and Zobel (2007) propose that such a list contains 363 English word types (composed mainly of function words). Likewise, Hughes, Foti, Krakauer, and Rockmore (2012) suggest a list of 309 English word types. To verify whether those lists may support a better overall performance, Table 3 reports the different performance measures with these two lists compared to the proposed scheme. As we can see, the performance differences are small and statistically not significant. Having a prior list of discriminative word types could simplify the attribution scheme. We need, however, to define such a list for each language used.

Deeper Analysis

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 develop a better explanation of the proposed assignment. To achieve this deeper understanding, we will analyze some problems extracted from the English essays (EE) corpus. Usually, the relative frequency (or probability) differences with very frequent word types such as *when*, *is*, *in*, *that*, *to*, or *it* can explain the decision. In the following discussion, and to simplify the presentation, we only mention the probability of one (the best) of the randomly chosen candidates (instead of considering the $m = 3$ candidates or impostors), and we will evaluate the decision after one iteration (instead of $r = 5$).

As a first correct (true negative) example, we selected Problem #EE002. In this case, the pronoun/determiner *that* has a probability of 0.009 in the query text compared to 0.019 in the proposed author profile and 0.009 in the best candidate. For the auxiliary verb *is*, the probabilities are 0.014 (query), 0.038 (profile), and 0.015 (candidate). The conjunction *and* appears with a relative frequency of 0.021 in the query text, compared to 0.039 (profile), and 0.023 (impostor). As we can see, these three terms tend to indicate that the profile of the proposed author is not the real one, while the best impostor appears more credible. However, not all of the 200 terms follow the same pattern. For example, the auxiliary verb *have* is the most decisive term in favor of the profile, with an estimated probability of 0.016 in the query text, compared to 0.010 (profile), and 0.003 (candidate). Moreover, some of the selected terms are related to the topic discussed in the essay, and thus they don't occur in the profile nor in the impostors. For example, we can encounter the words *listening* and *accent*, both appearing with a probability of 0.004 in the query text but not in the others. The L1-distance between the query text and the best impostor is 0.560 while this distance is 0.663 with the profile of the proposed author. The correct decision taken by SPATIUM-L1 was to answer *different authors* due to the large distance difference.

With Problem #EE224 SPATIUM-L1 also makes the correct decision (*same author*, true positive). When inspecting the determiner *a*, we have very similar relative frequencies in both the proposed author profile (0.021) and in the query text (0.020), but not in the best candidate (0.014). With the preposition *in*, we found a similar pattern (0.016 in query, 0.018 in the profile, and 0.026 in the impostor). The term *to* tends to confirm this finding with very similar relative frequencies in both the profile and in the query text (0.035) justifying the decision *same author*. For some terms, the probability differences are not always as close. In most cases, however, the probability estimate differences between the query text and the candidate are even higher. As an example, we can inspect the preposition *of* having a probability of 0.007 in the query text, 0.016 in the profile, and 0.023 in the candidate. The conjunction *and* follows the same pattern. In this case, the author uses less frequently the word types *and* and *of* in the query text compared to his profile. Some stylistic variations are always possible, as shown in this example. Finally, the L1-distance of the query to the proposed author profile is 0.601, and the one with the best impostor is about 10% larger (0.663). Most of the probabilities estimates are similar, justifying the decision *same author* (with a moderate degree of belief).

As an example of incorrect decisions returned by SPATIUM-L1, we can analyze Problem #EE064 (false negative). In this case, the probability for the article *the* is 0.048 in the query text, 0.062 in the author profile, and 0.047 in the best candidate. The negation *not* reinforces this pattern. The probability estimates are 0.012 in the query text, 0.005 in the profile, and 0.012 in the candidate. The punctuation symbol , (comma) is also clearly against the profile with the probabilities 0.059 (query), 0.074 (profile), and 0.055 (candidate). On the other hand, the punctuation mark (period)

DOI: 10.1002/asi

supports the opposite decision; its probability estimates are 0.036 (query), 0.032 (profile), and 0.051 (candidate). The words *Brutus* and *Cassius* are topical terms appearing frequently in the query text (probability estimates 0.017 and 0.015) but they are absent from the other texts. The L1-distance between the query and the best candidate is 0.485, while the distance to the profile is 0.524. The 8% difference leads to the incorrect decision *different authors* (with, however, a weak support).

With Problem #EE527 SPATIUM-L1 achieves an incorrect decision (false positive), partly because the probability estimate for the term *to* is 0.038 in the query text, 0.035 in the author profile, and 0.022 in the best impostor. With the determiner *the*, the same pattern occurs (0.027 in query, 0.034 in the profile, and 0.051 in the candidate). The pronoun *it* reinforces this finding, with similar frequencies in the query text (0.015), and in the profile (0.018), compared to the best impostor (0.009). The words *unfamiliar* and *subtitles* are topical terms occurring only in the query (0.002) but never in the other texts. The L1-distance between the query and the candidate is 0.479, while the difference with the profile is 0.410, leading to the incorrect decision *same author*. The difference of 17% can be interpreted as a moderate degree of belief supporting this assignment.

Conclusion

This paper proposes a simple, unsupervised technique to solve the authorship verification problem. Unlike many other attribution techniques, the proposed classifier does not require a learning stage to define appropriate values assigned to different parameters. As features to discriminate between the proposed author and different impostors, we propose using the top 200 most frequent terms types (word types and punctuation symbols). This choice was found effective for other related tasks such as authorship attribution (Burrows, 2002). Moreover, compared to various feature selection strategies used in text categorization (Sebastiani, 2002), the most frequent terms tend to select the most discriminative features when applied to stylistic studies (Savoy, 2015). In order to make the attribution decision, we propose using a simple distance measure called SPATIUM-L1 based on the L1 norm.

The proposed unsupervised approach tends to perform very well in four different languages (English, Dutch, Spanish, and Greek) as well as with four genres (essay, novel, review, and newspaper article). Compared to the PAN CLEF 2014 results, the proposed attribution scheme achieved a performance usually among the three best systems within the six different test collections. When computing an overall mean over the six test collections, SPATIUM-L1 shows the best performance level. Thanks to this simple implementation, the proposed scheme can be easily used as a strong baseline to evaluate other verification strategies. Such a classifier strategy can be described as having a high bias but a low variance (Hastie, Tibshirani, & Friedman, 2009). Even if the proposed system cannot capture all possible stylistic features (bias), changing the available data does not modify significantly the overall performance (variance).

Moreover, SPATIUM-L1 returns a numerical value (between 0 and 1) that can be used to determine a degree of certainty (Savoy, 2016). More important, the proposed attribution can be clearly explained because it is based on a reduced set of features, on the one hand, and, on the other, those features are word types 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 combining several similarity measures. The SPATIUM-L1 decision can be explained by large differences in relative frequencies (or probabilities) of frequent words, usually corresponding to functional terms.

To improve the current classifier, we will investigate the effect of other distance measures as well as other feature selection strategies. In this latter case, we want to maintain a reduced number of term types. 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.

Finally, being able to accurately estimate the degree of belief or certainty of a proposed decision is an important aspect, however often neglected in authorship attribution studies. Producing many wrong decisions, especially without warning, will seriously damage the credibility of an attribution scheme. Therefore, each automatic decision should be given with some degree of support reflecting the quality and quantity of evidence in favor of the proposed decision.

Acknowledgments

We would like to thank the reviewers for their helpful suggestions and remarks. This research was supported, in part, by the NSF under Grant #200021_149665/1.

References

Baayen, H.R. (2008). Analysis linguistic data: A practical introduction to statistics using R. Cambridge, UK: Cambridge University Press.

Binonga, J.N.G., & Smith, M.W. (1999). The application of principal component analysis to stylometry. Literary and Linguistic Computing, 14(4), 445–465.

Burrows, J.F. (2002). Delta: A measure of stylistic difference and a guide to likely authorship. Literary and Linguistic Computing, 17(3), 267–287.

Castillo, E., Cervantes, O., Vilriño, D., Pinto, D., & León, S. (2014). Unsupervised method for the authorship identification task. In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.), Proceedings CLEF-2014, Working Notes (pp. 1035–1041). Aachen, Germany: CEUR.

Conover, W.J. (1980). Practical nonparametric statistics. New York: John Wiley & Sons.

Craig, H., & Kinney, A.F. (2009). Shakespeare, computers, and the mystery of authorship. Cambridge, UK: Cambridge University Press.

Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., & Harshman, R. (1990). Indexing by latent semantic indexing. Journal of American Society for Information Science & Technology, 41(6), 391–407.

DOI: 10.1002/asi

Eder, M. (2015). Does size matter? Authorship attribution, small samples, big problem. Digital Scholarship in the Humanities, 30(2), 167–182.
Fréry, J., Largeron, C., & Juganaru-Mathieu, M. (2014). UJM at CLEF in author identification. In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.), Proceedings CLEF-2014, Working Notes (pp. 1042–1048). Aachen, Germany: CEUR.
Hand, D.J. (2006). Classifier technology and the illusion of progress. Statistical Science, 21(1), 1–14.
Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning. data mining, inference, and prediction. New York: Springer.
Holmes, D.I. (1998). The evolution of stylometry in humanities scholarship. Literary and Linguistic Computing, 13(3), 111–117.
Holmes, D.I., & Crofts, D.W. (2010). The diary of a public man: A case study in traditional and non-traditional authorship attribution. Literary and Linguistic Computing, 25(2), 179–197.
Holte, R.C. (1993). Very simple classification rules perform well on most commonly used datasets. Machine Learning, 11(1), 63–90.
Hoover, D.L. (2004). Testing Burrows's Delta. Literary and Linguistic Computing, 19(4), 453–475.
Hughes, J.M., Foti, N.J., Krakauer, D.C., & Rockmore, D.N. (2012). Quantitative patterns of stylistic influence in the evolution of literature. Proceedings of the National Academy of Science 109(20), 7682–7686.
Jockers, M.L., & Witten, D.M. (2010). A comparative study of machine learning methods for authorship attribution. Literary and Linguistic Computing, 25(2), 215–223.
Jockers, M.L., Witten, D.M., & Criddle, C.S. (2010). Reassessing authorship of the *Book of Mormon* using delta and nearest shrunken centroid classification. Literary and Linguistic Computing, 23(4), 465–491.
Juola, P. (2006). Authorship attribution. Foundations and Trends in Information Retrieval, 1(3), 1–104.
Khonji, M., & Iraqi, Y. (2014). A Slightly-modified GI-based Author-verifier with Lots of Features (ASGALF). In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.), Proceedings CLEF-2014, Working Notes (pp. 977–983). Aachen, Germany: CEUR.
Koppel, M., & Winter, Y. (2014). Determining if two documents are by the same author. Journal of American Society for Information Science & Technology, 65(1), 178–187.
Koppel, M., Schler, J., & Bonchek-Dokow, E. (2007). Measuring differentiability: Unmasking pseudonymous authors. Journal of Machine Learning Research, 8(6), 1261–1276.
Labbé, D. (2007). Experiments on authorship attribution by intertextual distance in English. Journal of Quantitative Linguistics, 14(1), 33–80.
Pennebaker, J.W. (2011). The secret life of pronouns. What our words say about us. New York: Bloomsbury Press.
Peñas, A., & Rodrigo, A. (2011). A single measure to assess nonresponse. In *Proceedings 49th ACL*, 1415–1424.
Savoy, J. (2012). Authorship attribution based on specific vocabulary. ACM—Transactions on Information Systems, 30(2), 170–199.
Savoy, J. (2015). Comparative evaluation of term selection functions for authorship attribution. Digital Scholarship in the Humanities, 30(2), 246–261.
Savoy, J. (2016). Estimating the probability of an authorship attribution. Journal of American Society for Information Science & Technology, DOI: 10.1002/asi.23455 in print.
Sebastiani, F. (2002). Machine learning in automatic text categorization. ACM Computing Survey, 34(1), 1–27.
Stamatatos, E. (2009). A survey of modern authorship attribution methods. Journal of the American Society for Information Science and Technology, 60(3), 538–556.
Stamatatos, E., Daelemans, W., Verhoeven, B., Sanchez-Perez, M.A., Stein, B., Juola, P., . . . Barrón-Cadeño, A. (2014). Overview of the author identification task at PAN 2014. In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.), Proceedings CLEF-2014, Working Notes (pp. 877–897). Aachen, Germany: CEUR.
Thisted, R., & Efron, B. (1987). Did Shakespeare write a newly-discovered poem? Biometrika, 74(3), 445–456.
Witten, I.H., Frank, E., & Hall, M.A. (2011). Data mining. Practical machine learning tools and techniques (3rd ed.). Burlington, MA: Morgan Kaufmann.
Zhao, Y., & Zobel, J. (2007). Searching with style: Authorship attribution in classic literature. In Proceedings of the Thirtieth Australasian Computer Science Conference (ACSC2007) (pp. 59–68). Ballarat: CRPIT.

Appendix

To have an overview of the individual test collections, we report in this Appendix the performance across the six benchmarks for the three best verification schemes. For example, Table A.1 reports the performance obtained with the English essays corpus (200 problems), while Table A.3 for the Dutch essays collection (96 problems), and Table A.5 for the Spanish newspaper articles corpus (100 problems). In these tables we used the Meta-classifier performance as a baseline. Statistically significant differences are indicated by an asterisk (*) after the corresponding "Merit-0" score.

The Dutch essay (Table A.3), the Spanish (Table A.5), and the Greek article (Table A.6) are the corpora that return the best overall performances (c@1 or Success). Unlike our expectation, the Dutch essay collection, with its short author profile (mean $2 \times 398 = 796$ word tokens), was not a challenging corpus. The two English collections were more difficult for all attribution schemes. It is difficult to clearly detect general trends. For a given language, the ranking of the systems differs from one genre to the next. The ranking across the genres seems a little bit more stable. For the two article collections (Tables A.5 and A.6), for example, we can find the Statium-L1 or Khonji & Iraqi systems as the best-performing classifiers, followed by Castillo's and Frery's systems. The performance differences, however, are not statistically significant.

TABLE A.1. Evaluation with the English essay (EE) collection (200 problems).

Rank	Run	c@1	Merit-0	Merit-1	Merit-2	Success
1	Frery et al. (2014)	**0.710**	**139.5**	**86.5**	**33.5**	**0.71**
2	*Meta-classifier*	0.680	136	72	8	0.68
3	Khonji and Iraqi (2014)	0.583	116.5*	33.5	–49.5	0.58
4	Castillo et al. (2014)	0.580	116*	32	–52	0.58
5	Spatium-L1	0.577	117.5*	60.5	3.5	0.62
6	Baseline (yes)	0.5	100*	0	–100	0.5

TABLE A.2. Evaluation with the English novel (EN) collection (200 problems).

Rank	Run	c@1	Merit-0	Merit-1	Merit-2	#Success
1	*Meta-classifier*	**0.906**	**87**	**78**	69	0.91
2	Frery et al. (2014)	**0.906**	**87**	**78**	69	0.91
3	SPATIUM-L1	0.899	80.5	75.5	**70.5**	**0.93**
4	Khonji and Iraqi (2014)	0.844	81	66	51	0.84
5	Castillo et al. (2014)	0.861	82	69	56	0.86
6	Baseline (yes)	0.5	48	0	–48	0.5

TABLE A.3. Evaluation with the Dutch essay (DE) corpus (96 problems).

Rank	Run	c@1	Merit-0	Merit-1	Merit-2	Success
1	*Meta-classifier*	**0.645**	**129**	**58**	**–13**	**0.65**
2	Castillo et al. (2014)	0.615	123	46	–31	0.62
3	Khonji and Iraqi (2014)	0.610	122	44	–34	0.61
4	Frery et al. (2014)	0.588	117.5	35.5	–46.5	0.59
5	SPATIUM-L1	0.581	116	47	–22	0.59
6	Baseline (yes)	0.5	100	0	–100	0.5

TABLE A.4. Evaluation with the Dutch review (DR) corpus (100 problems).

Rank	Run	c@1	Merit-0	Merit-1	Merit-2	Success
1	Khonji and Iraqi (2014)	**0.650**	**65**	30	–5	**0.65**
2	SPATIUM-L1	0.621	61.5	**30.5**	**–0.5**	0.64
3	*Meta-classifier*	0.580	58	16	–26	0.58
4	Frery et al. (2014)	0.578	57.5	17.5	–22.5	0.58
5	Baseline (yes)	0.5	50	0	–50	0.5
6	Castillo et al. (2014)	0.370	59	56	53	0.87

TABLE A.5. Evaluation with the Spanish article (SA) collection (100 problems).

Rank	Run	c@1	Merit-0	Merit-1	Merit-2	Success
1	SPATIUM-L1	**0.866**	**83**	**73**	**63**	**0.88**
2	*Meta-classifier*	0.790	82	64	46	0.82
3	Khonji and Iraqi (2014)	0.778	77.5	55.5	33.5	0.78
4	Castillo et al. (2014)	0.760	76	52	28	0.76
5	Frery et al. (2014)	0.750	75	50	25	0.75
6	Baseline (yes)	0.5	50	0	–50	0.5

TABLE A.6. Evaluation with the Greek article (EA) collection (100 problems).

Rank	Run	c@1	Merit-0	Merit-1	Merit-2	Success
1	Khonji and Iraqi (2014)	**0.810**	**81**	**62**	**43**	**0.81**
2	SPATIUM-L1	0.785	76.5	57.5	38.5	0.79
3	*Meta-classifier*	0.760	76	52	28	0.76
4	Castillo et al. (2014)	0.730	73	46	19	0.73
5	Frery et al. (2014)	0.642	63.5	30.5	–2.5	0.65
6	Baseline (yes)	0.5	50	0	–50	0.5

Note. The performances of the SPATIUM-L1 system depicted in the previous tables depend on a random factor, namely, the choice of the impostors. To verify the impact of this selection in the reported performance measures, we show in Table A.7 the c@1 measures based on 500 different choices. In this table, and per test collection, we have indicated the mean, the standard deviation, and the estimated confidence interval covering 95% of the cases. As we can see, the possible variation around the mean performance is relatively small. The reported measures on previous tables are usually closely related to the mean.

DOI: 10.1002/asi

TABLE A.7. Variation around the c@1 performance for the SPATIUM-L1 system.

	c@1		
Test collection	Mean	Standard deviation	Interval (95%)
English Essay (EE)	0.5763	0.0163	[0.5444–0.6083]
English Novel (EN)	0.5889	0.0158	[0.5580–0.6197]
Dutch Essay (DE)	0.8778	0.0143	[0.8498–0.9057]
Dutch Review (DR)	0.6128	0.0198	[0.5739–0.6517]
Spanish Article (SA)	0.8441	0.0196	[0.8057–0.8825]
Greek Article (EA)	0.7917	0.0207	[0.7510–0.8323]

With the English essay corpus, the hardest for our system, SPATIUM-L1 encounters more difficulties. With the Spanish collection (Table A.5), SPATIUM-L1 shows high performance levels. In this case, we have longer texts both in the query (mean 1,537 word tokens) and in the proposed author profile (on average 7,685 word tokens).

DOI: 10.1002/asi

A.3 Distance Measures in Author Profiling

Mirco Kocher, Jacques Savoy.
In *Information Processing and Management*, 53(5), 1103-1119, 2017.

Information Processing and Management 53 (2017) 1103–1119

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Distance measures in author profiling

Mirco Kocher, Jacques Savoy*

University of Neuchatel, rue Emile Argand 11, 2000 Neuchatel, Switzerland

ARTICLE INFO

Article history:
Received 29 November 2016
Revised 4 April 2017
Accepted 19 April 2017

Keywords:
Distance measure
Author profiling
PAN-CLEF
Text categorization

ABSTRACT

Determining some demographics about the author of a document (e.g., gender, age) has attracted many studies during the last decade. To solve this author profiling task, various classification models have been proposed based on stylistic features (e.g., function word frequencies, *n*-gram of letters or words, POS distributions), as well as various vocabulary richness or overall stylistic measures. To determine the targeted category, different distance measures have been suggested without one approach clearly dominating all others. In this paper, 24 distance measures are studied, extracted from five general families of functions. Moreover, six theoretical properties are presented and we show that the Tanimoto or Matusita distance measures respect all proposed properties. To complement this analysis, 13 test collections extracted from the last CLEF evaluation campaigns are employed to evaluate empirically the effectiveness of these distance measures. This test set covers four languages (English, Spanish, Dutch, and Italian), four text genres (blogs, tweets, reviews, and social media) with respect to two genders and between four to five age groups. The empirical evaluations indicate that the Canberra or Clark distance measures tend to produce better effectiveness than the rest, at least in the context of an author profiling task. Moreover, our experiments indicate that having a training set closely related to the test set (e.g., the same collection) has a clear impact on the overall performance. The gender accuracy rate is decreased by 7% (19% for the age) when using the same text genre during the training compared to using the same collection (leaving-one-out methodology). Employing a different text genre in the training and in the test phases tends to hurt the overall performance, showing a decrease of the final accuracy rate of around 11% for the gender classification to 26% for the age.

1. Introduction

In our digital world, author profiling and authorship attribution are viewed as important questions from a security perspective or regarding the increased number of pseudonymous posts and messages (Olsson, 2008). In literary studies, being able to verify the gender of a given character may open new research directions (e.g., is Juliet really a female figure? (Craig & Kinney, 2009)).

To solve these questions, various approaches have been suggested based on vocabulary richness measures (Holmes, 1998), (Baayen, 2008), stylometric similarities (Burrows, 2002; Savoy, 2012), or machine learning models (Stamatatos, 2009; Jockers & Witten, 2010). In many cases, texts are represented by vectors in which the different dimensions correspond to words, characters, *n*-grams of letters or words, part-of-speech (POS) categories, or other possible stylistic measures (e.g., sentence

* Corresponding author.
E-mail addresses: Mirco.Kocher@unine.ch (M. Kocher), Jacques.Savoy@unine.ch (J. Savoy).

http://dx.doi.org/10.1016/j.ipm.2017.04.004
0306-4573/

length, lexical density, etc.). These models assume usually that the corresponding dimensions are orthogonal and the number of dimensions varies widely from one model to another (e.g., 3 in (Fung, 2003), 10 in (Mosteller & Wallace, 1964), 2907 in (Jockers & Witten, 2010), and more than 73,000 in (Khonji & Iraqi, 2014)).

To define the exact demographic category of the author, several proposed approaches need to compute a distance (or similarity) measure between the query text and the representations of the different categories. The shortest distance (or the maximum similarity) determines the predicted class. The choice of the distance measure is often based on *ad hoc* considerations, tradition, or limited empirical evidence.

The objectives of this paper are the following three. First, we want to establish a set of useful properties that a distance measure must respect. Second, and based on a large number of different test collections, we want to determine a reduced set of distance measures showing the most effective performance. Third, using a relatively large number of test collections, we have the opportunity to quantify the influence of the training set on the test set. Thus, we want to estimate the possible performance variations when using the same collection during the training and test phases, when using different collections with the same text genre, or when there are different text genres in both stages.

The rest of this paper is organized as follows. The next section presents the state of the art in author profiling with the focus on the gender and age determination. The third section explains the distance measures and the properties we can expect from an effective one in the context of authorship attribution or profiling. In the fourth section, we perform a theoretical assessment of the different distance measures. The fifth section describes the test collections and the evaluation methodology used in the experiments. The evaluation of the different distance measures is exposed in the sixth section, together with the evaluation of different combinations during the training and test phase. A conclusion draws the main findings of this study.

2. Related work

The main objective of an author profiling task is to determine, as accurately as possible, some author's demographics from text (e.g., gender, age, some personality traits, social class, native language, etc. (Argamon, Koppel, Pennebaker, & Schler, 2009)). The gender distinction might be viewed as the simplest one. The classification decision can be binary and a relatively large amount of data can be collected. However, such a classification system can be effective only if the writing style between genders does differ (Eckert & McConnell-Ginet, 2013) and if such stylistic differences can be detected.

Past studies tend to demonstrate that such differences do occur when considering pervasive and frequent features such as determiners, pronouns, or part-of-speech (POS) distributions. According to Pennebaker (2011), women tend to employ more personal pronouns (especially more *I* and *we*) than men (in relative frequencies, 14.2% vs. 12.7% in blog posts). The signal does not seem to be really strong, but it exists. Looking at other lexical groups, Pennebaker (2011) indicates that men tend to employ more big words (composed of six letters or more), determiners, prepositions, nouns, numbers, and swear words. On the other hand, women use more verbs, negations (e.g., never, not), cognitive words (e.g., consider, explain, think), social words (e.g., family, folks), emotion words (e.g., fears, crying, losses) (Talbot, 2010; Rangel & Rosso, 2016), and certainty words (e.g., always, must). Of course, each individual can depict a more or less strong masculine or feminine figure.

As another way to detect the author gender, Alowibdi, Buy, and Yu (2013) suggest taking account of the first names and user names both transformed into phonemes (with the set of possible phonemes limited to 40). With other languages than English, the gender detection can be determined by considering a few words (e.g., in Portuguese, thank you is *obrigado* for a man, and *obrigada* for a woman) (Ciot, Sonderegger, & Ruths, 2013).

For most of those features, simple lists of words can be created mainly because some grammatical categories such as determiners or pronouns form a closed set. Within a given language, a new preposition cannot be created. For other POS such as nouns or verbs, new instances can occur (e.g., to google). Their identification requires however a language-dependant POS tagger. As an alternative, LIWC (Linguistic Inquiry and Word Count) (Tausczik & Pennebaker, 2010) proposes a set of word lists to measure some stylistic features (e.g., determiners, personal pronouns, modal verb forms) as well as other semantic-based categories such as positive emotions or social words.

A simple count based on a single feature cannot provide a reliable measure. The text register has an impact on those predictors, as for example, pronouns are in general less frequent in a formal context. On the other hand, political speeches delivered by US presidents contain more pronouns, even when the context is official (Savoy, 2016). Therefore, generalization based on a single experiment or using a unique text register should be viewed with caution.

As expected, some topical words are used more frequently by one of the genders (e.g., sports, job, money vs. family, shopping, friends) (Schler, Koppel, Argamon, & Pennebaker, 2006). The two genders have their preferred subjects and this aspect is reflected in their lexical choice. Based on around 100,000 blog posts (50% were written by men, 50% by women), the computer can correctly classify 72% of them based on very frequent words (Argamon, Dhawle, Koppel, & Pennebaker, 2005; Argamon et al., 2009) (19,320 authors; mean text length: 7250 words). Including also the topical terms, the machine can reach an accuracy rate of 76%. In this case, men use more terms related to technology (e.g., game, software, Linux) while women prefer writing about friends and social relations (e.g., love, cute, mom). Those examples are however related to the weblog in which other lexical features can be used to discriminate between the two genders (e.g., emoticons (Crystal, 2006)). Changing the text source requires that the most discriminative topical words between the two genders should be redefined (e.g., selecting the 1000 words depicting the highest information gain ratio (Argamon et al., 2009)).

At the syntactical level, differences between genders can be found (Yule, 2010). For example, women tend to use higher-prestige constructions (*I saw it* vs. *I seen it*) more frequently. On the other hand, double negatives (e.g., *I don't want none*) is a structure occurring more with men than women who have a higher sensitivity to linguistics norms (Coates & Pichler, 2011). In dialogue, men are more likely to interrupt women than the opposite (Talbot, 2010).

In the author profiling task at PAN (Uncovering Plagiarism, Authorship, and Social Software Misuse) in CLEF (Conference and Labs of the Evaluation Forum) 2014, there are distance based approaches, but some solutions used distance measures in implicit form. For instance, the best approaches to solve the task were based on machine learning approaches such as SVM (Support Vector Machines) (López-Monroy, Montes-Y-Gómez, Jair-Escalante, & Villaseñor-Pineda, 2014), logistic regression (Maharjan et al., 2014), or a mix of different classifiers (Weren, Moreira, & de Oliveira, 2014). Similarly, in 2015, the best approaches (Álvarez-Carmona, López-Monroy, Montes-Y-Gómez, Villaseñor-Pineda, & Jair-Escalante, 2015; González-Gallardo, Montes, Sierra, Núñez, Adolfo, & Ek, 2015; Grivas et al., 2015) were based on SVM models. Finally, for the cross-genre classification task in 2016, both SVM (Busger Op Vollenbroek et al., 2016) and logistic regression approaches (Modaresi, Liebeck, & Conrad, 2016; Bilan et al., 2016) have demonstrated the highest performances by correctly determining the gender in 3 out of 4 texts on average. The gender detection remains a hard task (Nguyen et al., 2014).

As a second important profiling variable, the author's age was also analyzed by different studies. In this case, the age year cannot be defined precisely and the challenge is usually to predict an age range. Moreover, the target value is the chronological age range and not the psychological one. It is known that differences may occur between them (Yule, 2010). To avoid problems in the limits between two groups, test collections tend to ignore intermediate age groups. For example, the final categories correspond to teenagers ([13–17]), twenties ([23–27]), and thirties and more ([33–47]) (Argamon et al., 2009).

Limited to those three classes, Argamon et al. (2009) achieve an overall accuracy of 66.9% with stylistic features alone, 75.5% when using topical terms alone, and 77.7% when considering both sets of predictors. While prepositions and determiners can characterize the two older classes, the youngest is more associated with contractions (*im, dont, cant*), and as content words with *haha, wanna*, or *school*. Pennebaker (2011) mentions that younger people tend to use more past tense forms while older persons prefer using the future tense. To discriminate between different age ranges, we can consider the average sentence length or the mean word length. Younger people tend to write shorter sentences and use less complex words. This last aspect can be evaluated by considering the mean number of letters per word, with a small value serving as an indicator in favor of a young author.

For Rosenthal and McKeown (2011), combining both internet writing characteristics and lexical features tends to improve the overall performance for age determination. For example, the number of emoticons (e.g., ;-)) decreases with the age as well as the frequency of internet acronyms (e.g., LOL), or slang expressions (e.g., wazzup). The number of URL or links fluctuates with the author age and thus cannot be used as a pertinent feature. Based on Facebook messages, Sap et al. (2014) suggest to build a lexicon of words with their weights reflecting their usage across age and gender. Applying the generated lexicon on other sources (e.g., blogs, tweets) tends to decrease the overall performance of the prediction, indicating that there are stylistic or content differences between the different sources.

In these previous studies, the main focus is set on determining the most effective features while the choice of the distance (or similarity) measure is usually marginal. In information retrieval (IR) (Manning, Raghavan, & Schütze, 2008; Zhai & Massung, 2016), the relative effectiveness of different similarity measures has been the subject of various studies and evaluation campaigns. As for example, Zobel and Moffat (1998) indicate that the overall effectiveness of a similarity measure depends on the corpus used in the evaluation, the performance measures, and the query type. Thus, a single measure does not always perform better than the others in all contexts. Moreover, implementation details may play a significant role, such as the base for the logarithm, adding one in a formula for smoothing purposes, etc. Gronenschild, Habets, Jacobs, Mengelers, van Os, and Marcelis (2012) and Collberg and Proebsting (2016) made similar findings in evaluating the output of a given system on various platforms. Those results are not directly applicable in our context, in part because in IR the query size is rather short (e.g., composed in mean of one to two words in web search (Manning et al., 2008)) compared to the document length.

3. Distance measures

To build a text classifier, the most effective features are selected and a distance or machine learning approach is applied to determine the author's demographic category. In this paper the most frequent words have been selected as features and various distance measures can then be applied. To be able to discriminate between them, this section presents some useful properties and explains the usefulness of some families of distance measures.

3.1. Distance properties

In the context of authorship attribution or author profiling, the definition of an effective distance measure should not be based on a simple *ad hoc* consideration. A set of properties have to be clearly defined first. To achieve this, in our notation, uppercase letters will denote vectors (or points) while lowercase letters with a subscript indicate the value inside a vector. Thus, A, B, or C specify vectors, a_i indicates the element in the ith position of vector A, and m is the length of the vector.

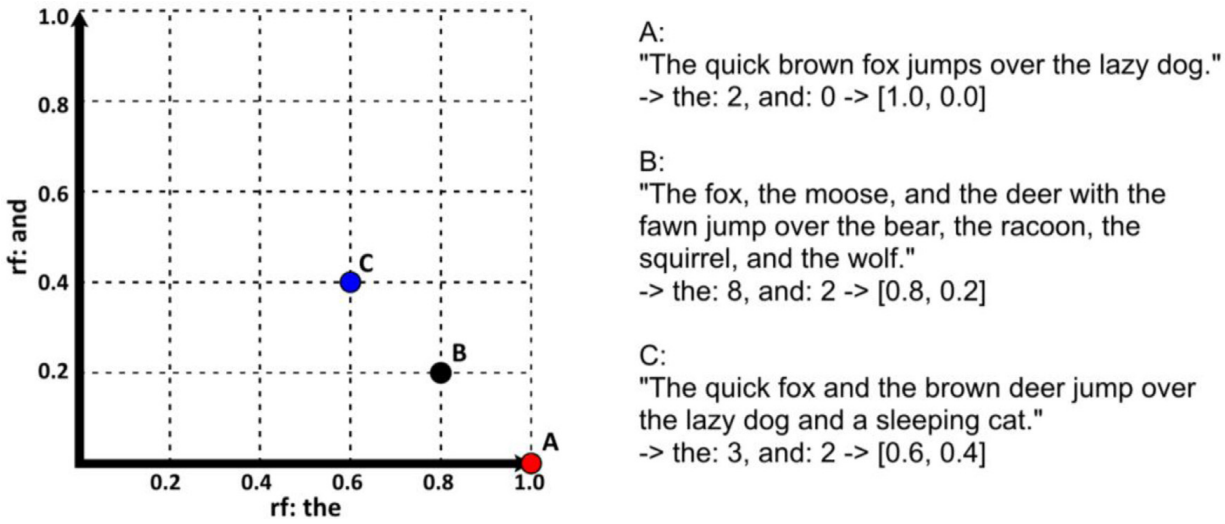


Fig. 1. Example of three points and the distance between them.

First, a distance measure must be equal to zero when computing the distance from a point to itself. Nothing is more similar to a vector than the vector itself. Second, all distance measures between two points must be greater than or equal to zero. Negative values that can be useful in some cases, do not have generally a clear semantics in our context. Thus, for now, we will assume that all a_i values are non-negative ($\forall i: a_i \geq 0$). Moreover, the distance is zero only when computing the distance from a point to itself. Otherwise, the distance between two distinct points must be greater than zero such that we can detect a difference. Third, it is usually convenient to admit that the distance between two points is symmetric as the ordering of the texts is flexible. Going from A to B corresponds to the same distance as going from B to A. This property is not always satisfied in practice (e.g., the presence of one-way streets). Moreover, in our context, one vector may reflect an author profile or a gender category and thus may correspond to a larger text than the second vector (the query text). Thus, for some measure definitions, the symmetry property is not respected without affecting largely the effectiveness of the distance function. More formally these first three properties can be specified as follows.

P1: Property 1. *Zero distance*
When two vectors are identical, their distance must be zero. $dist(A, A) = 0$.

P2: Property 2. *Positivity*
When two vectors differ, the distance between them must be positive.
$dist(A, B) > 0$.

P3: Property 3. *Symmetry*
Computing the distance from vector A to B must return the same value as computing the distance from B to A. $dist(A, B) = dist(B, A)$.
Furthermore, a distance measure can only be a metric if it respects the previous three properties plus the triangle inequality. This last property specifies that adding a point between two points cannot decrease the distance between the first two points. For instance, when measuring the style difference in the first half of a text and the second half separately, then the sum of those two can not be smaller than when directly measuring the overall change in style. This property is usually respected by the majority of the measures used in practice.

P4: Property 4. *Triangle inequality*
For any triangle, the sum of the distances of any two sides must be greater than or equal to the distance of the remaining side.
$dist(A, C) \leq dist(A, B) + dist(B, C)$.
The next two properties are more specific to our context in which each feature included in a vector usually corresponds to a stylistic marker. The fifth property emphasizes the fact that the absence of a feature used frequently in one vector must have a bigger impact on the distance than the absence of an infrequent element. This property underlines the fact that not all dimensions have the same importance, and frequent features should have a larger influence in the distance measure. This helps to reduce the influence of noise or outliers which could obscure meaningful information.

P5: Property 5. *Frequent feature*
The absence of a frequent feature must be penalized more than the absence of an infrequent one.
Finally, we consider the case when the distance between two pairs of points is equal according to the previous properties. In this situation, the last property indicates that the distance including the presence of a feature must be smaller than when this feature is absent. For example, in Fig. 1 we show three short sample texts and represent them in a vector space that has the relative frequencies of the two words “the” and “and” as its dimensions. The point B

is located in equidistance between points A and C. As shown in the figure, $dist(A,B) = dist(B,C)$. To respect this last criterion, when comparing two cases returning the same distance, the preference has to be given to the vector pair depicting the presence of more features. In our example, the distance between B and C should be viewed as smaller than the distance between B and A since A is missing one information that was important to B.

P6. *Presence of the feature*

When the distance measure returns the same value, the presence of a feature is better than the absence of it.

3.2. L^p family

The distance measures can be regrouped under different families (Cha, 2007; Duda, Hart, & Stork, 2001; Manning et al., 2008) where the most frequent one is the L^p family (or L^p norm). In this paradigm, when changing the value of the parameter p, several distance measures can be defined.

Fixing $p = 1$, the Manhattan distance is obtained as defined in Eq. (1). The underlying assumption is that the distance is computed according to the sum of the absolute differences for all dimensions. When the number of dimensions $m = 2$, this L^1 metric corresponds to the city block distance in New York. Similarly, the Gower distance is the L^1 distance divided by the vector length m as shown in the right part of Eq. (1). With this formulation, the distance value can be decomposed into contributions made by each dimension (or feature). When only the rank of the different distances is required, both the Manhattan and Gower measure return the same ordering.

$$dist_{Manhattan}(A,B) = \sum_{i=1}^{m} |a_i - b_i| \;\propto\; dist_{Gower}(A,B) = \frac{1}{m} \sum_{i=1}^{m} |a_i - b_i| \tag{1}$$

Changing the value of p to 2, the Euclidean (L^2 norm) distance is obtained as depicted in Eq. (2). This metric corresponds to our physical concept of a distance between two points, which is the direct straight line. Ignoring the square root can shorten the computation time without changing the ordering of the distances.

$$dist_{Euclidean}(A,B) = \sqrt{\sum_{i=1}^{m} |a_i - b_i|^2} \;\propto\; \sum_{i=1}^{m} |a_i - b_i|^2 \tag{2}$$

The parameter p can take other values and this parameter can be included in the definition of the distance. This general case is known as L^p norm or Minkowski distance as depicted in Eq. (3).

$$dist_{Minkowski}(A,B,p) = \sqrt[p]{\sum_{i=1}^{m} |a_i - b_i|^p} \tag{3}$$

When p goes to infinity, the L^∞ norm or Chebyshev distance is obtained as depicted in Eq. (4). This formulation is also known as maximum metric, or minimax approximation.

$$dist_{Chebyshev}(A,B) = \sqrt[\infty]{\sum_{i=1}^{m} |a_i - b_i|^\infty} = max(|a_1 - b_1|, \ldots, |a_m - b_m|) \tag{4}$$

Some studies proposed to compute the mean between the L^1 and L^∞ distance measure to take account of the advantages of both functions. Eq. (5) shows the corresponding formulation called "Average" distance.

$$dist_{Average}(A,B) = \frac{1}{2}\left(\sum_{i=1}^{m} |a_i - b_i| + max(|a_1 - b_1|, \ldots, |a_m - b_m|)\right) \tag{5}$$

Before comparing these distance measures according to their respective properties and effectiveness (see below), the Appendix visualizes with colors how the distance decreases when the second point is moving away from a given fixed point. For example, we can see that the Euclidian distance describes circles (isodistances) around a given fixed point while the Manhattan approach is based on squares.

3.3. Variants of the L^1 family

Based on the L^1 norm (absolute difference), several variants of the Manhattan distance measure have been proposed. In fact, the value returned by the Manhattan distance is not normalized and it is sometimes difficult to figure out if a given distance is small or large. To propose a solution, the Sørensen, also called Czekanowski or Bray–Curtis distance (Eq. (6)), suggests to normalize the classical Manhattan distance by the sum of all components. As we assume that all vector values are non-negative, the Sørensen distance returns a value between 0 and 1, allowing a clearer interpretation of the distance value than the Manhattan one.

$$dist_{Sørensen}(A,B) = \frac{\sum_{i=1}^{m} |a_i - b_i|}{\sum_{i=1}^{m} (a_i + b_i)} \tag{6}$$

The Tanimoto distance (Eq. (7)), also called Soergel, and similarly the Kulczynski distance (Eq. (8)) also correspond to the Manhattan distance with a normalization factor (divided by the max (or min) of the coefficients). The Motyka distance (Eq. (9)) is based on the maximum value instead of the difference, and the normalization is the sum of all coefficient pairs.

The Canberra distance (Eq. (10)) suggests that the absolute differences of the individual terms are normalized based on the sum of them. One drawback of this last definition is its sensitivity to small changes near zero. The Lorentzian distance (Eq. (11)) is based on the natural logarithm while the Wave-Hedges distance (Eq. (12)) normalizes the difference of each pair of coefficients with its maximum.

$$dist_{\text{Tanimoto}}(A,B) = \frac{\sum_{i=1}^{m} |a_i - b_i|}{\sum_{i=1}^{m} max(a_i, b_i)} \tag{7}$$

$$dist_{Kulcynski}(A,B) = \frac{\sum_{i=1}^{m} |a_i - b_i|}{\sum_{i=1}^{m} min(a_i, b_i)} \tag{8}$$

$$dist_{Motyka}(A,B) = \frac{\sum_{i=1}^{m} max(a_i, b_i)}{\sum_{i=1}^{m} (a_i + b_i)} \tag{9}$$

$$dist_{Canberra}(A,B) = \sum_{i=1}^{m} \frac{|a_i - b_i|}{a_i + b_i} \tag{10}$$

$$dist_{Lorentzian}(A,B) = \sum_{i=1}^{m} ln(1 + |a_i - b_i|) \tag{11}$$

$$dist_{Wave-Hedges}(A,B) = \sum_{i=1}^{m} \frac{|a_i - b_i|}{max(a_i, b_i)} \tag{12}$$

3.4. Variants of the L^2 family

Based on the Euclidian distance or L^2 norm, different variants have been suggested. First, we have the Matusita distance (which imposes the presence of non-negative values for all vector elements). As other variations we have the squared χ^2 and the Clark measure. Those distance measures are variants of the squared difference as used in L^2 and they all result in almost the same visualization (see Appendix).

$$dist_{Matusita}(A,B) = \sqrt{\sum_{i=1}^{m} \left(\sqrt{a_i} - \sqrt{b_i}\right)^2} \tag{13}$$

$$dist_{Squared\ \chi^2}(A,B) = \sum_{i=1}^{m} \frac{(a_i - b_i)^2}{a_i + b_i} \tag{14}$$

$$dist_{Clark}(A,B) = \sqrt{\sum_{i=1}^{m} \left(\frac{|a_i - b_i|}{a_i + b_i}\right)^2} \tag{15}$$

3.5. Inner product family

As another well-known family, different variants based on the inner product (or dot product, see Eq. (16)) have been suggested. The main drawback of the inner product is the absence of a normalization. It is not clear when a distance value should be interpreted as large or small. Therefore, different variants have been proposed, and the most popular is certainly the Cosine similarity (Eq. (17)) which can be transformed into a distance value between 0 and 1 (Eq. (18)) (Zobel & Moffat, 1998; Manning et al., 2008). According to this measure, two similar points indicate similar direction.

$$dist_{Inner\ Product}(A,B) = \sum_{i=1}^{m} a_i b_i \tag{16}$$

$$sim_{Cosine}(A,B) = \frac{\sum_{i=1}^{m} a_i b_i}{\sqrt{\sum_{i=1}^{m} a_i^2} \sqrt{\sum_{i=1}^{m} b_i^2}} \tag{17}$$

$$dist_{Cosine}(A,B) = \frac{1}{\pi} cos^{-1}(sim_{Cosine}(A,B)) \tag{18}$$

As other possible variants, the Jaccard (Eq. (19)) and Dice (Eq. (20)) distances are based on two different normalization approaches. Therefore, similar points have to be in the same direction but also located closely. The Appendix shows clearly the difference between the Cosine and the Jaccard distance.

$$dist_{Jaccard}(A,B) = 1 - \frac{\sum_{i=1}^{m} a_i b_i}{\sum_{i=1}^{m} a_i^2 + \sum_{i=1}^{m} b_i^2 - \sum_{i=1}^{m} a_i b_i} \tag{19}$$

$$dist_{Dice}(A,B) = 1 - \frac{2\sum_{i=1}^{m} a_i b_i}{\sum_{i=1}^{m} a_i^2 + \sum_{i=1}^{m} b_i^2} \tag{20}$$

When comparing the retrieval effectiveness of these four measures in the IR domain, Zobel and Moffat (1998) indicate that the Cosine distance usually tends to produce the best performance. This conclusion cannot be however confirmed in a clear and systematic way.

3.6. Entropy family

Shannon's concept of entropy (Manning et al., 2008) is also the main source of a family of distance measures. The Kullback–Leibler divergence (KLD), also known as relative entropy or information deviation, computes the difference between two probability distributions (see Eq. (21)). In this case, it is required that all values a_i of each vector are non-negative and that their sum is equal to 1. Moreover, the basis of the logarithm is fixed to two in Shannon's entropy measure. However, in the author profiling context, or when only the ranking of the different categories is relevant, changing the basis of the logarithm doesn't affect the ordering of the answers. As for other distance measures, a larger value indicates a larger distance between the two vectors (or points).

$$dist_{KLD}(A,B) = \sum_{i=1}^{m} a_i \; log\left(\frac{a_i}{b_i}\right) \tag{21}$$

The term *divergence* emphasizes the fact that this distance measure is not symmetric. As a variant of KLD, we can mention the Jeffrey or JDivergence defined by Eq. (22) while the KDivergence is depicted in Eq. (23).

$$dist_{JDivergence}(A,B) = \sum_{i=1}^{m} (a_i - b_i) \; log\left(\frac{a_i}{b_i}\right) \tag{22}$$

$$dist_{KDivergence}(A,B) = \sum_{i=1}^{m} a_i \; log\left(\frac{2\; a_i}{a_i + \; b_i}\right) \tag{23}$$

To obtain symmetric measure, one solution is to add to the distance from A to B the distance from B to A. Based on this technique, the KDivergence is used to define the Topsoe distance as shown in Eq. (24). When dividing the Topsoe distance by 2, we obtain the Jensen-Shannon divergence (which is also symmetric). Finally, the Jensen difference is shown in Eq. (25) representing a more complex formulation.

$$dist_{Topsoe}(A,B) = \sum_{i=1}^{m} \left(a_i \; log\left(\frac{2\; a_i}{a_i + b_i}\right) + \; b_i \; log\left(\frac{2\; a_i}{a_i + \; b_i}\right)\right) \tag{24}$$

$$dist_{Jensen}(A,B) = \sum_{i=1}^{m} \left(\frac{a_i \log a_i + \; b_i \log b_i}{2} - \frac{a_i + b_i}{2} log\left(\frac{a_i + b_i}{2}\right)\right) \tag{25}$$

3.7. Combination family

To define a more appropriate distance, different propositions suggest to combine two or more sources of distance measures. For example, Taneja proposes to take account of the arithmetic mean and the geometric mean divergence to define the distance measure given in Eq. (26).

$$dist_{Taneja}(A,B) = \sum_{i=1}^{m} \frac{a_i + b_i}{2} ln\left(\frac{a_i + b_i}{2\sqrt{a_i b_i}}\right) \tag{26}$$

In a related vein, the Kumar-Johnson distance is based on the symmetric χ^2, and both the arithmetic and geometric divergence as shown in Eq. (27).

$$dist_{Kumar-Johnson}(A,B) = \sum_{i=1}^{m} \frac{\left(a_i^2 - b_i^2\right)^2}{2(a_i b_i)^{3/2}} \tag{27}$$

4. Theoretical assessment

The previous section shows that numerous distance measures can be derived and regrouped under five large families. In this section, we verify whether those distance measures respect the six defined properties. Table 1 describes the results where the 24 distance measures are listed with an indication specifying whether or not they obey to the corresponding property. In the last column, we indicate the number of properties respected by the given measure. Overall, only the Tanimoto and Matusita distance measures fulfill all theoretical properties.

Table 1
Summary of the evaluation of the theoretical properties.

Measure	Eq.	P1	P2	P3	P4	P5	P6	Total
Manhattan	1	Yes	Yes	Yes	Yes	Yes	**No**	5
Euclidean	2	Yes	Yes	Yes	Yes	Yes	**No**	5
Chebyshev	4	Yes	Yes	Yes	Yes	Yes	**No**	5
Average	5	Yes	Yes	Yes	Yes	Yes	**No**	5
Sørensen	6	Yes	Yes	Yes	**No**	Yes	Yes	5
Tanimoto	7	Yes	Yes	Yes	Yes	Yes	Yes	6
Kulczynski	8	Yes	Yes	Yes	**No**	Yes	Yes	5
Motyka	9	**No**	Yes	Yes	Yes	Yes	Yes	5
Canberra	10	Yes	Yes	Yes	Yes	**No**	Yes	5
Lorentzian	11	Yes	Yes	Yes	Yes	Yes	**No**	5
Wave-Hedges	12	Yes	Yes	Yes	Yes	**No**	Yes	5
Matusita	13	Yes	Yes	Yes	Yes	Yes	Yes	6
Squared χ^2	14	Yes	Yes	Yes	**No**	Yes	Yes	5
Clark	15	Yes	Yes	Yes	Yes	**No**	Yes	5
Cosine	17	Yes	**No**	Yes	Yes	Yes	Yes	5
Jaccard	19	Yes	Yes	Yes	**No**	Yes	Yes	5
Dice	20	Yes	Yes	Yes	**No**	Yes	Yes	5
KLD	21	Yes	**No**	**No**	**No**	Yes	Yes	3
JDivergence	22	Yes	Yes	Yes	**No**	Yes	Yes	5
KDivergence	23	Yes	**No**	**No**	**No**	Yes	Yes	3
Topsoe	24	Yes	Yes	Yes	**No**	Yes	Yes	5
Jensen	25	Yes	Yes	Yes	**No**	Yes	Yes	5
Taneja	26	Yes	Yes	Yes	**No**	Yes	Yes	5
Kumar-Johnson	27	Yes	Yes	Yes	**No**	Yes	Yes	5

To illustrate some of the entries in this table, a few numerical examples will be given based on the following set of points in a two-dimensional space.

$$A = \begin{pmatrix}0.30\\0.70\end{pmatrix} B = \begin{pmatrix}0.15\\0.35\end{pmatrix} C = \begin{pmatrix}0.50\\0.50\end{pmatrix} D = \begin{pmatrix}0.70\\0.30\end{pmatrix} E = \begin{pmatrix}0.30\\0.00\end{pmatrix} F = \begin{pmatrix}0.00\\0.70\end{pmatrix} G = \begin{pmatrix}0.15\\0.70\end{pmatrix}$$

The first property specifies that the distance from a point to itself must be zero. This feature seems evident, and usually respected by all distance measures. A closer look reveals that applying the Motyka formulation (see Eq. (9)), the distance to itself is not equal to zero but 0.5. The following numerical example illustrates this issue with vector A.

$$dist_{Motyka}(A, A) = \frac{0.3 + 0.7}{0.3 + 0.3 + 0.7 + 0.7} = 0.5$$

This property does not always imply the reverse. Thus, when the distance between two points is zero, one cannot infer the two points are identical. For example, computing the Cosine similarity (Eq. (17)) between the points A and B, the similarity value is 1.0 and therefore the distance between them, according to the Cosine distance, is zero. The second property imposes that the distance between two distinct points must be larger than zero; this is not the case here.

$$sim_{Cosine}(A, B) = \frac{0.3 * 0.15 + 0.7 * 0.35}{\sqrt{0.3^2 + 0.7^2} * \sqrt{0.15^2 + 0.35^2}} = 1.0$$

$$dist_{Cosine}(A, B) = \frac{1}{\pi} cos^{-1}(1.0) = 0.0$$

Both vectors are pointing towards the same direction, but they do not have the same length. As one can see, the vector A is twice the vector B, and therefore the angle between them is zero, resulting in a Cosine distance of 0.0.

When considering the positivity (P2) and the symmetry property (P3), the distance measure based on the Kullback–Leibler divergence (KLD) (Eq. (21)) does not respect these two characteristics. In the following computation, one can see that the resulting value is negative. When KLD is applied between two probabilistic distributions, the returned value is always non-negative. In our context, it is not imposed that the sum of the elements of a vector is 1.0. Therefore, in some cases, the returned value could be negative.

$$dist_{KLD}(B, \mathrm{A}) = 0.15\ ln\left(\frac{0.15}{0.3}\right) + 0.35\ ln\left(\frac{0.35}{0.7}\right) = -0.347$$

Using the same argument, one can verify that the KDivergence distance (Eq. (23)) can return negative values. For the symmetry, the following computation shows that with the Kullback-Leibler divergence (KLD), this property is not respected. The distance is 0.693 while the distance from B to A is −0.347.

$$dist_{KLD}(A, \mathrm{B}) = 0.3\ ln\left(\frac{0.3}{0.15}\right) + 0.7\ ln\left(\frac{0.7}{0.35}\right) = 0.693$$

Many distance measures do not respect the fourth property, the triangle inequality. When considering the triangle {A, C, D}, the distance from A to D must be smaller (or equal) to the distance from A to C plus the distance from C to D. For example, with the Dice formula (Eq. (20)), one can obtain:

$$dist_{Dice}(A, D) = 1 - \frac{2 \cdot (0.3 \cdot 0.7 + 0.7 \cdot 03)}{0.3^2 + 0.7^2 + 0.7^2 + 0.3^2} = 0.276$$

$$dist_{Dice}(A, C) = 1 - \frac{2 \cdot (0.3 \cdot 0.5 + 0.7 \cdot 0.5)}{0.3^2 + 0.7^2 + 0.5^2 + 0.5^2} = 0.074$$

$$dist_{Dice}(C, D) = 1 - \frac{2 \cdot (0.5 \cdot 0.7 + 0.5 \cdot 0.3)}{0.5^2 + 0.5^2 + 0.7^2 + 0.3^2} = 0.074$$

and $dist(A, D) = 0.276 > dist(A, C) + dist(C, D) = 0.074 + 0.074 = 0.148$. As shown in Table 1, this property is not respected by several distance measures.

Regarding the fifth property (absence of an important feature), a few distance measures do not respect it, as, for example, the Canberra (Eq. (10)) or the Clark equation (see Eq. (15)). In our example, the vector A is composed of an important second component (with a value 0.7) while the first is smaller (0.3). The vector E has a zero value for the second coordinate, an important feature in describing vector A. On the other hand, the vector F owns exactly the same value for the second coordinate than A, but does not have the first one, a less important feature. Computing the distance from A to E or A to F with the Canberra measure, the same value is obtained. To obey the fifth property, the distance (A, E) must be larger than the distance (A, F).

$$dist_{Canberra}(A, E) = \frac{|0.3 - 0.3|}{|0.3| + |0.3|} + \frac{|0.7 - 0.0|}{|0.7| + |0.0|} = 1$$

$$dist_{Canberra}(A, F) = \frac{|0.3 - 0.0|}{|0.3| + |0.0|} + \frac{|0.7 - 0.7|}{|0.7| + |0.7|} = 1$$

Concerning the last property, we specify that the presence of a feature is better than its absence given the fact that the absolute difference is the same. This property is usually satisfied by numerous formulations. However, some of them do not follow it as, for example, the Manhattan distance. With the following numerical examples, the distance between vector A and G is the same as the distance between vector F and G. But in this example, the vector F does not have the first feature, and thus to respect this sixth property, the distance (F, G) must be larger than between A and G.

$$dist_{Manhattan}(A, G) = |0.3 - 0.15| + |0.7 - 0.7| = 0.15$$

$$dist_{Manhattan}(F, G) = |0.0 - 0.15| + |0.7 - 0.7| = 0.15$$

Respecting the set of proposed properties is important from a theoretical point of view and can form a set of criteria to select the most appropriate distance formulation. However, in practice a distance measure should also be easy to compute and provide overall good effectiveness for the targeted task. To evaluate this last aspect, we will evaluate the 24 distance measures according to the 13 test collections described in the next section.

5. Test collections and evaluation methodology

To provide large and reusable test collections, the CLEF was launched in 1999. In 2010, the PAN CLEF track was created to detect plagiarism, and in 2011 the authorship attribution issue was added. During the PAN CLEF 2013 campaign (Rangel Pardo, Rosso, Koppel, Stamatatos, & Inches, 2013) and in 2014 (Rangel et al., 2014), a profiling task was proposed. In this case, only the gender and age range are required to be determined based on blog posts, sequences of tweets, or reviews written in the English or Spanish language. The corresponding demographic category was extracted from the author's profile with some verifications (e.g., consulting Facebook or LinkedIn websites). The selected text register corresponds to messages written more or less spontaneously, without corrections done by an editor (as for newspaper articles) or a group of advisors/experts (as in official speeches).

In 2015, the PAN CLEF campaign (Rangel, Celli, Rosso, Potthast, Stein, & Daelemans, 2015; Stamatatos, Potthast, Rangel, Rosso, & Stein, 2015) added the Italian and Dutch languages but the text genre was limited to tweets. For the two new languages, only the gender of the author is provided. Finally, in 2016 (Rosso et al., 2016), the evaluation text genre was unknown but the training had to be performed using Twitter data in which the Dutch collection did not require an age range detection. More general information about these 13 author profiling collections extracted from the PAN CLEF campaigns are given in Table 2.

In this table, the corpus name corresponds to the concatenation of the last two digits of the year, the first letter of both the language and text genre. For example, 15ET denotes a test collection of year 2015, written in English, and containing tweets. The following columns indicate the year, language, and text genre of the corresponding corpus. Under the label "Problems", the value indicates the number of texts for which the system determines the gender, and the age range. The

Table 2
PAN CLEF 2014 to 2016 test collection statistics.

Name	Year	Language	Genre	Problems	Gender	Age Groups
14EB	2014	English	Blog	147	Male female	18–24, 25–34, 35–49, 50–64, 65-xx
14SB	2014	Spanish	Blog	88	Male female	18–24, 25–34, 35–49, 50–64, 65-xx
14ET	2014	English	Twitter	306	Male female	18–24, 25–34, 35–49, 50–64, 65-xx
14ST	2014	Spanish	Twitter	178	Male female	18–24, 25–34, 35–49, 50–64, 65-xx
14ER	2014	English	Review	4160	Male female	18–24, 25–34, 35–49, 50–64, 65-xx
14SS	2014	Spanish	Social media	1272	Male female	18–24, 25–34, 35–49, 50–64, 65-xx
15DT	2015	Dutch	Twitter	34	Male female	NA
15ET	2015	English	Twitter	152	Male female	18–24, 25–34, 35–49, 50-xx
15IT	2015	Italian	Twitter	38	Male female	NA
15ST	2015	Spanish	Twitter	100	Male female	18–24, 25–34, 35–49, 50-xx
16DT	2016	Dutch	Twitter	384	Male female	NA
16ET	2016	English	Twitter	436	Male female	18–24, 25–34, 35–49, 50–64, 65-xx
16SP	2016	Spanish	Twitter	250	Male female	18–24, 25–34, 35–49, 50–64, 65-xx

last two columns provide the possible value for the gender and age classes. A closer look on this table reveals that in 2015 only four age groups have been specified. For that year, the two oldest classes (50–64 and 65-xx) were merged into a single class (50-xx). When considering all test collections, one can find 7545 problems in total where 5201 are written in the English language, 1888 in Spanish, 418 in Dutch, and 38 in Italian.

As a performance measure, the accuracy rate (or success rate) has been adopted in the PAN CLEF evaluation campaign. This measure varies from 0 to 1 (or 100%), where a higher rate means a better result. This performance score can be computed for each demographic category individually, namely gender and age group. For example, if the system correctly predicts the author gender 7 times in 10 problems, the accuracy will be 0.7 for this category alone. The accuracy rate for both demographic categories will be reported in our experiments.

However, in the CLEF evaluation campaigns, the different systems are ranked according to a single value. To obtain a single overall effectiveness value, the fraction of problems where both the gender and age group are correctly predicted for the same problem is computed. Continuing with our previous example: If, for the age ranges, the classifier predicts correctly 5 times the age class over 10 problems, the accuracy is 0.5. To determine the quality of this classifier with respect to both the gender and the author age, the arithmetic mean could be used (e.g., $\frac{1}{2}(0.7+0.5)=0.6$). In the CLEF campaigns, the evaluation corresponding to both demographic categories is based on the number of correct assignments for *both* the gender and the age class. In our example, the classifier was able to correctly determine the gender and the age group for 4 problems, giving us an accuracy rate of $4/10=0.4$. As one can see in the evaluation shown in the following tables, the accuracy rate for determining both categories is always lower than the simple arithmetic mean.

6. Evaluation

Before presenting the evaluation results, the first section describes the k nearest neighbors classifier (k-NN) used in all our experiments. The following section presents the results over the 13-test collection using two different learning phases. In the last section, distinct text genres are employed in the training and test phase. With this experiment, one can estimate the loss of accuracy due to the use of different text genres in the training and testing phases.

6.1. K-nearest neighbors classifier

To evaluate the different distance measures, the top m most frequent terms (isolated words without stemming but with the punctuation symbols) forms the feature set. For determining the value of m, previous studies have shown that a value between 50 to 300 tends to provide the best performance (Burrows, 2002; Savoy, 2012, 2015; Kocher & Savoy, 2016). For all our experiments, we fixed $m=200$.

When considering the m most frequent terms from a query text, the terms appearing once (*hapax legomenon*) are ignored. Of course, for some short texts, the resulting representation can include less than 200 terms. The character appearing in uppercase are replaced by the corresponding lowercase letter. Thus, from the text "The cat jumps over the cat and over the table" the representation is [the: 3, cat: 2, over: 2]. The word "jumps", "and", or "table" occurring once are ignored. Finally, instead of directly using the occurrence frequencies of the ith term (denoted tf_i), the estimated probability is computed by dividing its occurrence frequency by the text length (measured in remaining tokens and denoted n). Therefore, for each vector component we have tf_i / n. In our previous example, the final representation of the query vector is [the: 3/7, cat: 2/7, over: 2/7] and for the comparison all other document vectors are built according to those three terms.

Web-based textual communication contains other forms than words. In a tweet, one can find hashtags (e.g., #nasa, #noobama) or various URLs (e.g., http://shakespeare.mit.edu, www.un.org). To take them into account, all hashtags or URLs are replaced by a common marker. The frequency of hashtags or URLs forms two additional features that have been shown to be effective, for example, to discriminate between Democrats and Republicans (Sylwester & Purver, 2015).

Table 3
Profiling results based on the same collection in training and test phase (macro-average over 13 test collections, leaving-one-out).

Measure	Training on the Same Corpus		
	Gender	Age	Both
Manhattan	0.6064	0.4535	0.3514
Euclidean	0.6131	0.4463	0.3566
Chebyshev	0.6049†	0.4313	0.3534†
Average	0.6105	0.4581	0.3643
Sørensen, Tanimoto, Kulczynski, Motyka	*0.6362†	*0.4626	***0.3865**
Canberra	*0.6290†	0.4504	0.3668†
Lorentzian	0.6281†	0.4446	0.3772†
Wave-Hedges	***0.6439**	0.4504	*0.3825†
Clark	0.6262†	0.4604	0.3682†
Matusita	0.6227†	***0.4868**	*0.3791†
Squared χ^2	*0.6293†	*0.4688	*0.3821†
Cosine	0.6114	0.4402	0.3569
Jaccard, Dice	0.6133	0.4489	0.3640
KLD	0.6102	*0.4608	0.3632†
JDivergence	0.6211†	0.4526	0.3670†
KDivergence	0.6246†	0.4476	0.3638†
Topsoe, Jensen	*0.6322†	*0.4727†	*0.3832†
Taneja	0.6162†	0.4594	0.3689†
Kumar–Johnson	0.6092	0.4558	0.3605
Mean	0.6204	0.4553	0.3682

Finally, in determining the corresponding demographic category of a query text, the distance with all other texts is computed and the five nearest neighbors (5-NN) are taken into account. From this set of the five closest neighbors, the majority determines the returned category, and in case of a tie, the closest neighbor defines the returned category. For example, if the age groups of the five nearest neighbors, in increasing distance, are 25–34, 18–24, 25–34, 18–24, and 35–49, the system assigns the label "25–34" to the query text because it is the closest group with the most members. This kind of classifier model has demonstrated overall good performance in a similar task (Kocher & Savoy, 2016). The remaining question is to know which distance measure offers the best performance.

6.2. Evaluation based on same text genre for training and test

To determine the accuracy rate for each of the 13 test collections and considering the 24 distance measures, a first set of experiments is based on the leaving-one-out (LOO) methodology (Witten, Frank, & Hall, 2011; Zhai et al, 2016). This evaluation approach guarantees an unbiased estimation of the true performance. Instead of reporting all possible combinations of each corpus according to each distance measure, only the average will be reported as shown in Table 3. To achieve this, the macro-average principle (Sebastiani, 2002) was applied, giving the same importance to each test collection. In other words, the mean is computed over all corpora instead of over each decision (micro-averaging). When considering the size of each corpus given in Table 2, the result of the micro-average method will be dominated by the 14ER corpus having 4160 problems while, for example, the 15DT (34 problems) will have an insignificant effect on the overall performance. Thus, we prefer giving the same importance to each test collection, and the macro-averaging method was adopted.

When adopting a distance measure, the returned distance is used to select the top five closest neighbors for each problem. The distance value by itself is not directly used. Therefore, even if one can see a small difference between two distance formulations, the effect in selecting the five nearest neighbors is nil. For example, applying the Jaccard (Eq. (19)) or Dice (Eq. (20)) distances, the returned value is not strictly the same but the selection of the five closest neighbors is the same (however, maybe not in the same order). Therefore, instead of presenting both measures in Table 3, both distances are merged into a single row. The same phenomenon appears with the Topsoe (Eq. (24)) and Jensen (Eq. (25)) measures, as well as with the following four distance formulas: Sørensen (Eq. (6)), Tanimoto (Eq. (7)), Kulczynski (Eq. (8)), and Motyka (Eq. (9)).

In Table 3, the first column indicates the name of the distance measure and the next three columns report the accuracy rates when applying the leaving-one-out approach. The first value corresponds to the gender problem, the second to the age class determination, and the third indicates the percentage of correct answers for both the gender and age group at the same time. As one can see, the gender categorization problem is always easier than the age group. The third evaluation was clearly the most difficult, and the reported accuracy rates are always smaller than for the age detection. For example, applying the Euclidian distance, the average over 13 test collections for the gender problem, the accuracy rate is 0.6131. Under the same condition, the age group determination achieved a mean performance of 0.4463 while the proportion of correct decisions for both the gender and age group is 0.3566.

In Table 3, the highest performance per column is depicted in bold and an asterisk indicates the top best five cells per columns. When considering measures appearing in the top five, the Sørensen (and Tanimoto, Kulczynski, and Motyka), Squared χ^2, and Topsoe (and Jensen) formulas occur three times, while the Wave-Hedges and Matusita appear two times.

Table 4
Corpus used in the training and test phase.

Training	Test	Training	Test
14ET	15ET	14ET	16ET
15ET	14ET	15ET	16ET
16ET	14ET	16ET	15ET
14ST	15ST	14ST	16ST
15ST	14ST	15ST	16ST
16ST	14ST	16ST	15ST
15DT	16DT	16DT	15DT

Table 5
Profiling results based on the same text genre in training and test phase (macro-average over 14 collections).

Measure	Training on the Same Corpus Leaving-one-out			Training on a Second Corpus Same Text Genre		
	Gender	Age	Both	Gender	Age	Both
Manhattan	0.6249	0.5065	0.3657	0.5857	*0.4377	0.3131
Euclidean	0.6270	0.5151	0.3803	0.5888	0.4191	0.3053
Chebyshev	0.6091	0.4805	0.3503	0.5752	0.3881	0.2821
Average	0.6257	0.5113	0.3801	0.5914	*0.4444	0.3192
Sørensen, Tanimoto, Kulczynski, Motyka	0.6569†	*0.5294	*0.4096†	0.5926	0.4131	0.3106
Canberra	*0.6665†	0.5126	0.3930	***0.6248**	*0.4353	*0.3436†
Lorentzian	0.6475	0.5022	0.3920†	0.5927†	0.4287†	0.3054
Wave-Hedges	***0.6707**	0.5076	0.4000†	*0.6200†	0.4263	*0.3379†
Clark	*0.6631†	*0.5285†	0.3939†	*0.6190†	***0.4528**	***0.3463**
Matusita	0.6556†	***0.5497**	*0.4096†	*0.6079†	0.4219	0.3144
Squared χ^2	0.6550†	*0.5353†	*0.4089†	0.6072†	0.4212	*0.3256
Cosine	0.6227	0.5053	0.3733	0.5767	0.4119	0.2977
Jaccard, Dice	0.6240	0.5094	0.3814	0.5796	0.4014	0.3018
KLD	0.6466	0.5253	0.3931	0.6065†	0.4217†	0.3130
JDivergence	0.6596†	0.5052	0.3895†	0.6065†	0.4259†	0.3098
KDivergence	0.6438	0.4891	0.3737	0.5900	0.3679	0.3002
Topsoe, Jensen	*0.6613†	*0.5387†	***0.4101**	*0.6183†	0.4190	*0.3260
Taneja	*0.6600†	0.5222	*0.4044†	0.5987†	*0.4292†	0.3048
Kumar–Johnson	0.6275	0.5177	0.3772	0.5628	0.4050	0.2763
Mean	0.6446	0.5153	0.3887	0.5971	0.4195	0.3123
Accuracy loss				7.4%	19.1%	19.7%

Canberra and KLD each make it once in the top five. On the other hand, the Manhattan, Euclidian, Chebyshev, Average, Lorentzian, Clark, Cosine, Jaccard (and Dice), JDivergence, KDivergence, Taneja, or Kumar-Johnson never appear in the best five measures on the three tasks. To statistically determine whether or not a given distance measure is statistically worse than the best one (depicted in bold), we applied the *t*-test whereby the null hypothesis H_0 states that both distance measures result in similar performance levels. In the experiments, statistically non-significant differences are indicated by a cross (†) (paired, two-sided test, significance level $\alpha = 5\%$).

Table 3 reports the mean accuracy rate achieved when using the same collection for both the training and test phase. However, the instances used during the test phase never occur during the training (leaving-one-out methodology) (Witten et al., 2011). When considering the available corpora, the training stage can be performed using another text collection. As shown in Table 2, the different collections share some common characteristics such as the language or the text genre. Thus, instead of deriving the text representations from the same corpus as previously, these profiles can be built according to another corpus written, of course, in the same language but also having the same text genre. For example, the decisions related to corpus ET14 can be based on corpus 15ET or 16ET. Table 4 indicates the 14 different combinations that can be obtained when considering the 13 test collection. Obviously, having just one corpus in the Italian language, it was impossible to perform this kind of evaluation in Italian. Moreover, one can observe that the English collections 14ET, 15ET, and 16ET appear twice in the test stage. The same occurs with the Spanish corpora 14ST, 15ST, and 16ST. However, the two Dutch collections (15DT and 16DT) appear only once.

To have a fair comparison between the two forms of training, the accuracy rates reported in Table 3 are not appropriate. Thus, in Table 5 the left part indicates the overall performance under the leaving-one-out methodology but using the set of collection appearing in Table 4 under the column "Test". This means the performance achieved by the corpus 14ET, 15ET, and 16ET are computed twice while the accuracy of the Italian collection is ignored. Therefore, in total 14 collections will be used to estimate the accuracy rate.

The left part of Table 5 indicates the overall performance achieved with those 14 corpora, using in the training the same collection (leaving-one-out). On the right side of this table, one can find the same three accuracy rates obtained with another

Table 6
Corpus used in the training and test phase (cross-genre evaluation).

Training	Test	Training	Test	Training	Test	Training	Test
14EB	14ER	14ER	16ET	14SS	14SB	15ET	14ER
14EB	14ET	14ET	14EB	14SS	14ST	15ST	14SB
14EB	15ET	14ET	14ER	14SS	15ST	15ST	14SS
14EB	16ET	14SB	14SS	14SS	16ST	16ET	14EB
14ER	14EB	14SB	14ST	14ST	14SB	16ET	14ER
14ER	14ET	14SB	15ST	14ST	14SS	16ST	14SB
14ER	15ET	14SB	16ST	15ET	14EB	16ST	14SS

corpus for the two demographic categories and the combined evaluation. The last row reports the arithmetic average over the 24 distance measures.

The effectiveness values reported in Table 5 indicate that the Clark and Canberra appear to achieve the best overall performances when the learning is based on a distinct collection having the same text genre. Overall, when considering measures appearing in the top five, the Clark and Topsoe (and Jensen) formula occur five times, and the Canberra four times. On the other hand, the Euclidian, Chebyshev, Lorentzian, Cosine, Jaccard (and Dice), KLD, JDivergence, KDivergence, or Kumar-Johnson never appear in the best five measures over these six categorization tasks.

Using a distinct corpus with the training stage tends to hurt the overall performance of the classification system, whatever the distance measure is. As depicted in the last row of Table 5, the mean degradation goes from 7.4% for the gender classification to 19.1% for the age. When considering both the gender and age classification, the decrease reaches 19.7%.

From an efficiency point of view, the increasing complexity from the L^1 family, to the L^2 family, to the Inner Product family, and to the Entropy family is directly reflected in an increasing runtime. Therefore, computing a distance from the L^1 family is faster than any of the distances from other families while one from the Entropy family is slower than all measures from any other distance family. Based on our experiments, the Manhattan distance (Eq. (1)) takes the least computing time, while Topsoe's formulation (Eq. (24)) can be from 20% up to 80% slower.

6.3. Cross-genre evaluation

The training phase can be performed on a corpus written, obviously, in the same language as the test instances, but having a distinct text genre. For some applications, the correspondence between the training and test sets cannot be as close as one would wish. Therefore, the training has to be performed on a different text genre, which is a solution that will have an impact on the overall classification performance. But to what extent? Certainly, the distance between the training and test set should be as close as possible. In the 13 test collections described in Table 2, one can observe that they correspond to web-based communication with an emphasis on tweets. The remaining question is to estimate the loss of effectiveness when learning on one web-based text genre to test on another one. Of course, considering two very distinct text genres (such as oral vs. written formal speech (Biber & Conrad, 2009)) will produce higher accuracy rate degradation.

To evaluate this degradation in the accuracy rate, we design a set of experiments in which the training text genre differs from the test phase. Table 6 depicts the different possible configurations applied to obtain results shown in Table 7. As for the previous tables, on the left one can see the accuracy rates achieved using the same collection during the training and the test phases. On the right, the performance values are obtained using different text genres in training and testing.

As in the evaluation on the same text genre in Table 5, the Clark measure appears again to achieve the best overall performance when the learning is based on a distinct collection having a different text genre. When considering measures appearing in the top five, the KLD, JDivergence, and Taneja formula occur three times in the cross-genre evaluation but they never occur in the results when the training was on the same corpus. Conversely, the Sørensen (and Tanimoto, Kulczynski, and Motyka) appears three times in the top five on the left-hand side of the table, but are missing from the top five in the right part when the training was on a different text genre. On the other hand, the Manhattan, Euclidian, Chebyshev, Cosine, Jaccard (and Dice), or Kumar-Johnson never appear in the best five measures over these six categorization tasks.

One can notice that the performance drop between same-genre and cross-genre gender predictions is more than 11.2%, the estimation for the age groups decreases about 26.1%, and the loss of accuracy when classifying both attributes is 35.1% in cross-genre compared to the same-genre evaluations. This means that not all style markers are transferred from one genre to another, which leads to misclassifications.

7. Conclusion

From a practical point of view, this paper investigates the problem of the selection of the best performing distance measure when designing a classifier to solve the author profiling question. In this perspective, the gender (two categories) and the age group (four to five classes) of the author are required to be determined as accurately as possible. This problem is characterized by a relatively large number of possible features, without having some dominating all the others. In this context, 24 distance measures have been briefly described reflecting five main families of functions (L^1, L^2, inner product, entropy-based, and combination approaches).

Table 7
Profiling results based on different text genres in training and test phase (macro-average over 28 collections).

Measure	Training on the Same Corpus Leaving-one-out			Training on a Second Corpus Different Text Genre		
	Gender	Age	Both	Gender	Age	Both
Manhattan	0.6094	0.4308	0.2655	0.5287	0.3177†	0.1729†
Euclidean	0.6075	0.4169	0.2544	0.5245	0.2999	0.1635
Chebyshev	0.5867	0.4102	0.2489	0.5289	0.2926	0.1571
Average	0.6076	*0.4353	0.2731†	0.5256	0.3133†	0.1698
Sørensen, Tanimoto, Kulczynski, Motyka	*0.6216†	*0.4341	***0.2815**	0.5373	0.3043	0.1683
Canberra	*0.6225†	0.4238	0.2725†	*0.5629†	0.3211†	*0.1877†
Lorentzian	0.6166	0.4199	*0.2782†	0.5382	0.3174	0.1734
Wave-Hedges	***0.6296**	0.4259	*0.2795†	0.5473	0.3176	0.1754
Clark	*0.6195†	0.4312	0.2751†	***0.5669**	0.3245†	***0.1923**
Matusita	0.6099	***0.4599**	*0.2805†	0.5490	*0.3266†	0.1837†
Squared χ^2	0.6132	*0.4404	0.2772†	0.5486†	0.3159	0.1798†
Cosine	0.6112	0.4122	0.2632	0.5398	0.3212†	0.1792†
Jaccard, Dice	0.6105	0.4229	0.2688	0.5289	0.3090	0.1639
KLD	0.6012	0.4331	0.2694	*0.5511†	***0.3461**	*0.1920†
JDivergence	0.6099	0.4301	0.2731†	*0.5499†	*0.3413†	*0.1861†
KDivergence	*0.6218†	0.4298	0.2724†	0.5348	0.2852	0.1518
Topsoe, Jensen	0.6181	*0.4444	*0.2813†	0.5498†	*0.3272†	0.1839†
Taneja	0.6055	0.4325	0.2739	*0.5548†	*0.3384†	*0.1893†
Kumar–Johnson	0.5963	0.4293	0.2595	0.5455	0.3148	0.1685
Mean	0.6115	0.4296	0.2710	0.5428	0.3176	0.1757
Accuracy loss				11.2%	26.1%	35.1%

From a theoretical point of view, a set of six theoretical properties have been presented. Only two formulations (Tanimoto and Matusita) respect all these requirements, with the other 20 respecting five of these properties. Looking at their definition, the difference between these 24 distance measures is usually rather small.

From an empirical point of view, an evaluation has been performed. To achieve this, 13 test collections extracted from PAN CLEF evaluation campaigns have been selected. These corpora cover four text genres (tweets, blogs, reviews, and social media) and four languages (English, Spanish, Dutch, and Italian). To evaluate the different distance measure, the *k*-NN classifier is used, and the top five closest neighbors are employed to determine the demographic category. Based on the leaving-one-out methodology, the Sørensen (and Tanimoto, Kulczynski, and Motyka), Wave-Hedges, and Matusita distance measures tend to show the best performance. The performance differences are however usually not significant between the best 5 distance measures. On the other hand, the Cosine distance measure, well-known in various distributed language models (Bengio, 2009), tend to produce rather lower performance levels. In the IR domain (Manning et al., 2008), the Dice and Jaccard distance measures are also recommended but both depict a lower accuracy rate than the best performing measures. From an efficiency perspective, the Manhattan measure presents a clear advantage, having usually the smallest computation time.

In this paper, we also evaluate the differences in accuracy rates when the training corpus is not the same as the test one. Compared to having the same corpus in both the training and test phases, the overall performance decreases from around 7.4% (gender classification only) to 19.7% (both gender and age classification). As an additional evaluation, the training was performed on a distinct text genre than the test one. However, both reflect the general web-based writing style. In this case, the total accuracy rate tends to decrease from 11.1% (limited to gender classification) to 35.0% (classification of both the gender and age group).

The current study has its own limitations. The focus is placed on a specific text categorization problem: that of author profiling. In this case, the number of features are relatively large and many of them tend to have similar frequencies. We do not have one or a few features dominating the others that can by themselves discriminate between the different targeted categories. The experiments were based only on web-based mediated texts and additional evaluations performed on other text genres should be done to confirm our main findings. The results are also not directly applicable to neural networks and word embedding approaches where the distance measures are not used in explicit form.

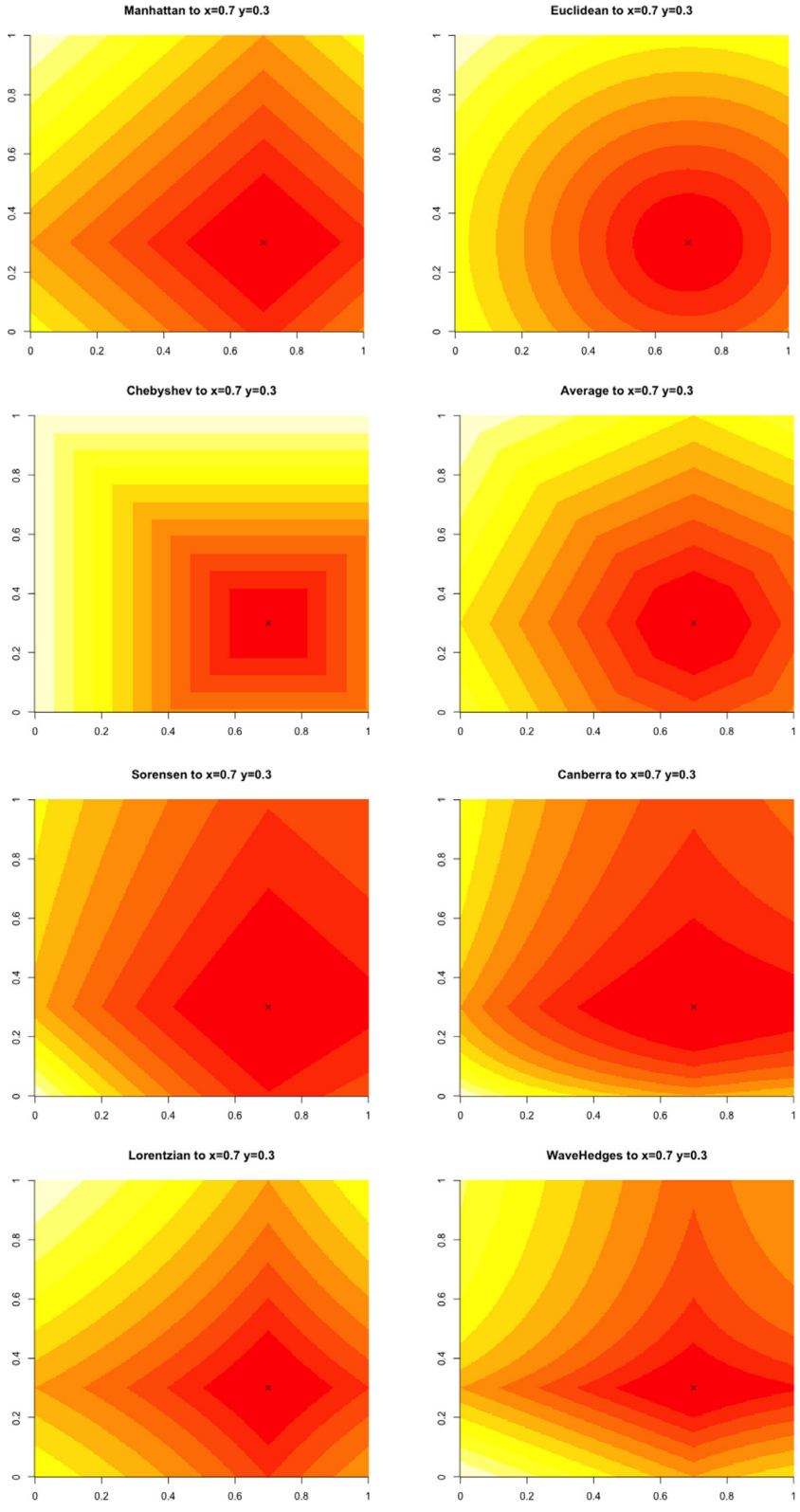
Acknowledgments

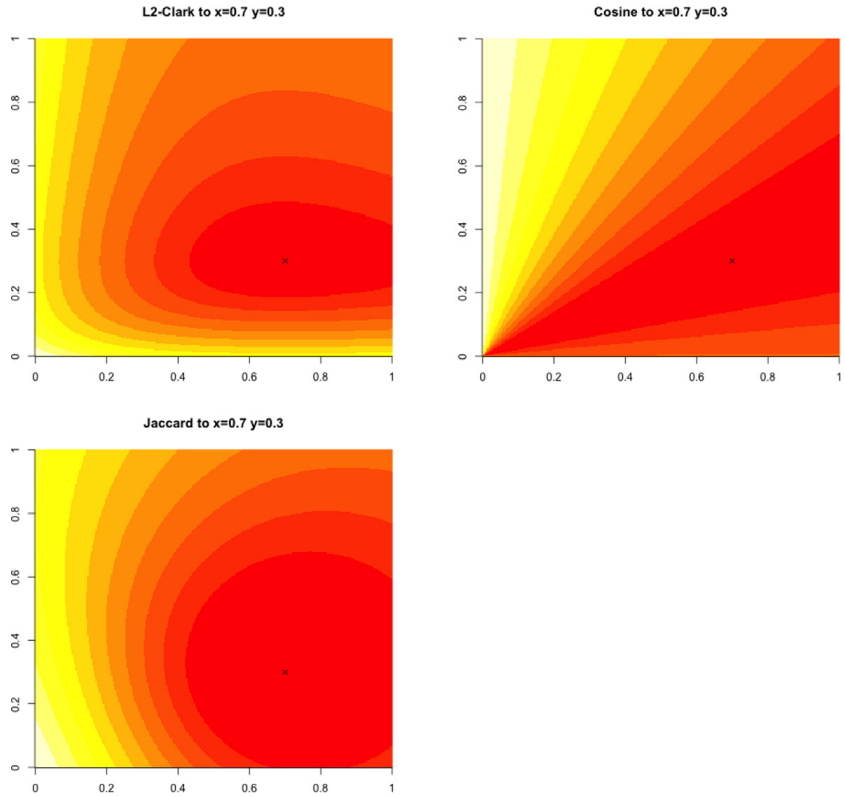
This research was supported, in part, by the NSF under Grant #200021_149665/1. The authors want to thank the anonymous reviewers for their helpful suggestions and remarks.

Appendix

To visualize the different distance measures, we consider a reduced vector space with only two dimensions. We assume that the coefficients of a vector represent the underlying probabilities. Therefore, the axis are ranging from 0 to +1. A point at position $x = 0.7$ and $y = 0.3$ is selected and the distance to all other points in $[0, 1] \times [0, 1]$ is calculated according to

various measures. The plots show dark red areas where the distance to the point is small (or the corresponding similarity is high), orange zones for more distant fields, and bright yellow regions for the farthest (least similar) groups.





References

Alowibdi, J. S., Buy, U. A., & Yu, P. (2013). Empirical evaluation of profile characteristics for gender classification on twitter. In *Proceedings of the international conference on machine learning and applications* (pp. 365–369).

Álvarez-Carmona, M. A., López-Monroy, A. P., Montes-Y-Gómez, M., Villaseñor-Pineda, L., & Jair-Escalante, H. (2015). INAOE's participation at PAN'15: Author profiling task. In L. Cappellato, N. Ferro, G. J. F. Jones, & E. SanJuan (Eds.), *Proceeding CLEF-2015, working notes*. CEUR.

Argamon, S., Dhawle, S., Koppel, M., & Pennebaker, J. W. (2005). Lexical predictors of personality type. In *Proceedings of the 2005 joint annual meeting of the interface and the classification society* (pp. 1–16).

Argamon, S., Koppel, M., Pennebaker, J. W., & Schler, J. (2009). Automatically profiling of the author of an anonymous text. *Commun. ACM, 52*(3), 119–123.

Baayen, H. R. (2008). *Analysis linguistic data: A practical introduction to statistics using r*. Cambridge: Cambridge University Press.

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning, 2*(1), 1–127.

Biber, C., & Conrad, S. (2009). *Register, genre, and style*. Cambridge: Cambridge University Press.

Bilan, I., & Zhekova, D. (2016). Caps: A cross-genre author profiling system. In K. Balog, L. Cappellato, N. Ferro, & C. Macdonals (Eds.), *Proceeding CLEF-2016, Working Notes* (pp. 824–835). CEUR.

Burrows, J. F. (2002). Delta: A measure of stylistic difference and a guide to likely authorship. *Literary and Linguistic Computing, 17*(3), 267–287.

Busger Op Vollenbroek, M., Carlotto, T., Kreutz, T., Medvedeva, M., Pool, C., Bjerva, J., et al. (2016). In K. Balog, L. Cappellato, N. Ferro, & C. Macdonals (Eds.), *Proceeding CLEF-2016* (pp. 846–857). CEUR. Working Notes.

Ciot, M., Sonderegger, M., & Ruths, D. (2013). Gender inference of Twitter users in non-English contexts. In *Proceedings of conference on empirical methods in natural language processing* (pp. 1136–1145).

Cha, S-H. (2007). Comprehensive survey on distance similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences, 1*(4), 300–307.

Coates, J., & Pichler, P. (2011). *Language and gender*. Chichester: Wiley-Blackwell.

Collberg, C., & Proebsting, T. A. (2016). Repeatability in computer systems research. *Communications of the ACM, 59*(3), 62–69.

Craig, H., & Kinney, A. F. (2009). *Shakespeare, computers, and the mystery of authorship*. Cambridge: Cambridge University Press.

Crystal, D. (2006). *Language and the internet*. Cambridge: Cambridge University Press.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification*. New York: Addison-Wesley.

Eckert, P., & McConnell-Ginet, S. (2013). *Language and gender*. Cambridge: Cambridge Univeristy Press.

Fung, G. (2003). The disputed *Federalist Papers*: SVM features selection via concave minimization. In Proceeding ACM-TAPIA Conference (pp. 42–46).

González-Gallardo, C. E., Montes, A., Sierra, G., Núñez, A., Adolfo, S., & Ek, J. (2015). In L. Cappellato, N. Ferro, G. J. F. Jones, & E. SanJuan (Eds.), *Tweets classification using corpus dependent tags, character and pos n-grams*. CEUR Working Notes.

Grivas, A., Krithara, A., & Giannakopoulos, G. (2015). Author profiling using stylometric and structural feature groupings. In L. Cappellato, N. Ferro, G. J. F. Jones, & E. SanJuan (Eds.), Proceeding CLEF-2015. CEUR *Working Notes*.

Gronenschild, B. M., Habets, P., Jacobs, I. L., Mengelers, N., van Os, J., & Marcelis, M. (2012). The effects of FreeSurfer version, workstation type, and Macintosh operating system version on anatomical volume and cortical thickness measurements. *PLOS*.

Holmes, D. I. (1998). The evolution of stylometry in humanities scholarship. *Literary and Linguistic Computing, 13*(3), 111–117.

Jockers, M. L., & Witten, D. M. (2010). A comparative study of machine learning methods for authorship attribution. *Literary and Linguistic Computing, 25*(2), 215–223.

Khonji, M., & Iraqi, Y. (2014). A slightly-modified GI-based author-verifier with lots of features (ASGALF). In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.), *Proceeding CLEF-2014* (pp. 977–983). CEUR. *Working Notes*.

Kocher, M., & Savoy, J. (2016). A simple and efficient algorithm for authorship verification. *Journal of the American Society for Information Science and Technology, 68*(1), 259–269.

López-Monroy, A. P., Montes-Y-Gómez, M., Jair-Escalante, H., & Villaseñor-Pineda, L. (2014). Using intra-profile information for author profiling. In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.), *Proceeding CLEF-2014* (pp. 1116–1120). CEUR. *Working Notes*.
Maharjan, S., Shrestha, P., & Solorio, T. (2014). A simple approach to author profiling in mapreduce. In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.), *Proceeding CLEF-2014* (pp. 1121–1128). CEUR. *Working Notes*.
Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge: Cambridge University Press.
Modaresi, P., Liebeck, M., & Conrad, S. (2016). Exploring the effects of cross-genre machine learning for author profiling in PAN 2016. In K. Balog, L. Cappellato, N. Ferro, & C. Macdonals (Eds.), *Proceeding CLEF-2016* (pp. 970–977). CEUR. *Working Notes*.
Mosteller, F., & Wallace, D. L. (1964). *Applied bayesian and classical inference: The case of the federalist papers*. Reading: Addison-Wesley.
Nguyen, D., Trieschnigg, D., Seza Doğruöz, A., Gravel, R., Theune, M., Meder, T., et al. (2014). Why gender and age prediction from tweets is hard: Lessons from a crowdsourcing experiment. In *Proceedings of 25th international conference on computational linguistics* (pp. 1950–1961).
Olsson, J. (2008). *Forensic linguistics*. London: Continuum.
Pennebaker, J. W. (2011). *The secret life of pronouns. What our words say about us*. New York: Bloomsbury Press.
Rangel, F., & Rosso, P. (2016). On the impact of emotions on author profiling. *Information Processing & Management, 52*(1), 73–92.
Rangel Pardo, F. M., Rosso, P., Koppel, M., Stamatatos, E., & Inches, G. (2013). Overview of the author profiling task at PAN 2013. *Working Notes for CLEF 2013 Conference*. Valencia, Spain.
Rangel, F., Rosso, P., Chugur, I., Potthast, M., Trenkmann, M., Stein, B., et al. (2014). Overview of the 2nd author profiling task at PAN 2014. In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.). In *Notebook papers of CLEF 2014 labs and workshop: 1180* (pp. 827–898). Aachen.
Rangel, F., Celli, F., Rosso, P., Potthast, M., Stein, B., & Daelemans, W. (2015). Overview of the 3rd author profiling task at PAN 2015. In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.). *Notebook papers of CLEF 2015 labs and workshop*: 1391. Aachen.
Rosenthal, S., & McKeown, K. (2011). Age prediction in blogs: A study of style, content, and online behavior in pre- and post-social media generations. In *Proceedings of the 49th annual meeting of the association for computational linguistics* (pp. 763–772).
Rosso, P., Rangel, F., Potthast, M., Stein, B., Stamatatos, E., Tschuggnall, M., et al. (2016). *Experimental IR meets multilinguality, multimodality, and interaction* (pp. 332–350). Heidelberg: Springer.
Sap, M., Park, G., Eichstaedt, J. C., Kern, M. L., Stillwell, D., Kosinski, M., et al. (2014). Developing age and gender predictive lexica over social media. In *Proceedings of conference on empirical methods in natural language processing* (pp. 1146–1151).
Savoy, J. (2012). Authorship attribution based on specific vocabulary. *ACM – Transactions on Information Systems, 30*(2), 170–199.
Savoy, J. (2015). Comparative evaluation of term selection functions for authorship attribution. *Digital Scholarship in the Humanities, 30*(2), 246–261.
Savoy, J. (2016). Text representation strategies: An example with the *State of the Union* addresses. *Journal of the American Society for Information Science and Technology, 67*(8), 1858–1870.
Schler, J., Koppel, A., Argamon, S., & Pennebaker, J. (2006). Effects of age and gender on blogging. In *Proceedings AAAI spring symposium on computational approaches for analyzing weblogs* (pp. 191–197).
Sebastiani, F. (2002). Machine learning in automatic text categorization. *ACM Computing Survey, 34*(1), 1–27.
Stamatatos, E. (2009). A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology, 60*(3), 214–433.
Stamatatos, E., Potthast, M., Rangel, F., Rosso, P., & Stein, B. (2015). Overview of the PAN/CLEF 2015 evaluation lab. In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.), *Proceedings of the notebook papers of CLEF 2015 labs and workshop: 1391*. CEUR.
Sylwester, K., & Purver, M. (2015). Twitter language use to reflect psychological differences between Democrats and Republicans. *PLoS One, 10*(9). doi:10.1371/journal. pone.0137422.
Talbot, M. (2010). *Language and gender*. Malden: Polity Press.
Tausczik, Y. R., & Pennebaker, J. W. (2010). The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of Language and Social Psychology, 29*(1), 24–54.
Weren, E. R. D., Moreira, V. P., & de Oliveira, J. P. (2014). Exploring information retrieval features for author profiling. In L. Cappellato, N. Ferro, M. Halvey, & W. Kraaij (Eds.), *Proceeding CLEF-2014* (pp. 1164–1171). CEUR. Working Notes.
Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data mining. Practical machine learning tools and techniques*. Burlington: Morgan Kaufmann.
Yule, G. (2010). *The study of language*. (4th Ed.) Cambridge: Cambridge University Press.
Zhai, C. X., & Massung, S. (2016). *Text data management and analysis*. New York: The ACM Press.
Zobel, J., & Moffat, A. (1998). Exploring the similarity space. *ACM-SIGIR Forum, 32*(1), 18–34.

A.4 Evaluation of Text Representation Schemes and Distance Measures for Authorship Linking

Mirco Kocher, Jacques Savoy.
In *Scientometrics (Special Issue Proposal on Scieno-Network-Mining)*, submitted.

Evaluation of Text Representation Schemes and Distance Measures for Authorship Linking

Mirco Kocher, Jacques Savoy
University of Neuchatel (Switzerland)
{Mirco.Kocher, Jacques.Savoy}@unine.ch

Abstract. Based on *n* text excerpts, the authorship linking task is to determine a way to link pairs of documents written by the same person together. This problem is closely related to authorship attribution questions and its solution can be used in the author clustering task. However, no training information is provided and the solution must be unsupervised. To achieve this, various text representation strategies can be applied, such as characters, punctuation symbols, or letter *n*-grams as well as words, lemmas, Part-Of-Speech (POS) tags, and sequences of them. To estimate the stylistic distance (or similarity) between two text excerpts, different measures have been suggested based on the L^1 norm (e.g., Manhattan, Tanimoto), the L^2 norm (e.g., Matusita), the inner product (e.g., Cosine), or the entropy paradigm (e.g., Jeffrey divergence). From those possible implementations, it is not clear which text representation and distance functions produce the best performance and this study provides an answer to this question. Three corpora, extracted from French and English literature, have been evaluated using standard methodology. Moreover, we suggest an additional performance measure called high precision capable of judging the quality of a ranked list of links to provide only correct answers. No systematic difference can be found between token- or lemma-based text representations. Simple POS tags do not provide an effective solution but short sequences of them form a good text representation. Letter *n*-grams (with $n = 4$ to 6) give high precision rates. As distance measures, this study found that the Tanimoto, Matusita, and Clark distance measures perform better than the often-used Cosine function. Finally, applying a pruning procedure (e.g., culling terms appearing once or twice or limiting the vocabulary to the 500 most frequent words) reduces the representation complexity and might even improve the effectiveness of the attribution scheme.

Keywords. *Authorship linking, clustering, authorship attribution, stylometry, classification.*

1. Introduction

Due to the presence of numerous pseudonymous posts, chats, threatening e-mails, and anonymous messages on the Web, the authorship attribution domain has encountered an increasing interest (Olsson, 2008). To accurately determine the true author of a text, various approaches have been proposed and evaluated (Juola, 2006; Stamatatos, 2009). This field can mainly be subdivided into four distinct questions. First, the closed-class attribution problem assumes that the real author is one of the specified candidates. Second, in the open-set situation, the real author could be one of the proposed authors or another unknown one. Third, the verification question provides a binary response as to whether a given author did in fact write a given text (Koppel *et al*., 2007). Finally, authorship attribution can be limited to determining demographic (gender, age class, native language) or psychological traits of the author (Argamon *et al*., 2009; Pennebaker, 2011; Rangel & Rosso, 2016).

In all these cases, the proposed methods assume that a set of documents written by the different possible authors (or categories of authors, such as men and women) are available.

The current study focusses on a radically different perspective where the presence of such labeled data is not provided. The targeted question, called authorship linking, is defined as follows. Having a set of *n* documents (or text excerpts) written by several distinct authors, determine the pairs of documents written by the same person. In the related task called author clustering, the objective is similar and usually builds upon this task. In the clustering case, the number *k* of distinct authors must be determined to form *k* distinct author clusters based on a preset threshold for the ranked list of authorship links. As possible applications for both problems, a set of proclamations written by different terrorist groups can be regrouped, a collection of reviews written by the same author can be gathered (Almishari & Tsudik, 2012), or a set of poems (or excerpts of literary works) can be assembled. To solve this task, an unsupervised approach must be designed and evaluated.

In this context, the first challenge is to represent the text in an effective way, and it is not clear which text representation proposes the highest linking effectiveness. Past studies indicate that very frequent word-types or functional words can closely reflect the personal style of each writer while other researchers suggest taking account of the entire vocabulary. As a third view, other experiments propose to ignore terms having a low occurrence frequency (e.g., appearing once or twice). Finally, the Part-Of-Speech (POS) distribution can be used to reflect the stylistic characteristics of the different authors. Further concerns can be found when choosing the most appropriate distance (or similarity) measure between two text extracts. For example, in the information retrieval (Manning *et al*., 2008) or deep learning community (Goodfellow *et al*., 2016), Cosine corresponds to the most popular measure. However, many other distance measures (Duda *et al*., 2001) do exist and their success in the authorship linking problem is largely unknown.

To provide an answer to these questions, and to determine the most effective text representation, the rest of this paper is organized as follows. The next section presents the state of the art in authorship attribution. The third section describes the three test collections used in our experiments while the fourth exposes the evaluation methodology. The fifth section evaluates various word-based text representations and distance measures for the authorship linking task. In the sixth section, an evaluation of different word-based representations is described while in the seventh, various POS text representations are presented and evaluated. Different letter *n*-gram text surrogates are built and evaluated in the eighth section while some efficiency questions and their effect on the effectiveness are described in the ninth section. Finally, a conclusion draws the main findings of this study.

2. Related Work

To achieve an effective solution for the authorship linking task, two main challenges must be solved. First, a text representation must be defined reflecting the stylistic aspects of the author, without specifically taking account of the text genre or the topics. Second, an effective distance measure between two text representations must be determined. Such a function must return a small value when the two documents are written by the same author, and a larger one otherwise. Instead of applying a distance measure, a similarity measure can be used to state that two texts were written by the same person when the similarity value is high enough.

The choice of the text representation and the distance measure are related to classical challenges in authorship attribution, but we must solve them in an unsupervised perspective.

In the current context, training data is not available and thus author profiles cannot be derived from a sample of documents for which the authorship is known.

To represent the stylistic aspects of an author, a first set of methods suggests defining an invariant stylistic measure (Holmes, 1998) reflecting the particular style of a given author and varying from one person to another. As possible solutions, different lexical richness measures or word distribution indicators have been proposed such as Yule's K measure, statistics related to the type-token ratio (TTR) (e.g., Herdan's C, Guiraud's R or Honoré's H), the proportion of word-types occurring once or twice (e.g., Sichel's S) as well as the average word length, or the mean sentence length. None of these measures has proven very satisfactory due, in part, to word distributions ruled by a large number of very low probability elements (Large Number of Rare Events) (Baayen, 2008).

As a second framework, a multivariate method can be applied to project each document representation into a reduced space under the assumption that texts written by the same author should appear close together. Some of the main approaches applicable here are principal component analysis (PCA) (Burrows, 1992; Binonga & Smith, 1999; Craig & Kinney, 2009), hierarchical clustering (Labbé, 2007; Cortelazzo *et al*., 2016), or discriminant analysis (Ledger & Merriam, 1994; Jockers & Witten, 2010). As stylistic features, these approaches tend to employ the top 50 to 200 most frequent word-types (MFW), as well as some POS information.

As a third useful paradigm, and based on various word selection schemes, different distance-based measures have been suggested. As well-known strategies, one can mention Burrows' Delta (2002) using the top *m* MFW (with *m* = 40 to 1,000), the Kullback-Leibler divergence (Zhao & Zobel, 2007) using a predefined set of 363 English words, or Labbé's method (2007) using the entire vocabulary and opting for a variant of the Manhattan distance from the L^1 norm distance measure.

Such distance measures can also be applied with less frequent words. For example, Burrows (2007) proposed two distinct but complementary tests. The first one is based on words used regularly by one author but sporadically by the others while the second is grounded on words used infrequently by one author and ignored by the others. The remaining question is to know whether restricting the representation to the top MFW is effective or whether the entire vocabulary would produce better performance for the authorship linking problem. This question will be discussed later in this paper.

If words seem a natural way to generate a text surrogate, other studies have suggested using the letter occurrence frequencies (Ledger & Merriam, 1994; Kjell, 1994) or the distribution of short sequences of letters (character *n*-grams) (Juola, 2006). As demonstrated by Kešelj *et al*. (2003) such a representation can produce high performance levels. This approach can be justified, for example, by considering that an author employing the continuous present time form more frequently can be detected by a high frequency of the tri-gram "ing" and verbal forms related to the verb "to be" (e.g., "am", "is", "are"). As another example, one can identify more adverbial forms with a word ending in "ly". However, it is not clear which *n* value for the character *n*-gram is needed to achieve the highest performance level, and this value may depend on the collection, language, as well as other factors (e.g., text genre, OCR text) (McNamee & Mayfield, 2004).

Finally, the fingerprint of an author can be identified by the POS distribution. For example, one writer prefers using noun phrases more frequently than verb phrases implying more nouns and adjectives. For example, when comparing President Kennedy's and Obama's speeches, one can clearly see this difference, with Obama adopting more verbal

constructions, meaning a style oriented towards action ("yes, we can") (Savoy, 2017). Such text representations do not usually produce very high performance levels, but instead of considering only the distribution of single POS tags, short sequences of POS tags can be a more effective way of detecting some discriminative stylistic aspects of different authors.

3. Test Collections and Evaluation Methodology

As test collections for evaluating authorship linking algorithms, the PAN CLEF evaluation campaigns (Stamatatos *et al.*, 2016) have generated some corpora written in the English, Dutch, and Greek languages. However, only the training texts are currently available, not the full collection. Besides, all those texts are rather short (e.g., from 126 to 1,086 words on average in each text) corresponding to newspaper articles or online reviews.

To gain a better understanding of the advantages and drawbacks of various approaches, a test collection containing longer texts is required. To achieve this, the Oxquarry corpus was selected (Labbé, 2007). This corpus regroups 52 excerpts from novels written by nine distinct authors (e.g., 8 excerpts written by Conrad, 7 by Stevenson, 6 each by Morris and Orczy, etc.). As a constraint when generating this corpus, each author must appear with at least two texts. The mean size per document (in number of word-tokens) is 10,377. Similarly, the French corpus (Labbé & Labbé, 2006), called Brunet, contains 44 texts of novels written by eleven different well-known writers (e.g., Marivaux, Voltaire, Sand, Balzac, Zola, Proust, etc.). In this corpus, each author is represented with exactly four text passages extracted from two of their novels. Table 1 provides some statistics about these corpora and a more complete description can be found in the Appendix.

Table 1. Selected statistics about the test collections

Name	Language	# Texts	# Authors	Mean length	# Links
Oxquarry	EN	52	9	10,377	160
Brunet	FR	44	11	8,231	66
St Jean	FR	100	18	9,410	464

As a new test collection, the St Jean (Series A) corpus will be used. The entire corpus (Series A + Series B) will contain 200 text excerpts, but only the first part is used in our experiments. Like the Brunet corpus, it contains passages of novels written in French and published during the 19th century. In this corpus, one can find 13 excerpts from novels written by Balzac, 11 by Flaubert, 10 by Maupassant and Zola, and 6 by Dumas, Sand, Stendhal, and V. Hugo. As shown in Table 1, this last corpus contains more authors and documents than previous test collections. Moreover, to select each text excerpt, the author must be identified without any doubt, and the text must not contain any modifications or alterations. As a counter-example, one can mention the difficulty with several of Shakespeare's works (Ledger & Merriam, 1994; Michell, 1996; Craig & Kinney, 2009; Tassinari, 2009). For some works, the original text may have been modified, such as *Le Secret de Wilhelm Storitz* published in 1910 after the death of the author (Jules Verne in 1898), in a version modified by his son.

In the last column of Table 1, the number of correct links is indicated. In this context, a link establishes a relationship between two texts written by the same author. For example, with the Burnet corpus, each author wrote four texts. To regroup those four texts into a cluster, we can create (4 x 3) / 2 = 6 links. Having 11 authors, the number of correct links to resolve this problem is 6 x 11 = 66 links.

4. Evaluation Methodology

As proposed in the PAN CLEF campaigns, an authorship linking algorithm is evaluated with the AP (average precision), a measure well-known in different NLP domains (Manning *et al.*, 2008). The usual output is a ranked list (denoted *L*) of links between two texts. Each link indicates that the same author wrote the two texts. Preferably, each link also contains a numerical value indicating a degree of belief (or a probability) that the pair of texts was written by the same author. With a test collection, the entire set of true links (denoted *R*) is known. A passage of such a ranked list is depicted in Table 2. For example, the first row indicates that the system correctly establishes a link between Text #3 and #48 (both written by Stevenson, author name added with a posteriori knowledge) with a distance of 0.431 (computed by the Manhattan function).

Table 2. Excerpt of an output based on the Oxquarry corpus, Manhattan distance, Token-based representation

Rank	Distance	ID 1	Author	ID 2	Author
1	0.431	3	Stevenson	48	Stevenson
2	0.455	5	Stevenson	30	Stevenson
3	0.458	10	Stevenson	48	Stevenson
4	0.470	13	Stevenson	30	Stevenson
5	0.473	3	Stevenson	10	Stevenson
6	0.479	18	Morris	38	Morris
7	0.493	2	Morris	34	Morris
8	0.497	12	Orczy	50	Orczy
9	0.502	4	Butler	16	Butler
10	0.503	34	Morris	38	Morris
…	…	…	…	…	…
58	0.621	16	Butler	29	Hardy

Based on this notation, one can verify whether the link at the *i*th position (denoted l_i) belongs to the set *R*. If it is the case, the link is relevant, otherwise not, as defined by Equation 1. Based on this indicator function, one can define the precision up to a fixed rank. Equation 2 defines this performance value, and normally, the performance is provided at rank 10 (denoted Prec@10) or 20 (Prec@20). These two limits are used frequently in the information retrieval domain because they correspond to the first two pages of results returned by a commercial search engine. As one can see in Table 2, the Prec@10 = 1.0; all links up to the 10th rank are correct.

$$\text{relevant}(i) = \begin{cases} 1, & \text{if } l_i \in R \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

$$\text{precision}(k) = \left. \sum_{j=1}^{k} \text{relevant}(j) \right/ k \quad (2)$$

Another interesting limit is defined by |*R*|, the number of true links in the test collection. This limit varies from one test collection to another and it is denoted as R-precision (or RPrec). These values are indicated in the last column of Table 1 for our three corpora. One can compute this value by using Equation 2, with $k = |R|$.

For all these measurements, the best value is 1.0, which is achieved when all links are relevant, while the lowest value is 0.0 when no relevant link is found. Such a performance

measure provides a direct and simple interpretation. For example, when the precision after 10 links is 0.8, the final user knows that in the 10 first results, 80% are correct (or 8 over 10 links). As a main drawback of this measure one can mention that the rank is not considered. In this example, the two incorrect answers can appear in the first two positions or in the last two. In both cases, the performance measure Prec@10 is the same and equal to 0.8 although users would certainly prefer having the incorrect results in the bottom part of the ranked list instead of in the top positions.

The definition of AP given by Equation 3 provides a solution to this issue (Manning *et al.*, 2008). With this measure, the ranks are considered. Suppose that the output list computed by System A and B contains 4 links. With System A, all links are relevant, and therefore the AP is 1.0. With System B, only the link indicated in the first position is incorrect. Therefore, the AP is (0.0 + 0.5 + 0.666 + 0.75) / 4 = 0.479 indicating a relative change of more than 100% between the two rankings.

$$\text{AP} = \left(\sum_{j=1}^{|L|} \text{precision}(j) \text{ x } \text{relevant}(j)\right) \Big/ |R| \qquad (3)$$

With the AP, a simple interpretation is not possible. Even if this measure takes account for the ranks, it is sensitive to the first rank(s) as shown in our example. On the other hand, AP does not punish verbosity, i.e., every true link counts even when appearing near the end of the ranked list. Therefore, by providing all possible authorship links, one can attempt to maximize AP, without penalizing the Prec@10.

Overall, the AP and Prec@10 (Prec@20 and RPrec as well) are useful to compare two (or more) linking strategies. However, in some cases, it is important to return only good results and to specify "*I don't know*" when a link between two texts is not fully certain. Returning an answer that appears to be wrong generates a sentiment of insecurity for the final user with respect to the system, causing a lack of confidence, or engendering a percept that the computer is stupid. It is known in the PR domain (Public Relations) that a happy customer will talk to only 4 to 6 friends but a dissatisfied user will tell 9-15 people about their bad experience (Blackshaw, 2008). This phenomenon is relatively unknown in the academic world where the traditional performance measures tend to under-estimate the real "cost" of incorrect classifications. As a counter-example, one can cite the robust track at TREC (Voorhees & Harman, 2005) in which the focus is to penalize the retrieval of irrelevant items from a search engine more severely.

To measure the capability of a system to return only good results (or links in our context), one can measure its high precision (denoted HPrec) by indicating the rank-1 of the first incorrect answer appearing on the top of the returned list. For example, HPrec = 57 indicates that the first 57 results are correct before the first incorrect answer appears at rank 58, as it is the case in our example in Table 2.

5. Text Representations and Distance Measures

To solve the authorship linking problem, each text (or excerpt) must be represented in a way to closely reflect its stylistic aspects instead of the topics. In this perspective, language style is present as pervasive and frequent forms used by an author for mainly aesthetical value (Love, 2002; Biber & Conrad, 2009). Previous studies have found that the top *m* most frequent words (MFW) (with *m* = 50 to 500) tends to produce a high effectiveness (Burrows, 2002; Savoy, 2015). This set may or may not include punctuation symbols. Moreover, the

distinction between uppercase and lowercase letters is ignored, meaning all uppercase letters are transformed into their lowercase equivalent.

As a possible variant, one can consider only functional words, namely determiners, prepositions, pronouns, conjunctions, and modal verbs (or all closed POS categories). Moreover, to define those frequent words, a stemmer can be applied to remove inflectional suffixes (e.g., related to a variation in number, gender, or grammatical case). For the English language, the S-stemmer (Harman, 1991) applies three ordered rules to replace the plural form of a word with the corresponding singular form (e.g., the last rule is to remove the ending "-s" unless the word ends in "-ss" or "-us").

Instead of restricting the vocabulary to very frequent word-types, Labbé (2007) suggests considering the entire vocabulary. This solution is also adopted by Burrows (2007) who proposes to subdivide the vocabulary into three strata based on the term occurrence frequency.

In addition, an effective text representation can be generated in relation to the letter distribution, or letter *n*-gram (Kešelj *et al*., 2003). Typical values of *n* vary from 1 to 4 or 5, but higher values can also be considered (McNamee & Mayfield, 2004). Moreover, the POS distribution or a short sequence of such POS tags will be analyzed as other possible stylistic representations.

On the other hand, considering more words or character *n*-grams increases the complexity of the system and requires more processing time. Therefore, the words (or *n*-gram of characters) appearing only once (*hapax legomenon*) or twice (*dis legomenon*) can be ignored. This filtering decision can be justified to prevent overfitting to single occurrences. Moreover, due to the Zipf distribution of term occurrence frequencies, removing words appearing once or twice tends to reduce the vocabulary size by half.

The numerous distance measures can be regrouped under different families (Duda *et al*., 2001; Manning *et al*., 2008) where the most frequent one is the L^p family (or L^p norm). In this paradigm, the value of the parameter *p* determines different groups. To define the distance measure, uppercase letters will denote vectors (or points) while lowercase letters with a subscript indicate the value inside a vector. Thus, A or B specify vectors, while a_i indicates the element in the *i*th position of vector A, and *m* is the length of the vector.

To limit the investigations on the distance functions, a reduced set of functions has been selected due to their usefulness in a related task (Kocher & Savoy, 2017). First, with fixing $p = 1$, the Manhattan distance is obtained as defined in Equation 4. The underlying assumption is that the distance must be computed in proportion to the sum of the absolute differences for all dimensions. The distance value can be decomposed into contributions made by each dimension (or stylistic feature).

$$dist_{Manhattan}(A,B) = \sum_{i=1}^{m} |a_i - b_i| \quad (4)$$

Based on the L^1 norm (absolute difference), several variants of this distance measure have been proposed, such as the Tanimoto formula depicted in Equation 5. The value returned by the Manhattan distance is not normalized and it is sometimes difficult to figure out when a given distance is small or large. With the Tanimoto distance, a normalization factor is used corresponding to the sum of the maximum values of the coefficients.

$$dist_{\text{Tanimoto}}(A,B) = \sum_{i=1}^{m} |a_i - b_i| \Big/ \sum_{i=1}^{m} max(a_i, b_i) \quad (5)$$

Changing the value of p to 2, the Euclidean (L^2 norm) distance is obtained and represents a straight line between two points. This approach usually does not perform well and thus variants of the Euclidian distance have been suggested such as the Matusita formulation shown in Equation 6 or the Clark distance given by Equation 7.

$$dist_{Matusita}(A,B) = \sqrt{\sum_{i=1}^{m}\left(\sqrt{a_i}-\sqrt{b_i}\right)^2} \quad (6)$$

$$dist_{Clark}(A,B) = \sqrt{\sum_{i=1}^{m}\left(\frac{|a_i-b_i|}{a_i+b_i}\right)^2} \quad (7)$$

As another well-known family, different variants based on the inner product (or dot product) have been suggested. The main drawback of the inner product is the absence of a normalization. It is not clear when a distance value must be interpreted as large or small. Therefore, different variants have been proposed, and the most popular is certainly the Cosine similarity (Eq. 8) which can be transformed into a distance value between 0 and 1 (Eq. 9) (Manning *et al*., 2008).

$$sim_{Cosine}(A,B) = \sum_{i=1}^{m} a_i b_i \Big/ \sqrt{\sum_{i=1}^{m} a_i^2}\sqrt{\sum_{i=1}^{m} b_i^2} \quad (8)$$

$$dist_{Cosine}(A,B) = cos^{-1}(sim_{Cosine}(A,B))/\pi \quad (9)$$

Shannon's concept of entropy (Manning *et al*., 2008) is also a main source of a family of distance measures. The Jeffrey divergence (denoted JDivergence) computes the difference between two probability distributions (see Eq. 10). In this case, all values a_i of each vector must be non-negative and they must sum up to 1. Moreover, the basis of the logarithm is fixed to two in Shannon's entropy measure. However, in the author profiling context, or when only the ranking of the different categories is relevant, changing the basis of the logarithm doesn't affect the ordering of the answers. As for other distance measures, a larger distance value indicates a larger difference between the writing style of the two authors (or points).

$$dist_{JDivergence}(A,B) = \sum_{i=1}^{m}(a_i-b_i)\,log\left(a_i/b_i\right) \quad (10)$$

6. Evaluation of Word-Based Text Representations

To compare different text representations, our experiments start by using all word-types with the six distance measures described previously. In the top part of Table 3, the word-tokens have been used for the three corpora (e.g., the label "T-Manhattan" indicates a text surrogate generated with word-tokens and a distance computed with the Manhattan measure).

In the bottom part, the text representations are built based on the lemmas (or the dictionary entries, denoted "L-Manhattan"). Mainly, in the English language, the difference between these two forms can be small (e.g., houses vs. house, running vs. run). For the French language, however, one can expect a larger difference due to a richer inflectional morphology (e.g., aimerais vs. aimer (to love), blanches vs. blanc (white)).

As performance measures, the average precision (AP) and R-precision (RPrec) have been reported, with the higher the value, the better the effectiveness. To reflect the quality of a

text representation and distance measure to return only good answers, the high precision (HPrec) value is also reported.

From data shown in Table 3 under the English corpus (Oxquarry), one can see that the AP values are, in mean, 8% higher for the lemma-based representation than for the tokens. The highest values (always depicted in bold) are however similar in both cases. The situation is similar for the French Brunet collection, with a mean AP difference of 3.3% in favor of the lemmas. The last corpus (St Jean) indicates a better AP performance when applying tokens (on average, 5.3%). When considering the high precision values (HPrec), usually the lemma-based representations tend to produce better answers, but for the Brunet corpus, the value 23 achieved with the Clark function using tokens is clearly an exception, compared to 15 obtained with lemmas (both values shown in italics in Table 3).

Table 3. Evaluation over two word-based text representations and six distance measures

Distance and text	Oxquarry			Brunet			St Jean		
representation	AP	RPrec	HPrec	AP	RPrec	HPrec	AP	RPrec	HPrec
T-Manhattan	0.588	0.525	57	0.648	0.561	**26**	**0.666**	**0.585**	64
T-Tanimoto	0.620	0.556	59	0.653	0.561	**26**	0.663	0.573	**65**
T-Matusita	0.561	0.519	51	0.569	0.485	16	0.504	0.470	44
T-Clark	**0.731**	**0.650**	**70**	0.603	0.515	*23*	0.533	0.512	47
T-Cosine	0.511	0.500	28	0.590	0.561	15	0.648	0.570	52
T-JDivergence	0.595	0.544	67	**0.659**	**0.636**	19	0.608	0.542	56
L-Manhattan	0.643	0.556	63	0.662	**0.606**	**25**	**0.652**	**0.576**	**68**
L-Tanimoto	0.685	0.563	66	**0.675**	**0.606**	**25**	0.651	0.573	**68**
L-Matusita	0.611	0.538	53	0.565	0.530	15	0.489	0.461	53
L-Clark	**0.737**	**0.644**	68	0.558	0.500	*15*	0.452	0.421	48
L-Cosine	0.553	0.500	38	0.568	0.545	15	0.589	0.542	30
L-JDivergence	0.613	0.538	**71**	0.656	0.636	20	0.603	0.536	62

When analyzing the variations related to the distance measures, one can see that none of them performs the best in all cases. For the Oxquarry corpus, the Clark measure (L^2 family) produces the best effectiveness while for the St Jean collection, the highest precisions are achieved with the Manhattan distance (L^1 family). For the Brunet corpus, the Jeffrey divergence offers the best precision values for one text surrogate (token-based) while Tanimoto is a better choice for the second (lemma-based). However, in all these experiments, the Cosine distance never produces the best answer. In mean, and compared to the best AP solution, the performance of the Cosine function is 9.5% lower with the token-based representation and 16% worse with the lemmas. Overall, and considering the two text representations, the Matusita distance offers the lowest AP values. Finally, one can see that the results achieved by Manhattan and Tanimoto distance are correlated.

When analyzing the ranked lists for the English corpus, we found that correctly linked texts appearing on the top are novels written by Stevenson (*Catriona*, *The Master of Ballantrae*), Morris (*News from Nowhere*), or Hardy (*Well-beloved, Jude the Obscure*). Determining the specific functional terms of those authors (Savoy, 2016), we found that Stevenson uses more frequently the words *I, my, me, ye, myself,* and the comma. With Morris, the most specific words are: *thou, shall, we, three, and, our,* and the comma while Hardy's characteristic terms are: *her, she, had, till, being,* and the quote. The other writers tend to share more specific terms in common such as the full stop between Conrad, Orczy, and Butler, the determiner *the* appearing with both Chesterton and Conrad, or the pronoun *it* belonging to the favorite terms of Tressel and Chesterton.

For the French corpus St Jean, correct links appearing in the top of the ranked lists connect novels written by Zola (*L'assomoir, La Fortune des Rougon*), Flaubert (*Mme Bovary, Bouvard et Péruchet*), or Maupassant (*Mont-Oriol, Bel-Ami*). In this case, the specific terms associated with Zola are *ça*, *avait*, *elle*, *aurait*, and *était* (it, had, she, he, would have, was) while Flaubert uses more frequently *des* (of), *ils* (they), *les* (the), the exclamation mark, and the semicolon. Finally, Maupassant can be discriminated among the others with the following terms: the colon, *il*, *puis*, *elle*, and *et* (he, then, she, and).

7. Evaluating POS-Based Text Representations

As another text representation, one can consider the POS distribution. A closer look reveals that the returned information of the tagger (Stanford POS tagger for the English language, (Toutanova *et al*., 2003) and Labbé's POS tagger for French (Labbé, 2007)) contains not only the POS category (e.g., verb, noun, pronoun) but also some morphological information (e.g., personal pronoun, 3rd person, plural). The punctuation symbols are also included as additional tags. In Table 4a, the performance with text surrogate generated with those POS tags are presented in the top part (e.g., "P-Manhattan"), while in the middle part ("G-Manhattan") only the grammatical categories (e.g., verb, pronoun, adverb) and punctuations have been used to build text representations.

Table 4a. Evaluation over three text representations and six distance measures

Distance and text representation	Oxquarry			Brunet			St Jean		
	AP	RPrec	HPrec	AP	RPrec	HPrec	AP	RPrec	HPrec
P-Manhattan	0.494	0.425	26	0.482	0.470	10	0.563	**0.555**	**53**
P-Tanimoto	0.505	0.444	24	0.494	0.485	10	0.508	0.494	26
P-Matusita	**0.517**	**0.463**	**36**	**0.497**	0.485	**12**	0.448	0.448	20
P-Clark	0.419	0.431	4	0.301	0.318	3	0.456	0.455	24
P-Cosine	0.465	0.406	25	0.474	0.485	11	0.448	0.448	20
P-JDivergence	0.513	**0.463**	**36**	0.495	**0.500**	**12**	**0.555**	0.491	41
G-Manhattan	0.470	0.406	29	0.435	**0.424**	5	**0.406**	**0.418**	**14**
G-Tanimoto	0.471	0.406	**30**	0.438	**0.424**	5	0.360	0.367	6
G-Matusita	**0.489**	**0.456**	27	**0.462**	0.439	11	**0.406**	0.415	13
G-Clark	0.410	0.431	4	0.216	0.182	4	0.248	0.291	9
G-Cosine	0.442	0.388	23	0.440	**0.424**	7	0.360	0.367	8
G-JDivergence	0.482	**0.456**	**30**	**0.462**	**0.424**	**12**	0.401	0.406	**14**
N-Manhattan	0.308	0.306	**6**	0.230	0.318	0	0.237	0.285	0
N-Tanimoto	0.316	0.338	**6**	0.233	0.333	0	0.238	0.285	0
N-Matusita	0.324	**0.344**	2	0.281	0.364	0	0.239	0.306	0
N-Clark	0.189	0.194	0	0.120	0.136	0	0.121	0.139	0
N-Cosine	0.314	0.319	5	0.230	0.273	0	0.226	0.270	0
N-JDivergence	**0.329**	**0.344**	4	**0.286**	**0.379**	0	**0.244**	**0.312**	**5**

Taking account of the POS tags (with the associated morphological information) produces a better text representation than only the grammatical categories. Comparing the two models, the AP measure depicts, in mean, a 5% difference with the Oxquarry corpus, 11.7% with the Brunet, and 26.6% with the St Jean corpus. Based on the HPrec values, this relative change is higher, up to 62.3% with the St Jean corpus. Finally, the POS text representation produces lower effectiveness levels than either the lemma- or the token-based models (see Table 3).

Compared to this last one, the mean relative change in AP is 18% for the Oxquarry, 26.2% with the Brunet, and 17.3% with the St Jean collection.

Concerning the distance measures, Table 4a indicates that the Matusita distance usually produces the best AP results with the three corpora with the single exception being the performance of the Manhattan function for the St Jean collection (0.563 vs. 0.448).

Finally, in the bottom part of Table 4a, the text representation is based on the distribution of the token length (e.g., "N-Manhattan"). This surrogate is not limited to a single value, i.e., the mean token size, but presents all possible token lengths with their occurrence frequency. The performance reported in Table 4a clearly indicates that such an approach is not a pertinent representation. Moreover, the HPrec value is often zero, indicating that even the first link is wrong.

Table 4b. Evaluation of short sequences of POS tags

Distance and text representation	Oxquarry			Brunet			St Jean		
	AP	RPrec	HPrec	AP	RPrec	HPrec	AP	RPrec	HPrec
P-Tanimoto	0.505	0.444	24	0.494	0.485	10	0.508	0.494	26
P2-Tanimoto	0.554	0.475	37	0.612	0.576	14	0.671	0.618	56
P3-Tanimoto	**0.555**	**0.500**	30	**0.663**	**0.621**	**18**	**0.712**	**0.645**	**59**
P4-Tanimoto	0.524	0.450	**39**	0.616	0.545	9	0.619	0.567	36
P-Matusita	0.517	0.463	36	0.497	0.485	12	0.448	0.448	20
P2-Matusita	**0.553**	**0.481**	**55**	0.631	0.561	**22**	0.691	**0.630**	**80**
P3-Matusita	0.529	**0.481**	38	**0.661**	**0.621**	18	**0.699**	0.609	68
P4-Matusita	0.490	0.419	37	0.561	0.515	13	0.531	0.482	37

As the number of distinct POS tags (42 for the English language, 29 for the French) is rather limited compared to the vocabulary size, a text representation can be built using short sequences of such tags. Considering only two distance measures, Table 4b reports the evaluations of these text surrogates generated form sequences of two to four POS tags. Compared to the baselines ("P-Tanimoto" or "P-Matusita" repeated from Table 4a) corresponding to single POS tags, sequences of two or three tags improves the result significantly. For example, with the Brunet corpus and using the Matusita function, the AP raises from 0.497 to 0.661 (+33%). The best performance depicted in Table 4b is usually below those achieved based on word-based representation (see Table 3). In some cases, however, the difference is rather small, i.e., with the Brunet corpus and Tanimoto function, 0.653 for token-based vs. 0.663 for sequences of three POS tags, corresponding to a relative change of -1.3%.

8. Letter N-Gram Evaluation

As another text representation, one can select short sequences of letters, denoted *n*-grams, extracted from the text. In this generation process, a few variants are possible. Each word boundary may stop the creation of the *n*-grams. The distinction between uppercase and lowercase could be preserved, and the adjacent *n*-grams could overlap. In our experiments, the word boundary does not stop the *n*-grams generation. All punctuation symbols are replaced by a space and the uppercase letters are replaced by their corresponding lowercase. As an example, based on the sentence "Paul's book is red.", the following overlapping 4-grams are extracted: "_pau", "paul", "aul_", "ul_s", "l_s_" "_s_b", …, "s_red", "_red", "red_" where "_" indicates a space.

As possible values for *n*, one can consider any value between one and ten. However, after *n* = 5, 6, or 7, the number of generated *n*-grams becomes huge, and most them have a very low occurrence frequency. Table 5 reports the performance obtained with the Tanimoto (L^1 norm) and Matusita (L^2) distance for *n* = 3 to 7. Before that, the first line for each measure indicates the performance achieved with a token-based representation and then the label "1/2" indicates a combined text representation based on both uni- and bigrams as suggested by Kjell (1994) and Goldberg (2017).

As depicted in Table 5, the best *n* value depends on the collection, but values larger than or equal to five tend to produce the highest performance (e.g., for the Oxquarry corpus, *n* = 7 with the Tanimoto distance, *n* = 5 with Matusita). Comparing across corpora or distance measures, slightly modifying the value *n* tends to produce similar results, e.g., Oxquarry with Tanimoto function gives an AP of 0.872 with *n* = 5 vs. 0.888 with *n* = 7 (+1.8%).

Table 5. Evaluation over six different *n*-gram text representations

n-gram length	Oxquarry			Brunet			St Jean		
	AP	RPrec	HPrec	AP	RPrec	HPrec	AP	RPrec	HPrec
Token-Tanimoto	0.620	0.556	59	0.653	0.561	**26**	**0.663**	**0.573**	65
1/2-Tanimoto	0.654	0.613	52	0.614	0.561	23	0.549	0.491	52
3 Tanimoto	0.817	0.738	61	0.641	0.576	20	0.641	0.576	20
4 Tanimoto	0.854	0.788	82	0.655	0.606	20	0.609	0.545	65
5 Tanimoto	0.872	0.806	99	0.670	0.606	21	0.622	0.548	**73**
6 Tanimoto	0.883	0.813	**101**	0.676	0.606	16	0.631	0.545	71
7 Tanimoto	**0.888**	**0.825**	99	**0.680**	**0.621**	20	0.624	0.539	52
Token-Matusita	0.561	0.519	51	0.569	0.485	16	0.504	0.470	44
1/2-Matusita	0.587	0.538	55	0.627	0.591	18	0.532	0.479	55
3 Matusita	0.828	0.731	65	0.638	0.606	18	**0.638**	**0.606**	18
4 Matusita	0.888	**0.825**	94	0.658	**0.621**	20	0.534	0.448	**68**
5 Matusita	**0.892**	**0.825**	**96**	**0.667**	0.606	**22**	0.539	0.476	58
6 Matusita	0.883	0.819	88	0.664	0.576	21	0.556	0.494	57
7 Matusita	0.876	0.775	88	0.660	0.576	20	0.556	0.500	38

As depicted in Table 5, the best value of *n* depends on the collection, but the difference between the three corpora or distance functions is just ±1 (e.g., for the Oxquarry corpus, *n* = 6 with the Tanimoto distance, *n* = 5 with the Matusita, *n* = 6 for Brunet corpus). Compared to the token-based representation, the *n*-gram approach tends to produce a higher effectiveness. With the English corpus, the improvement is significant. For example, with the Tanimoto function, the AP increases from 0.620 to 0.883 (+42.4%), and with the Matusita distance, from 0.561 to 0.892 (+59%). With the Brunet corpus and applying the Tanimoto distance, the performance difference is smaller, but still present, e.g., the AP varies from 0.653 to 0.680 (+4.1%), or from 0.569 to 0.667 (+17.2%) with the Matusita function. With the St Jean corpus, the *n*-gram approach improves the performance only with the Matusita measure.

9. Efficiency Improvement

In the previous sections, text representations were built considering the entire vocabulary or all possible *n*-grams. Ranking the terms (word-types or *n*-grams) in proportion to their occurrence frequency, a Zipfian distribution can be observed. If the most frequent ones cover a large proportion of all texts, the terms appearing only once or twice tend to correspond to

50% of all word-types, and usually a larger percentage when considering character *n*-grams. Moreover, assigning a text to an author based on a few words occurring only once is an unsafe decision and prone to impostors (a writer can easily pass for another).

To reduce the text representation, various pruning strategies can be applied. To assess their effects, Table 6 reports in the top part the mean number of word-tokens (labeled "Token") and the mean vocabulary size (|V|) per document. The first row indicates the mean values before any pruning procedure ("All tokens"). For the Brunet corpus, the averages are 10,628 tokens per document and 2,204 word-types in each document. In the next four rows, the terms appearing once ("*tf* > 1") to four times ("*tf* > 4") in a document representation are eliminated. The representation size decreases slowly, for example in the Oxquarry collection, from 11,650 tokens to 10,351 when ignoring terms appearing once, or to 8,934 when only keeping terms appearing at least five times. On the other hand, the mean vocabulary size per document decreases faster. With the Oxquarry corpus, the mean number of distinct terms begins with 2,169 and decreases to 314 when removing all terms having a term frequency smaller than or equal to 4.

In the middle of Table 6, we report the mean number of tokens per document when using only the 50 to 1,000 most frequent word-tokens (MFW) when generating the text surrogates. These word lists were defined in relation to the entire corpus. Reducing the vocabulary to the top 50 MFW, the text representation size is reduced, in mean, by 50%, e.g., with the Oxquarry collection, from 11,650 to 5,840 tokens. Looking at the vocabulary, the reduction is more severe. For example, with the Oxquarry collection, the mean vocabulary/document decreases from 2,169 to 48 (-97.8%).

The bottom part of Table 6 shows the statistics when considering letter *n*-grams with $n = 6$. For the English corpus, the most frequent 6-gram is "_that_", for the Brunet corpus it's "_vous_" (you/plural), and "_elle_" (she/singular) appears the most in the St Jean collection.

Table 6. Statistics of different pruning strategies

Pruning strategy	Oxquarry		Brunet		St Jean	
	Token	|V|	Token	|V|	Token	|V|
All tokens	11,650	2,169	10,628	2,204	12,331	2,466
tf > 1	10,351	871	9,183	759	10,711	845
tf > 2	9,688	539	8,550	443	10,005	492
tf > 3	9,254	394	8,161	313	9,576	350
tf > 4	8,934	314	7,883	244	9,272	274
50 tokens	5,840	48	5,664	50	6,654	50
100 tokens	6,886	96	6,588	99	7,694	94
200 tokens	7,817	194	7,236	193	8,422	194
300 tokens	8,317	282	7,593	274	8,843	286
500 tokens	8,862	431	8,027	408	9,308	430
1,000 tokens	9,560	710	8,562	642	9,913	691
6-gram (all)	52,409	26,003	44,302	21,724	50,800	24,187
tf > 1	34,657	8,251	29,868	7,289	34,884	8,271
tf > 2	26,621	4,233	22,933	3,822	27,145	4,401
tf > 3	21,800	2,626	18,561	2,364	22,242	2,767
tf > 4	18,525	1,807	15,554	1,613	18,821	1,912

The main concern with the *n*-gram model is the huge number of distinct *n*-grams that can be generated. With the St Jean corpus, the mean number of terms in a text representation goes

from 2,466 word-types to 24,187 6-grams (around 10 times more). Here too, the pruning of terms appearing less than twice is useful to reduce the complexity of the text representation, as for example, with the St Jean corpus, the size decreases from 24,187 to 4,401 6-grams appearing more than twice (-81.8% in relative value), or to 1,912 6-grams occurring at least five times (-92.1%).

Pruning text representations by ignoring features with a very low occurrence frequency may reduce the complexity of text representation. However, such procedures may hurt the overall success. To verify this aspect, Table 7 reports the three performance measures using two distance functions and word-based text representation. In the row labeled "Tok-Tanimoto" (and "Tok-Matusita"), the performance obtained with all tokens are depicted as a baseline.

As a general trend, one can observe that removing very low frequency word-types might even increase the performance. For example, comparing the baseline with the row labeled "*tf* > 2", the AP value increases for the Brunet and St Jean corpora for both distance measures. With the St Jean corpus and the Matusita distance the performance goes from 0.504 to 0.604 (+19.8%). Conversely, with the Oxquarry and the Tanimoto distance, a slight decrease can be seen (from 0.620 to 0.616, -0.6%). This pruning strategy reduces the vocabulary from slightly more than 2,000 word-types to 443 (Brunet) or 539 (Oxquarry) as shown in Table 6.

Table 7. Evaluation of different pruning strategies on word-based representation

Pruning strategy	Oxquarry			Brunet			St Jean		
	AP	RPrec	HPrec	AP	RPrec	HPrec	AP	RPrec	HPrec
Tok-Tanimoto	**0.620**	0.556	59	0.653	0.561	**26**	0.663	0.573	65
tf > 1 Tanimoto	0.620	0.550	63	0.661	0.606	23	0.702	0.627	58
tf > 2 Tanimoto	0.616	0.531	**64**	0.661	0.606	23	0.701	0.621	49
tf > 3 Tanimoto	0.613	0.538	62	0.657	0.636	22	0.542	0.606	53
tf > 4 Tanimoto	0.611	0.525	59	0.657	0.636	22	0.692	0.606	52
50-Tanimoto	0.533	0.533	46	0.637	0.591	19	0.711	0.639	**70**
100-Tanimoto	0.562	0.519	46	0.632	0.591	21	0.718	0.655	65
200-Tanimoto	0.580	0.519	48	0.646	0.576	20	0.725	0.667	63
300-Tanimoto	0.600	0.531	48	0.653	0.591	21	0.736	0.676	65
500-Tanimoto	0.613	**0.613**	50	0.665	0.636	23	**0.751**	**0.679**	63
1,000-Tanimoto	0.628	0.556	61	**0.676**	**0.652**	23	0.750	0.676	60
Tok-Matusita	0.561	0.519	51	0.569	0.485	16	0.504	0.470	44
tf > 1 Matusita	0.575	0.506	55	0.619	0.545	**25**	0.591	0.527	60
tf > 2 Matusita	0.571	0.488	55	0.648	0.591	24	0.604	0.542	51
tf > 3 Matusita	0.575	0.494	55	0.652	0.576	21	0.605	0.542	58
tf > 4 Matusita	0.577	**0.577**	51	0.652	0.606	22	0.595	0.530	58
50-Matusita	0.521	0.475	49	0.627	0.545	23	0.700	0.621	**84**
100-Matusita	0.554	0.500	51	0.608	0.530	17	0.692	0.621	73
200-Matusita	0.573	0.500	58	0.607	0.530	17	0.712	0.652	70
300-Matusita	0.597	0.538	54	0.626	0.545	17	**0.733**	0.667	67
500-Matusita	0.614	0.556	69	0.663	0.652	20	0.751	**0.676**	70
1,000-Matusita	**0.632**	0.563	**74**	**0.674**	**0.667**	20	0.720	0.639	77

As another example, one can analyze the row labeled "500-Matusita" where the 500 most frequent word-types are defined in relation to the whole vocabulary. As can be seen in the data depicted in Table 6, such a pruning scheme tends to reduce the mean vocabulary size per document in the range of 408 (-81.5% for Brunet) to 431 (-80.1% for the Oxquarry). The results achieved with this strategy generally indicates an improvement over the baseline

performance. Considering the AP values, the increase is around +9.4% (from 0.561 to 0.614) with the Oxquarry corpus using the Matusita distance. After the pruning stage, spurious features and especially words with single occurrences have no longer an influence on the distance calculation and are therefore ignored to create the authorship links.

To obtain an overview of the time required to compute the different text representations, Table 8 reports the elapsed time in seconds (a mean based on four runs with both the Tanimoto and Matusita distance functions). The first row (labeled "Token") corresponds to a token-based surrogate built with the entire vocabulary while the second (labeled "1,000 MFW") signals the value when considering only the 1,000 most frequent tokens. The last three rows report the time needed when considering the *n*-gram models with different values for *n*.

Table 8. Elapsed time in sec. for different text representations

Text representation	Oxquarry	Brunet	St Jean
Token	297	224	1,387
1,000 MFW	49	31	106
3-grams	235	128	669
4-grams	3,381	1,594	9,603
5-grams	16,063	7,336	46,699
6-grams	41,449	19,537	127,868

When time is a critical resource, adopting a pruning scheme based on the *k* most frequent tokens must be viewed as an effective approach. It can be from 7 times (Brunet corpus) to 13 times (St Jean) faster than taking account of the entire vocabulary. Moreover, such an approach is possibly more effective (see Table 7). On the other hand, adopting an effective *n*-gram model ($n \geq 5$ as depicted in Table 5) implies a larger processing time as indicated in the last rows of Table 8 due to the huge number of generated *n*-grams.

10. Conclusion

The authorship linking problem raises new challenges, and one of them is the absence of a training phase useful in determining the most effective feature set and distance measures. In this unsupervised context, our study evaluates the effectiveness of six different distance functions using three test collections. Moreover, the main findings are based on two different languages (English and French) with relatively long text excerpts (from 8,231 to 10,377 tokens/document).

None of the selected distance functions performs the best in all cases. From the L^1 norm, the Tanimoto, strongly correlated to the Manhattan function, usually produces high performance levels (see Table 3), at least for the two French collections. In the L^2 family, the Matusita function performs well with some text representations (see Table 4) while the Clark distance produces better answers with token-based representation (see Table 3). In some cases, the Jeffrey divergence might produce a high performance. However, in all cases, the Cosine distance function, frequently used in various applications (Goldberg, 2017), does not perform very well.

As text representation, the word-tokens or the lemmas (dictionary entries) correspond to well-known approaches. Using lemma-based representation requires that an additional morphological analysis be performed. The results of our experiments indicate that lemmas

tend to be more useful (see Table 3). While this conclusion is valid for the English corpus, the two French corpora indicate contradictory findings. With the Brunet collection, lemmas perform better than tokens, but with the St Jean corpus, it is the reverse. The performance differences are however not substantial, i.e., +3.3% with the Brunet corpus and -5.3% in the St Jean collection.

As another text representation that can extract stylistic features, POS tags (grammatical category with morphological information together with the punctuation symbols) can be applied. Compared to the word-based representation (token or lemma), the AP tends to decrease around 20%. Limiting the representation to single grammatical categories (see Table 4a) lowers the results significantly more with an average decrease of 25% compared to word-based with the English collection and over 30% with the two French corpora. Thus, single POS tags do not effectively discriminate the stylistic differences between authors. However, a short sequence of two or three POS tags is clearly more effective. Such text representations can even be more effective than a word-based model. For the AP and considering sequence of two POS tags, the mean improvement is around 8% for the English corpus and 25% for the Brunet collection. Working with sequences of four or more POS tags is not effective in all collections.

As a third paradigm to generate a text representation, character *n*-grams can be used. As shown in Table 5, the best value for *n* depends on the collection, but values larger than or equal to 5 tend to produce the best answers. Compared to token-based models, the *n*-grams may perform significantly better, for example, with the Oxquarry corpus using the Tanimoto distance, the average improvement is 31.6%, and with the Matusita function, 45.4%. For the Brunet collection, this enhancement is smaller as we can observe in mean +14.4% with the Matusita distance with a favor for the *n*-gram model but roughly the same precision values using the Tanimoto function.

For efficiency reasons, one can apply a pruning procedure to reduce the vocabulary size by ignoring terms appearing once or twice. This culling procedure reduces the size of the number of word-types by 50%, and around 80% for the letter 6-grams (see Table 6). Such a pruning scheme can significantly reduce the complexity of text representations based on character *n*-grams with a value of *n* larger than 4 or 5. Moreover, and as shown in Table 7, the success rate is usually higher after the pruning than before. As an alternative, reducing the word-types to the 500 most frequent word-types (MFW) is still a pertinent strategy allowing better performance than considering the entire vocabulary (see Table 7).

There are various ways to extend the current study and to deepen the acquired knowledge from a ranked list of authorship links. Since the proposed methods are based on a reduced set of features, an interpretation of the results can be beneficial for the final user. We could extract information about why (and why not) the highest (and lowest) pairs of texts have a shared authorship. Furthermore, while this study is focused on authorship linking, the experience obtained can be transferred to other domains and could be used to improve the performance of author clustering approaches.

Acknowledgments

This research was supported, in part, by the NSF under Grant #200021_149665/1.

References

Almishari, M., & Tsudik, G. (2012). Exploring Linkability of User Reviews. *Proceedings Computer Security ESORICS*, 307-324.

Argamon, S., Koppel, M., Pennebaker, J.W., & Schler, J. 2009. Automatically Profiling the Author of an Anonymous Text. *Communications of the ACM*, 52(2), 119-123.

Baayen, H.R. 2008. *Analysis Linguistic Data: A Practical Introduction to Statistics Using R*. Cambridge: Cambridge University Press.

Biber, D., & Conrad, S. 2009. *Register, Genre, and Style*. Cambridge: Cambridge University Press.

Binonga, J.N.G., & Smith, M.W. (1999). The Application of Principal Component Analysis to Stylometry. *Literary and Linguistic Computing.* 14(4), 445-465.

Blackshaw, P. (2008). *Satisfied Customers Tell Three Friends, Angry Customers Tell 3,000.* Crown Business

Burrows, J.F. (1992). Not Unless you Ask Nicely: The Interpretative Nexus Between Analysis and Information. *Literary and Linguistic Computing*, 7(1), 91-109.

Burrows J.F. 2002. Delta: A Measure of Stylistic Difference and a Guide to Likely Authorship. *Literary and Linguistic Computing*, 17(3), 267-287

Burrows, J. 2007. All the Way Through: Testing for Authorship in Different Frequency Strata. *Literary and Linguistic Computing*. 22(1), 27-47.

Cortelazzo, M.A., Nadalutti, P., Ondelli, S., & Tuzzi, A. 2016. Authorship Attribution and Text Clustering for Contemporary Italian Novels. *Proceedings Qualico 2017*, 7-8.

Craig, H., & Kinney, A.F. 2009. *Shakespeare, Computers, and the Mystery of Authorship*. Cambridge: Cambridge University Press.

Duda, R.O., Hart, P.E., & Stork, D.G. (2001). *Pattern Classification.* New York: Addison-Wesley.

Goldberg, Y. 2017. *Neural Network Methods for Natural Language Processing.* Morgan & Claypool Publ.: San Rafael, CA.

Goodfellow, I., Bengio, Y., & Courville, A. 2016. *Deep Learning.* The MIT University Press: Cambridge.

Harman, D. 1991. How Effective is Suffixing? *Journal of the American Society for Information Science*, 42(1), 7-15.

Holmes, D.I. 1998. The Evolution of Stylometry in Humanities Scholarship. *Literary and Linguistic Computing*, 13(3), 111-117.

Hughes, J.M., Foti, N.J., Krakauer, D.C., & Rockmore, D.N. 2012. Quantitative Patterns of Stylistic Influence in the Evolution of Literature. *Proceedings of the PNAS*, 109(20), pp. 7682-7686.

Jockers, M.L., & Witten, D.M. 2010. A Comparative Study of Machine Learning Methods for Authorship Attribution. *Literary and Linguistic Computing*. 25(2), 215-223.

Juola, P. 2006. Authorship Attribution. *Foundations and Trends in Information Retrieval*, 1(3).

Kešelj, V., Peng, F., Cercone, N., & Thomas C. (2003). N-Gram-Based Author Profiles for Authorship Attribution. In *Proceedings of the Conference Pacific Association for Computational Linguistics, PACLING'03*, Halifax, 255-264.

Kjell, B. 1994. Authorship Determination Using Letter Pair Frequency Features with Neural Network Classifier. *Literary and Linguistics Computing*, 9(2), 119-124.

Kocher, M., & Savoy, J. 2017. Distance Measures in Author Profiling. *Information Processing and Management*, 53(5), 1103-1119

Koppel, M., Schler, J., & Bonchek-Dokow, E. 2007. Measuring Differentiability: Unmasking Pseudonymous Authors. *Journal of Machine Learning research*, 8(6), 1261-1276.

Labbé, D., & Labbé, C. 2006. A Tool for Literary Studies. *Literary & Linguistic Computing*, 21(3), 311-326.

Labbé, D. 2007. Experiments on Authorship Attribution by Intertextual Distance in English. *Journal of Quantitative Linguistics*, 14(1), 33-80.

Ledger, G., & Merriam, R. (1994). Shakespeare, Fletcher, and the *Two Noble Kinsmen*. *Literary and Linguistic Computing*, 9(3), 235-248.

Love, H. 2002. *Attributing Authorship: An Introduction.* Cambridge University Press: Cambridge.

Manning, C.D., Raghavan, P., & Schütze, H. 2008. *Introduction to Information Retrieval.* Cambridge University Press: Cambridge.

Michell, J. 1996. *Who Wrote Shakespeare?* Thames and Hudson: New York (NY).

McNamee, P., & Mayfield, J. 2004. Character N-Gram Tokenization for European Language Text Retrieval. *Information Retrieval Journal*, 7(1/2), 73-98.

Olsson, J. 2008. *Forensic Linguistics.* Continuum, London.

Pennebaker, J.W. 2011. *The Secret Life of Pronouns. What our Words Say about us*. Bloomsbury Press: New York.

Rangel, F., & Rosso, P. 2016. On the Impact of Emotions on Author Profiling. *Information Processing & Management,* 52(1), 73-92

Savoy, J. 2015. Comparative Evaluation of Term Selection Functions for Authorship Attribution. *Digital Scholarship in the Humanities*, 30(2), 246-261.

Savoy, J. 2016. Text Representation Strategies: An Example with the *State of the Union* Addresses. *Journal of the American Society for Information Science & Technology*, 67(8), 1858-1870.

Savoy, J. 2017. Analysis of the Style and the Rhetoric of the American Presidents Over Two Centuries. *Glottometrics*, 38, 55-76.

Stamatatos, E. 2009. A Survey of Modern Authorship Attribution Methods. *Journal of the American Society for Information Science & Technology*, 60(3), 433-214.

Stamatatos, E., Tschuggnall, M., Verhoeven, B., Daelemans, W., Specht, G., Stein, B. & Potthast, M. 2016. Clustering by Authorship Within and Across Documents. *Notebook Papers of CLEF 2016 Labs and Workshop*. CEUR: Aachen.

Tassinari, L. 2009. *John Florio, the Man who was Shakespeare* Giano Books: New York (NY).

Toutanova, K., Klein, D., Manning, C., & Singer, Y. (2003). Feature-rich Part-of-Speech Tagging with a Cyclid Dependency Network. *Proceedings of HLT-NAACL 2003*, 252-259.

Voorhees, H., & Harman, D. 2005. *The TREC Experiment and Evaluation in Information Retreival.* The MIT University Press: Cambridge.

Zhao, Y., & Zobel, J. 2007. Searching with Style: Authorship Attribution in Classic Literature. Proceedings of the *Thirtieth Australasian Computer Science Conference (ACSC2007)*, Ballarat, 59-68.

Appendix

Table A.1. List of 52 text excerpts from the Oxquarry corpus

#	Author	Short title	#	Author	Short title
A1	Hardy	Jude	A2	Butler	Erewhon
B1	Butler	Erewhon	B2	Morris	Dream of JB
C1	Morris	News	C2	Tressel	Ragged TP
D1	Stevenson	Catriona	D2	Hardy	Jude
E1	Butler	Erewhon	E2	Stevenson	Ballantrae
F1	Stevenson	Ballantrae	F2	Hardy	Wessex Tales
G1	Conrad	Lord Jim	G2	Orczy	Elusive P
H1	Hardy	Madding	H2	Conrad	Lord Jim
I1	Orczy	Scarlet P	I2	Morris	News
J1	Morris	Dream of JB	J2	Hardy	Well-beloved
K1	Stevenson	Catriona	K2	Conrad	Almayer
L1	Hardy	Jude	L2	Hardy	Well-beloved
M1	Orczy	Scarlet P	M2	Morris	News
N1	Stevenson	Ballantrae	N2	Conrad	Almayer
O1	Conrad	Lord Jim	O2	Forster	Room with view
P1	Chesterton	Man who was	P2	Forster	Room with view
Q1	Butler	Erewhon	Q2	Conrad	Almayer
R1	Chesterton	Man who was	R2	Stevenson	Catriona
S1	Morris	News	S2	Hardy	Madding
T1	Conrad	Almayer	T2	Hardy	Well-beloved
U1	Orczy	Elusive P	U2	Chesterton	Man who was
V1	Conrad	Lord Jim	V2	Forster	Room with view
W1	Orczy	Elusive P	W2	Stevenson	Catriona
X1	Hardy	Wessex Tales	X2	Hardy	Well-beloved
Y1	Tressel	Ragged TP	Y2	Orczy	Scarlet P
Z1	Tressel	Ragged TP	Z2	Hardy	Madding

Table A.2. List of 50 text excerpts from the Brunet corpus

#	Author	Short title	#	Author	Short title
1	Marivaux	La vie de Marianne	23	Marivaux	La vie de Marianne
2	Marivaux	Le paysan parvenu	24	Marivaux	Le paysan parvenu
3	Voltaire	Zadig	25	Voltaire	Zadig
4	Voltaire	Candide	26	Voltaire	Candide
5	Rousseau	La nouvelle Héloise	27	Rousseau	La nouvelle Héloïse
6	Rousseau	Emile	28	Rousseau	Emile
7	Chateaubriand	Atala	29	Chateaubriand	Atala
8	Chateaubriand	La vie de Rancé	30	Chateaubriand	La vie de Rancé
9	Balzac	Les Chouans	31	Balzac	Les Chouans
10	Balzac	Le cousin Pons	32	Balzac	Le cousin Pons
11	Sand	Indiana	33	Sand	Indiana
12	Sand	La mare au diable	34	Sand	La mare au diable
13	Flaubert	Madame Bovary	35	Flaubert	Madame Bovary
14	Flaubert	Bouvard et Pécuchet	36	Flaubert	Bouvard et Pécuchet
15	Maupassant	Une vie	37	Maupassant	Une vie
16	Maupassant	Pierre et Jean	38	Maupassant	Pierre et Jean
17	Zola	Thérèse Raquin	39	Zola	Thérèse Raquin
18	Zola	La bête humaine	40	Zola	La bête humaine
19	Verne	De la terre à la lune	41	Verne	De la terre à la lune
20	Verne	Secret de Wilhelm Storitz	42	Verne	Secret de Wilhelm Storitz
21	Proust	Du côté de chez Swann	43	Proust	Du côté de chez Swann
22	Proust	Le temps retrouvé	44	Proust	Le temps retrouvé

Table A.3. List of 100 text excerpts from the St Jean corpus

#	Author	Short title	#	Author	Short title
1	Balzac	Cousine Bette	51	Dumas	Les trois mousquetaires
2	Chateaubriand	Atala	52	Flaubert	Mme Bovary
3	Dumas	Monte Cristo	53	Gautier	Jettatura
4	Flaubert	Bouvard et Pécuchet	54	Goncourt	Germinie Lacerteux
5	Gautier	Avatar	55	Victor	Notre Dame de Paris
6	Goncourt	Mme Gervaisais	56	Maupassant	Notre cœur
7	Victor	Misérables	57	Sand	Indiana
8	Huysmans	A rebours	58	Stendhal	Rouge et Noir
9	Lamartine	Graziella	59	Verne	Tour du monde
10	Maupassant	Bel-Ami	60	Zola	L'Assommoir
11	Musset	Confession	61	Balzac	César Birotteau
12	Nerval	Aurélia	62	Dumas	Les trois mousquetaires
13	Sand	Petite Fadette	63	Flaubert	Mme Bovary
14	Stendhal	Chartreuse de Parme	64	Gautier	Spirite
15	Verne	Terre à la lune	65	Goncourt	Germinie Lacerteux
16	Vigny	Cinq-Mars	66	Victor	Notre Dame de Paris
17	Zola	L'Argent	67	Maupassant	Fort comme la mort
18	Balzac	Cousine Bette	68	Sand	La mare au diable
19	Chateaubriand	Atala	69	Stendhal	Rouge et Noir
20	Dumas	Monte Cristo	70	Vigny	Servitude et grandeur
21	Flaubert	Bouvard et Pécuchet	71	Zola	Bête humaine
22	Gautier	Avatar	72	Balzac	Colonel Chabert
23	Goncourt	Mme Gervaisais	73	Dumas	Les trois mousquetaires
24	Victor	Misérables	74	Flaubert	Un coeur simple
25	Huysmans	A rebours	75	Victor	Notre Dame de Paris
26	Lamartine	Graziella	76	Flaubert	Education Sentimentale
27	Maupassant	Bel-Ami	77	Maupassant	Fort comme la mort
28	Musset	Confession	78	Sand	La mare au diable
29	Nerval	Aurélia	79	Vigny	Servitude et grandeur
30	Sand	Petite Fadette	80	Zola	Bête humaine
31	Stendhal	Chartreuse de Parme	81	Balzac	Colonel Chabert
32	Verne	Terre à la lune	82	Flaubert	Education Sentimentale
33	Vigny	Cinq-Mars	83	Maupassant	Mont-Oriol
34	Zola	L'Argent	84	Zola	Fortune des Rougon
35	Balzac	Cousine Bette	85	Balzac	Le père Goriot
36	Chateaubriand	René	86	Flaubert	Hérodias
37	Dumas	Monte Cristo	87	Maupassant	Mont-Oriol
38	Flaubert	Mme Bovary	88	Zola	Fortune des Rougon
39	Gautier	Jettatura	89	Balzac	Eugénie Grande
40	Goncourt	Germinie Lacerteux	90	Flaubert	Salammbô
41	Victor	Misérables	91	Maupassant	Mont-Oriol
42	Lamartine	Graziella	92	Zola	Germinal
43	Maupassant	Notre cœur	93	Balzac	Eugénie Grandet
44	Musset	Confession	94	Flaubert	Salammbô
45	Sand	Indiana	95	Balzac	Le père Goriot
46	Stendhal	Chartreuse de Parme	96	Maupassant	Une vie
47	Verne	Le tour du monde	97	Balzac	Scènes de la vie
48	Vigny	Cinq-Mars	98	Zola	Germinal
49	Zola	L'Assommoir	99	Stendhal	Rouge et le Noir
50	Balzac	César Birotteau	100	Balzac	Scènes de la vie

A.5 Author Clustering Using Spatium

Mirco Kocher, Jacques Savoy.
Short Paper JCDL 2017, Toronto, Canada, June 19-23, 2017, ACM/IEEE, 265-268.

Author Clustering Using Spatium

Mirco Kocher
Computer Science Dept.
University of Neuchâtel
Neuchâtel, Switzerland
Mirco.Kocher@unine.ch

Jacques Savoy
Computer Science Dept.
University of Neuchâtel
Neuchâtel, Switzerland
Jacques.Savoy@unine.ch

ABSTRACT

This paper presents the author clustering problem and compares it to related authorship attribution questions. The proposed model is based on a distance measure called Spatium derived from the Canberra measure (weighted version of L_1 norm). The selected features consist of the 200 most frequent words and punctuation symbols. An evaluation methodology is presented and the test collections are extracted from the PAN CLEF 2016 evaluation campaign. In addition to those, we also consider two additional corpora reflecting the literature domain more closely. Based on four different languages, the evaluation measures demonstrate a high precision and F1 for all 20 test collections. A more detailed analysis provides reasons explaining some of the failures of the Spatium model.

CCS CONCEPTS

• **Computing methodologies → Unsupervised learning**;

KEYWORDS

Authorship attribution; stylometry; clustering algorithm

ACM Reference format:
Mirco Kocher and Jacques Savoy. 2017. Author Clustering Using Spatium. In *Proceedings of ACM JCDL conference, Toronto, Ontario Canada, June 2017 (JCDL'2017),* 4 pages.
DOI: 10.475/123_4

1 INTRODUCTION

During the last decades, computer-assisted authorship attribution methods have gained an increasing interest, in part due to the presence of numerous pseudonymous posts, chats, threatening e-mails or anonymous documents on the Web. To determine, as accurately as possible, the true author of a document or a text excerpt, various approaches have been proposed [1], [2].

The general authorship attribution issue can be subdivided into four main distinct questions. First, the closed-class attribution problem assumes that the real author is one of the specified candidates. In the open-set situation, the real author could be one of the proposed authors or another unknown one. Third, the verification question provides an answer to whether or not a given author did in fact write a given text [3]. Finally, the authorship attribution can be limited to determining demographic (gender, age class, native language) or psychological traits of the author [4], [5].

In all these cases, the proposed methods assume that a set of documents written by the different possible authors (or categories of authors such as man and woman) are available. The current study focusses on a different perspective where the presence of such labeled data is not provided. This underlying question is called *author clustering* and can be formulated as follows. Having a set of n documents (or text excerpts) written by several distinct authors, determine the number k of distinct authors, and regroup into separate clusters documents written by the same person. As possible applications, a set of proclamations written by different terrorist groups can be clustered, as well as a collection of reviews that can have the same author [6], or a set of novels (or excerpts of literary works). To solve this question, an unsupervised approach must be designed and evaluated.

Recently, the CLEF PAN 2016 evaluation campaign [7]) was launched to stimulate research and evaluation in this direction. The main output is the generation of 18 test collections covering mainly two text genres (newspaper articles & reviews) and written in three languages (English, Spanish, and Greek). In addition, two test corpora extracted from literary novels will be used (one in English [8], and the second in French [9]).

The rest of this paper is organized as follows. The next section presents the state of the art in authorship attribution while the third section describes the test collections and the evaluation methodology used in the experiments. The fourth section explains our new proposed author clustering model while the evaluation appears in the fifth section. Finally, a conclusion draws the main findings of this study.

2 RELATED WORK

The first main component for solving the author clustering problem is to define an effective distance measure between two text representations. Such a function must return a small value when the two documents are written by the same author, and larger ones otherwise. The second issue is to generate an efficient and successful text representation. The third problem consists of developing or applying a clustering procedure able to establish links between texts written by the same author.

The first and second questions are strongly related to classical authorship attribution, but in an unsupervised perspective. A first set of methods suggests to define an invariant stylistic measure [10] that must reflect the particular style of a given author and should vary from one person to another. As possible solutions, different lexical richness measures or word distribution indicators have been proposed [10], as well as the average word length and mean sentence length. None of these measures has proven very satisfactory

JCDL'2017, Toronto, Ontario Canada

DOI: 10.475/123_4

due in part to word distributions ruled by a large number of very low probability elements [11].

As a second framework, a multivariate method can be applied to project into a reduced space each document representation under the assumption that texts written by the same author will appear close together. Some of the main approaches applicable here are principal component analysis (PCA) [12], clustering [13], [8] or discriminant analysis [14]. As stylistic features, these approaches tend to employ the top 50 to 200 most frequent word types (MFW). In a related vein, Layton *et al.* [15] also propose a clustering approach based on their iterative Silhouette method to determine the number of authors in a set of documents.

As a third useful paradigm, different distance-based measures have been suggested. Based on the differences in word distribution between two texts, this paradigm proposes several distance measures. As well-known strategies, one can mention Burrows' Delta [16] using the top m MFW (with m = 40 to 1,000), the Kullback-Leibler divergence [17] using a predefined set of 363 English words, or Labbé's method [8] using the whole vocabulary.

Such distance measures can also be applied with less frequent words. For example, Burrows [18] proposed two distinct but complementary tests. The first one is based on words used regularly by one author but sporadically by the others while the second is grounded on words used infrequently by one author and ignored by the others.

As a clustering algorithm, the complete link seems the more conservative, requiring that all members in a cluster share a high similarity between them. As an alternative, the k-means procedure [19] can be applied. Based on PAN CLEF 2016 results [7], this approach tends to produce lower effectiveness levels than approaches based on distance measures.

3 TEST COLLECTIONS AND EVALUATION METHODOLOGY

3.1 Test Collections

As training collections, the PAN CLEF evaluation campaign has generated the following 6 x 3 collections. Written in English, the first main corpus (Enews) contains newspaper articles extracted from *The Guardian*. The second English set of collections (Erev) contains book reviews (also coming from *The Guardian*). The third is a selection of opinion articles (Dnews) published in the Flemish newspaper *De Standaard*. The Dutch reviews (Drev) were written about different products by students from the University of Antwerp. The next main test corpus is written in Greek and is extracted from the online forum *Protogon* (Gnews) while the Greek reviews (Grev) come from the website Ask4Food (pubs reviews).

Table 1 provides some general information about these corpora, and a more complete description is available in [7]. The first column indicates the name of the collection set, then the language, and the number of collection is provided in the third column. Under the label "Text/Author" the number of texts and authors per sub-corpus are given. In the last column, the mean size of each text is depicted. For example, the second row shows that the Enews corpus contains 3 collections, the first third is composed on 50 texts written by respectively 35, 25, and 43 distinct authors. The mean size of these newspaper articles is 741 words/article for the first collection, 745 for the second, and 734 for the last.

Table 1: Statistics about the Test Collections

Name	Lang.	Col.	Text/Author	Size (words)
Enews	EN	3	50 / {35, 25, 43}	741; 745; 734
Erev	EN	3	80 / {55, 70, 40}	969; 1080; 1020
Dnews	DT	3	57 / {51, 28, 40}	1086; 1335; 1027
Drevs	DT	3	100 / {54, 67, 91}	129; 136; 126
Gnews	GR	3	55 / {28, 38, 48}	756; 750; 735
Grev	GR	3	55 / {50, 28, 40}	534; 646; 576
Oxquarry1	EN	1	52 / 9	10,101
Brunet	FR	1	44 / 11	8,240

From these test collections extracted from the PAN CLEF 2016 campaign, we added the Oxquarry1 corpus regrouping 52 excerpts from novels written by 9 distinct authors (e.g., Stevenson, Hardy, Conrad). As a constraint when generating this corpus, each author must appear with at least two texts. Similarly, the French corpus, called Brunet, contains 44 excerpts of novels written by 11 different well-known writers. In this corpus, each author is represented with four text excerpts extracted from two novels. A more complete description of these last two corpora can be found in [8] and [9].

3.2 Evaluation Methodology

As evaluation methodology, the complete author clustering was chosen. Under this evaluation method, each document must be assigned to exactly one cluster and each cluster must contain all texts written by the same writer.

As performance measure, the purity value can be selected. We prefer to opt for the evaluation measures chosen during the PAN CLEF 2016 campaigns. In this case, the correctness function ($c()$) between two documents d_i and d_j is computed as follows:

$$c(d_i, d_j) = \begin{cases} 1 & if\ A(d_i) = A(d_j) \wedge C(d_i) = C(d_j) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $A(d_i)$ indicates the real author of d_i, and $C(d_i)$ is the cluster in which d_i occurs. Based on this notation, the BCubed precision (denoted $pr(d_i)$) and recall ($re(d_i)$) for d_i is defined as:

$$pr(d_i) = \frac{\sum_{d_j \in C(d_i)} c(d_i, d_j)}{|C(d_i)|} \qquad re(d_i) = \frac{\sum_{d_j \in C(d_i)} c(d_i, d_j)}{|A_i|} \quad (2)$$

Having n texts in a given test collection, the BCubed precision and recall for the whole corpus is defined as:

$$precision = 1/n \cdot \sum_{i=1}^{n} pr(d_i) \qquad recall = 1/n \cdot \sum_{i=1}^{n} re(d_i) \quad (3)$$

from which the well-known F1 performance measure can be computed (as F1 = (2 x precision x recall) / (precision + recall)), with the higher the value, the better the performance.

A closer look at this performance measure indicates that a 100% precision can be achieved by having one cluster for each document. Thus, both the precision and the recall are required for the F1 measure to be useful to rank the different approaches.

4 PROPOSED METHOD

To represent mainly the stylistic aspects of a text, various studies [2], [16], [20] have shown that the m (with m = 50 to 1,000) most frequent words (MFW) tend to provide high accuracy rates. We follow this feature selection procedure and fix m = 200. In this study, a word is defined as a sequence of letters or punctuation symbols. The distinction between uppercase and lowercase is ignored. Thus, in the sequence "Paul's book is red", we count six words, namely {paul, ', s, book, is, red}.

To represent the stylistic aspects of a text our Spatium model [21] takes account of the m = 200 MFW derived from Text A. Therefore, this measure is not symmetric because the features for Δ(A,B) can be different from those used in Δ(B,A). This m value must be viewed as an upper limit because in many cases the input text is smaller than this limit, as depicted in Table 1, for example, for the Dutch reviews (Drev) collection. Moreover, even when the mean text size is larger than m, the number of distinct word types can be smaller than 200. To measure the distance between two Texts A and B, the Spatium model is based on the Canberra distance (see Eq. 4).

$$\Delta(A,B) = \sum_{i=0}^{m} \frac{|P_A(t_i) - P_B(t_i)|}{P_A(t_i) + P_B(t_i)} \quad (4)$$

where $P_A(t_i)$ and $P_B(t_i)$ represent the estimated occurrence probability of the term t_i in the Text A or B. To estimate these probabilities, the term occurrence frequency (denoted tf_i) is divided by the length in tokens of the used text (n), $P(t_i) = tf_i/n$.

The Spatium model must however be adapted in the current context. Observing a small value for Δ(A,B) is an indication that both texts are written by the same author. On the other hand, a large value suggests the opposite. The problem is then to define what is a "small distance value". Having a collection, the distance *from* A to all other texts is computed and, from this distribution, the mean (denoted m(A,.)) and standard deviation (std(A,.)) is calculated. Moreover, the distribution of distance *to* Text A can be computed to provide the mean m(.,A) and the standard deviation std(.,A) of the intertextual distance to Text A.

As a first solution to define "small" and "large" distance, we can assume that a small distance value from Text A is defined by Eq. 5. In this equation, δ is a parameter to be fixed. Assuming a Gaussian distribution, setting δ =1.64 means that 5% of the observations are smaller than the mean - 1.64·std.

$$Hint1 : \Delta(A,j) \leq \phi(A,.) = m(A,.) - \delta \cdot std(A,.) \quad (5)$$

Similarly, a small distance *to* Text A can be defined as:

$$Hint\ 2 : \Delta(j,A) \leq \phi(.,A) = m(.,A) - \delta \cdot std(.,A) \quad (6)$$

Our idea is to ground our attribution decision on a more solid foundation. In this case, to verify whether the distance value Δ(A,B) is small or not (implying that both texts are written by the same author), we need to consider more than a single limit. In our Spatium model, the value Δ(A,B) is defined as *small* if it is smaller than or equal to two of the four limit values: $\phi(A,.)$, $\phi(.,A)$, $\phi(B,.)$, $\phi(.,B)$. The choice of the δ parameter, and the number of limits to be respected (two in our case) indicate the willingness of having more or less strict assignments. A smaller value for δ generates more links between texts and thus increases the risk of observing incorrect assignments. When assuming that a corpus is composed by many authors with clusters having a few elements, the parameter δ can be fixed at a higher level (e.g., 1.96, corresponding to 2.5% of the values of a Gaussian distribution).

Table 2: Evaluation over the 20 Test Collections

Name	Precision.	Recall	F1
Enews	0.91; 0.95; 0.98	0.72; 0.5; 0.87	0.803; 0.667; **0.924**
Erev	0.86; 0.87; 0.85	0.72; 0.88; 0.55	0.780; **0.875**; 0.666
Dnews	0.98; 0.98; 0.96	0.91; 0.56; 0.73	**0.943**; 0.709; 0.830
Drev	0.90; 0.93; 0.92	0.56; 0.68; 0.91	0.701; 0.782; **0.915**
Gnews	0.86; 0.96; 0.96	0.60; 0.75; 0.92	0.708; 0.841; **0.939**
Grev	0.83; 0.95; 0.95	0.96; 0.62; 0.82	**0.893**; 0.748; 0.879
Mean 18	0.935	0.735	0.815
Oxquar.	1.00	0.85	0.917
Brunet	0.85	0.67	0.749
Mean 2	0.924	0.758	0.833

5 EVALUATION

After applying the proposed approach to a corpus, a set of links between texts are determined. From them, one can generate the clusters grouping all texts having at least a link between them. For example, having a link between A and B, and another between B and C, the cluster {A, B, C} is formed.

Table 2 exposes the evaluation done according to Eq. 3 for the 18+2 corpora. With the line 'Mean 18" ("Mean 2"), the average is given for the PAN collections (18), and for the two literature corpora. As indicated previously, the suggested method is rather conservative and the precision values are relatively high. When analyzing the F1 measure, the best values are shown in bold and correspond to sub-corpora having the largest number of authors (see Table 1). To be more precise, the parameter δ was fixed to 1.96 for the 18 PAN corpora, and 1.64 for the Oxquarry1 & Brunet collections (we expect larger clusters for these two corpora).

When inspecting the results for the Oxquarry1 collection (excerpts of English novels written in the 19th century), our model was able to regroup correctly the 12 texts written by Hardy, 7 by Stevenson, 6 by Morris, 6 by Orzcy, 4 by Butler, and 3 by Chesterton. But the proposed clustering was not perfect. The 8 excerpts from Conrad are split in two clusters with four documents apiece. Each of the three texts written by Tressel, and Forster can be found in single clusters. For those texts, the computed distance between the texts, even written by the same author, are too high. This could be the case when an author has the ability to write with distinct styles.

With the French literature corpus (Brunet), the overall performance is lower compared to the English one (F1: 0.749 vs. 0.917). In the Brunet's corpus, we have exactly four texts per author (and 11 distinct writers). Our approach is able to form a cluster with the four texts written by Voltaire or Proust. The four excerpts of Maupassant, and Flaubert were also detected, but Spatium adds a link between Flaubert's and Maupassant's clusters. In a similar way, the four texts of Marivaux form a cluster but a link is added with a cluster of two works written by Sand. These two-incorrect links are the only false positive ones. Six small clusters have been

Table 3: Mean Evaluation over Different Baselines

Name	Precision	Recall	F1
Our model over 18 PAN corpora	0.935	0.735	**0.815**
1 text, 1 cluster	1.0	0.697	0.812
All texts, 1 cluster	0.034	1.0	0.065
Our model over 2 litera. corpora	0.924	0.758	**0.833**
1 text, 1 cluster	1.0	0.212	0.348
All texts, 1 cluster	0.114	1.0	0.204

generated composed by two texts written by the same author, and the last ten excerpts are left in their own cluster.

The analysis of the results of these corpora shows that Spatium tends to produce few false positive assignments. This aspect is one of our main objectives. We designed the attribution system in order to *tell the truth* (high precision) but as the performance is not perfect, the implementation does not *tell the whole truth* (recall).

As baselines (not shown in Table 3), we take the average F1 scores from the best three or best five PAN participants [7]. In the English corpora we achieve a similar performance (-1%) as the baseline, but for the Dutch and Greek texts our results were better (+10% and +7% resp.). For our approach the text genre had a non-significant influence on the outcome (F1: 0.81 for news vs. 0.80 for reviews).

In Table 3, one can see on the top the evaluation over the 18 test collections extracted from the PAN CLEF 2016, and in the bottom part, the performance achieved with the two literature corpora. As additional baselines indicated in Table 3, we can assume that each text was written by a distinct author ("1 text, 1 cluster") producing a precision of 1.0. This baseline presents a high F1 values as indicated in Table 3. Our method however proposes a slightly higher F1 performance. On the other hand, all texts can be grouped into a single cluster ("all texts, 1 cluster") leading to a recall of 1.0, but a small F1 value. Compared to this second baseline, our model shows clearly a better overall performance.

6 CONCLUSION

The author clustering problem can be encountered more frequently on web-based communication (e.g., e-mail, chat, blogs). But more classical applications do exist (Are all of Shakespeare's plays written by a single author? Who is behind the contemporary novels of Elena Ferrante, a single author or a few). The detection of plagiarism constitutes another related application for the author clustering problem. Formally, the author clustering problem is defined as follows: Having a corpus of n texts, find the number k of writers and regroup them into k clusters, one per author.

To resolve this problem using an unsupervised approach, our Spatium model represents each text according to its stylistic aspects. To achieve this the m most frequent word types (MFW), and punctuation symbols (with m=200 in this study) form the text surrogate. Considering the MFW, the focus is placed on functional words (articles, prepositions, pronouns, conjunctions, and modal verbs) reflecting closely the style. Of course, other stylistic elements could be added such as POS distributions, overall stylometric measurements (e.g., mean sentence length).

The second decision consists of selecting an intertextual distance. In our approach, a modified Canberra function has been chosen. As the term selection depends on the argument order, the proposed measure is not symmetric. Based on this function, we suggest to define small distance values by considering the 2.5% or 5% smallest distance values computed according to the corpus.

Experiments done on 20 test collections indicate that the Spatium model demonstrates high performance levels. A failure analysis indicates that the proposed solution tends to produce a reduced number of false positive links between two papers not written by the same author. On the other hand, this model does not capture all relationships between text excerpts written by the same person. Working with a literary corpus, the proposed method can reveal writers having a single and discriminative style and those who can adopt different styles thus producing texts that are more difficult to regroup under the same cluster. As further empirical evidence about the quality of the proposed solution, we can mention that during the PAN CLEF 2016 evaluation campaign, Spatium achieved the second-best clustering performance.

ACKNOWLEDGMENTS

This research was supported by the NSF (Grant #200021_149665/1).

REFERENCES

[1] H. Love. *Attributing Authorship.* Cambridge University Press, 2002.
[2] E. Stamatatos. A survey of modern authorship attribution methods. *JASIST*, 60 (3):433–214, 2009.
[3] J. A. Stover, Y. Winter, M. Koppel, and M. Kestemont. Computational authorship verification method attributes a new work to a major 2nd century african verification. *JASIST*, 67(1):239–242, 2016.
[4] S. Argamon, M. Koppel, J.W. Pennebaker, and J. Schler. Automatically profiling the author of an anonymous text. *Communications ACM*, 52(2):119–123, 2009.
[5] J. W. Pennebaker. *The Secret Life of Pronouns. What our Words Say about us.* Bloomsbury Press, 2011.
[6] M. Almishari and G. Tsudik. Exploring linkability of user reviews. In *Proceedings Computer Security ESORICS*, pages 307–324, 2012.
[7] E. Stamatatos, M. Tschuggnall, B. Verhoeven, W. Daelemans, G. Specht, B. Stein, and M. Potthast. Clustering by authorship within and across documents. In CEUR, editor, *Notebook Papers of CLEF 2016 Labs and Workshop*, Aachen, 2016.
[8] D. Labbé. Experiments on authorship attribution by intertextual distance in english. *Journal of Quantitative Linguistics*, 14(1):33–80, 2007.
[9] C. Labbé and D. Labbé. A tool for literary studies. *Literary & Linguistic Computing*, 21(3):311–326, 2006.
[10] D. I. Holmes. The evolution of stylometry in humanities scholarship. *Literary and Linguistic Computing*, 13(3):111–117, 1998.
[11] H. R. Baayen. *Analysis Linguistic Data.* Cambridge University Press, 2008.
[12] H. Craig and A. F. Kinney. *Shakespeare, Computers, and the Mystery of Authorship.* Cambridge University Press, 2009.
[13] D. I. Holmes and R. S. Forsyth. The *Federalist* revisited: New directions in authorship attribution. *Literary and Linguistic Computing*, 10(2):111–127, 1995.
[14] M. L. Jockers and D. M. Witten. A comparative study of machine learning methods for authorship attribution. *Literary - Linguistic Computing*, 25(2):215–223, 2010.
[15] R. Layton, P. Watters, and R. Dazeley. Evaluating authorship distance methods using the positive silhouette coefficient. *Natural Language*, 19:517–535, 2013.
[16] J. F. Burrows. Delta: A measure of stylistic difference and a guide to likely authorship. *Literary and Linguistic Computing*, 17(3):267–287, 2002.
[17] Y. Zhao and J. Zobel. Searching with style: Authorship attribution in classic literature. In *Proceedings of the Thirtieth Australasian Computer Science Conference (ACSC2007)*, pages 59–68, Ballarat, 2007.
[18] J. Burrows. All the way through: Testing for authorship in different frequency strata. *Literary and Linguistic Computing*, 22(1):27–47, 2007.
[19] I. H. Witten, E. Frank, and M. A. Hall. *Data Mining. Practical Machine Learning Tools and Techniques.* Morgan Kaufmann, 2011.
[20] J. Savoy. Comparative evaluation of term selection functions for authorship attribution. *Digital Scholarship in the Humanities*, 30(2):246–261, 2015.
[21] M. Kocher and J. Savoy. A simple and efficient algorithm for authorship verification. *JASIST*, 68(1):259–269, 2017.

A.6 Author Clustering with an Adaptive Threshold

Mirco Kocher, Jacques Savoy.
In *Jones, G. J. F., Lawless, S., Gonzalo, J., Kelly, L., Goeuriot, L., Thomas M., Cappellato, L., & Ferro, N. (Eds)*, Experimental IR Meets Multilinguality, Multimodality, and Interaction, Proceedings of the 8th International Conference of the CLEF Association, CLEF 2017, Dublin, Ireland, September 11-14, 2017, 186-198.

Author Clustering with an Adaptive Threshold

Mirco Kocher(✉) and Jacques Savoy

Computer Science Department, University of Neuchâtel, Neuchâtel, Switzerland
{Mirco.Kocher,Jacques.Savoy}@unine.ch

Abstract. This paper describes and evaluates an unsupervised author clustering model called SPATIUM. The proposed strategy can be adapted without any difficulty to different natural languages (such as Dutch, English, and Greek) and it can be applied to different text genres (newspaper articles, reviews, excerpts of novels, etc.). As features, we suggest using the m most frequent terms of each text (isolated words and punctuation symbols with m set to at most 200). Applying a distance measure, we define whether there is enough evidence that two texts were written by the same author. The evaluations are based on six test collections (PAN AUTHOR CLUSTERING task at CLEF 2016). A more detailed analysis shows the strengths of our approach but also indicates the problems and provides reasons for some of the potential failures of the SPATIUM model.

Keywords: Author clustering · Threshold · Author identification · PAN

1 Introduction

With the increased communication facilities and the ubiquity of social media, we encounter an enlarged number of authorship problems. With the believed anonymity offered by the Web, the number of anonymous and pseudonymous texts or threats is increasing. To be able to automatically determine the real author of a text presents a clear interest for criminal investigations as well as for historical or literature studies (e.g., who really is the novelist Elena Ferrante?).

In this perspective, the classical question is to determine the real author of a given text, usually based on a set of documents with known authorship. But the *author clustering* task is more demanding. This problem can be formulated as follows: given a corpus of n texts, regroup all documents written by the same author such that each of the k clusters corresponds to a distinct author. For example, based on a set of n passages extracted from a collaborative work, we should first determine the number of authors k and then regroup the texts into k clusters according to their real author.

This paper is organized as follows. The next section presents the related work while Sect. 3 briefly describes the test collections and the evaluation methodology used in our experiments. Section 4 describes our proposed algorithm based on the SPATIUM model. Section 5 evaluates the proposed scheme and compares it to the best performing schemes using six different test collections extracted from CLEF PAN 2016. Then, Sect. 6 provides an analysis to assess the variability of the performance measures. Finally, Sect. 7 exposes our adaptive threshold system that can extract some correct assignments

G.J.F. Jones et al. (Eds.): CLEF 2017, LNCS 10456, pp. 186–198, 2017.
DOI: 10.1007/978-3-319-65813-1_19

even when the information available is rather limited. A conclusion draws the main findings of this study.

2 Related Work

The author clustering problem was introduced as a new task in the PAN CLEF 2016 track. In this view, Stamatatos *et al*. [16] provide a good overview of the proposed methods. Overall, the first main component for solving this issue is to define an effective distance measure between two text representations. Such a function returns a small value when the two documents are written by the same author, and a larger one otherwise. Of course, instead of defining a distance measure, one can propose a similarity measure and accept that two texts were written by the same person when the similarity value is high enough. The second problem consists of developing or applying a clustering procedure capable of establishing links between texts written by the same author. In this case, after assuming that Text A and B have the same author, as well as Text A and C, one can infer that Text B and C have been written by the same source as well (single link strategy).

An answer to the first question is related to classical authorship attribution, but in an unsupervised perspective. A first set of methods suggests defining an invariant stylistic measure [5] that must reflect the particular style of a given author and should vary from one person to another. Furthermore, we can assume that an author's writing style is stable over period of time (e.g., one decade) before showing measurable differences [4]. A multivariate method can be applied to project each document representation into a reduced dimensional space under the assumption that texts written by the same author will appear close together. Some of the main approaches applicable here are principal component analysis (PCA) [3], clustering [10], or discriminant analysis [6]. As stylistic features, these approaches tend to employ the top 50 to 200 most frequent word types (MFW), as well as some part-of-speech (POS) information. In a related vein, Layton *et al*. [11] also propose a clustering approach based on their iterative Silhouette method to determine the number of authors in a set of documents.

Based on the differences in word distribution between two texts, several distance-based measures have been proposed [9]. As well-known functions defined more specifically for solving the authorship attribution question, one can mention Burrows' Delta [2] using the top m MFW (with $m = 40$ to 1,000), the Kullback-Leibler divergence [18] using a predefined set of 363 English words, or Labbé's method [10] using the whole vocabulary.

Finally, as a clustering algorithm, the complete link seems the more conservative strategy, requiring that all members in a cluster share a high similarity between them. As an alternative, the k-means procedure [17] can be applied. Based on PAN CLEF 2016 results [16], this approach tends to produce lower effectiveness levels than approaches based on distance measures.

3 Test Collections and Evaluation Methodology

To promote research and to evaluate author clustering algorithms, the CLEF PAN 2016 generated a benchmark composed of six test collections covering three languages (English, Dutch, and Greek) and two text genres (newspaper articles and customer reviews). For each of the six language/text genre combinations, one can find three "collections" denoted "problems" in the PAN parlance. Thus, for each language, one can find three problems composed of newspaper articles and three others containing reviews.

During the PAN CLEF 2016, there were 3×6 problems available for training with their main statistics as reported in Table 1. In this table, the number of texts belonging to each language/genre combination is indicated under the label "Texts". For example, with the EA (English Articles), one can find three problems, each containing 50 articles. The number of distinct authors per problem is indicated in the column "Authors", and the number of authors with a single document under the label "Single". Thus, the first problem in the EA test collection has 35 authors, from which 27 have written only one article. In the last column, the mean number of words per text is depicted.

Table 1. PAN CLEF 2016 *training* corpora statistics

Corpus	Texts	Training problems		
		Authors	Single	Words
English Articles (EA)	50	35; 25; 43	27; 17; 37	741; 745; 734
English Reviews (ER)	80	55; 70; 40	39; 62; 17	969; 1080; 1020
Dutch Articles (DA)	57	51; 28; 40	46; 20; 32	1086; 1334; 1026
Dutch Reviews (DR)	100	54; 67; 91	31; 44; 83	128; 135; 126
Greek Articles (GA)	55	28; 38; 48	10; 26; 42	756; 750; 735
Greek Reviews (GR)	55	50; 28; 40	46; 13; 29	534; 646; 756

During the PAN CLEF 2016 evaluation campaign, 18 additional problems were built (test phase) with the same distribution over the languages and text genres as the training collections (shown in Table 1). As the correct statistics for those corpora are still undisclosed, our study will focus mainly on the training corpora.

When inspecting the training problems, we note that the number of words available in DR is rather small (in mean, 130 words for each document). Moreover, there are many authors who only wrote a single text, so the number of authors per problem is rather large (as well as the number of expected clusters). This means that we should only regroup two documents if there is enough evidence for a single authorship.

As proposed in the PAN CLEF 2016 track, an author clustering algorithm is evaluated with two distinct metrics. First, the purity of the generated clusters is evaluated. In this perspective, a perfect system must create only k clusters, each containing all the documents written by the same person. The evaluation measures are the precision, the recall, and the harmonic mean between the two values (denoted BCubed F_1) [1]. Moreover, each document must belong to exactly one cluster. To achieve a perfect precision, the solution is to generate one cluster per document. Therefore, the purity of each cluster

is maximal and the resulting precision is 1.0. On the other hand, to achieve a recall of 1.0, all documents can be regrouped into a single cluster. Thus, the two measurements are in opposition. The F_1 value will serve as an effectiveness measure of the resulting clusters, with a higher value meaning a better distribution.

As a second measure, one can ask the clustering algorithm to return a list of links between text pairs, ordered by an estimated probability of having the same author for the two cited documents. To evaluate such an ordered list, one can apply the mean average precision (MAP) [16]. As complementary measures, the precision after 10 ranks (P@10) or the RPrec can be computed. MAP is a classical evaluation measure in the IR domain [12]. It is known that this measure is sensitive to the first rank(s), and providing an incorrect answer in the top ranks intensively hurts the MAP value. On the other hand, MAP does not punish verbosity, i.e., every true link counts even when appearing near the end of the ranked list. Therefore, by providing all possible authorship links, one can attempt to maximize MAP, without penalizing the P@10.

4 Simple Clustering Algorithm

To solve the clustering problem, we propose an adapted approach based on a simple feature extraction and distance metric called Spatium [7]. The selected stylistic features correspond to the top m most frequent terms (isolated words without stemming, but with the punctuation symbols) from the query text. For determining the value of m, previous studies have shown that a value between 200 and 300 tends to provide the best performance in the authorship attribution domain [2, 13]. Moreover, we will exclude the words appearing only once (*hapax legomenon*) in the text for the feature selection. This filtering decision was taken to prevent overfitting to single occurrences.

As shown in Table 1, some documents were rather short. Therefore, the real number of terms m was set to at most 200 terms but, in most cases, was 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.

To measure the distance between a Text A and another Text B, the Spatium model uses a weighted variant of the L^1-norm which was already found to be useful in a related task [9]. The Canberra distance suggests that the absolute differences of the individual terms are normalized based on the sum of them as indicated in Eq. 1.

$$\Delta_{AB} = \Delta(A,B) = \sum_{i=1}^{m} \frac{\left|P_A[t_i] - P_B[t_i]\right|}{P_A[t_i] + P_B[t_i]} \tag{1}$$

where m indicates the number of terms (words or punctuation symbols) occurring in A more than once, $P_A[t_i]$ and $P_B[t_i]$ represent the estimated occurrence probability of the term t_i in Text A and Text B respectively. To estimate these probabilities, we divide the term occurrence frequency (tf_i) by the length in tokens of the text (n), $Prob[t_i] = tf_i/n$, without smoothing and an estimation of 0.0 may occur in Text B.

For example, assume that Text A corresponds to "The fox, the moose, and the deer jump over a wolf." Based on the term frequency, the resulting vector is [the (3), (2)] after ignoring the letter case. The other words occurring once are ignored. The final representation is: [the (3/5), (2/5)]. Assuming Text B contains the following sentence: "The quick fox and the brown deer jump over the lazy dog and a cat." When computing the distance Δ_{AB}, the following terms are used {the ,} because they are extracted from the representation of Text A. The representation of Text B is therefore [the (3/3), (0/3)]. Applying Eq. 1 with these two terms gives us $\Delta_{AB} = 1.25$. On the other hand, when estimating the distance Δ_{BA}, only terms belonging to B's representation are considered, namely {the and}, giving us the representation [the (3/5) and (2/5)] for Text B and [the (3/4) and (1/4)] for Text A, resulting in a distance $\Delta_{BA} = 0.34$. This distance measure is not symmetric due to the choice of the terms.

Observing a small value for Δ_{AB} provides evidence that both documents are written by the same author. On the other hand, a large value suggests the opposite assuming the text length is long enough to support this finding. The real problem consists in defining precisely what a "small distance value" is. To verify whether the resulting Δ_{AB} value is small, a comparison basis must be determined.

To achieve this with a specific collection, the distance *from* A to all other texts is computed (or $\Delta(A, j)$). From this distribution, the mean (denoted $m(A, .)$) and standard deviation ($std(A, .)$) are estimated. Moreover, the distribution of distance values *to* Text B (or $\Delta(j, B)$) can be computed to provide the mean $m(., B)$ and the standard deviation $std(., B)$ of the intertextual distances *to* Text B.

As a first definition of a "small" distance, we can assume that a small distance value *from* Text A must respect Eq. 2. In this formulation, δ is a parameter to be fixed. Assuming a Gaussian distribution, setting $\delta = 1.645$ means that 5% of the observations are smaller than the $mean - 1.645 * std$.

$$Hint\ 1\text{: } \Delta(A, j) \leq \phi(A, .) = m(A, .) - \delta * std(A, .) \tag{2}$$

Similarly, a small distance *to* Text B can be defined as:

$$Hint\ 2\text{: } \Delta(j, B) \leq \phi(., B) = m(., B) - \delta * std(., B) \tag{3}$$

With these two decision rules, one can verify if a distance *from* Text A (Eq. 2) or *to* Text B (Eq. 3) is small or not. We propose to be more cautious, mainly because proposing an incorrect assignment must be viewed as more problematic than missing a link between two documents written by the same author.

To follow this idea, having a distance value Δ_{AB}, we can verify the magnitude of its value according to Eq. 2 (*from* A) and Eq. 3 (*to* B). In the same way, one can verify whether the resulting Δ_{BA} value is small or rather large. Therefore, we propose to create two additional decision rules with Eq. 4 (based on the distribution of distance values *from* Text B) and Eq. 5 (for distance *to* Text A) as follows:

$$Hint\ 3\text{: } \Delta(B, j) \leq \phi(B, .) = m(B, .) - \delta * std(B, .) \tag{4}$$

$$Hint\ 4\text{: } \Delta(j, A) \leq \phi(., A) = m(., A) - \delta * std(., A) \tag{5}$$

To ground our attribution decision on a solid foundation, we compute both the distance Δ_{AB} and Δ_{BA} and check all four hints. An authorship between Text A and B is expected if at least two of the four hints are satisfied.

The choice of the parameter value δ, and the number of limits to be respected (two in our case) indicate the willingness of having more or less strict assignments. A smaller value for δ generates more potential links between texts and thus increases the risk of observing incorrect assignments. If a corpus is composed of many authors with each cluster contains only a few items, the parameter δ can be fixed at a higher level (e.g., $\delta = 1.96$, corresponding to 2.5% of the values of a Gaussian distribution).

5 Evaluation

Based on the gold standard provided by the CLEF PAN 2016 dataset, the SPATIUM model with the threshold value $\delta = 2$ can be evaluated as shown in Table 2. This table reports the performance measures applied during the PAN CLEF campaign, namely the BCubed F_1 and the MAP presented in Sect. 3. These measures are not provided for each problem but only the average over the three problems included in each test collection. Under the term "Score" we report the mean between the F_1 and MAP value.

Table 2. Evaluation for the six *training* collections

Corpus	Score	F_1	MAP
English Article (EA)	0.4601	0.7972	0.1229
English Review (ER)	*0.4242*	*0.7656*	*0.0828*
Dutch Article (DA)	0.5184	0.8387	0.1981
Dutch Review (DR)	*0.4192*	*0.7895*	*0.0488*
Greek Article (GA)	0.5649	0.8294	0.3004
Greek Review (GR)	**0.6878**	**0.8588**	**0.5168**
Average	0.5124	0.8124	0.2116

The best performance values are depicted in bold. As one can see, the Spatium returns the best results for the GR collection with a final score of 0.6878 followed by the GA and DA test collection. The worst result is achieved with the ER and DR collections (values depicted in italics). Moreover, the BCubed F_1 is very similar over all collections but the variability of the MAP is remarkable. The achieved MAP with the GR corpus is almost ten times higher than in the DR or ER corpus.

The evaluation performed on the test set is depicted in Table 3. The differences between the training and test corpus are relatively small. Similar clustering performances can be achieved using either the training or test set, indicating a strong correlation between the two samples. Since our model is unsupervised, there is no influence of one collection on the other, and no resources have been used to fix any parameter values or to build a learning structure.

Compared to the other participants of the PAN 2016 author clustering task, we achieve the second best overall score with one of the fastest systems. Some texts were

wrongly grouped up, which decreases the document precision part of the BCubed F-Score a bit. Overall, we cluster many documents correctly together (which increases document recall part) and assign them a high score for their authorship link (which increases MAP).

Table 3. Evaluation for the six *test* collections

Corpus	Score	F_1	MAP
English Article (EA)	0.4348	0.7518	0.1178
English Review (ER)	0.4320	0.7869	*0.0772*
Dutch Article (DA)	0.4742	0.8183	0.1301
Dutch Review (DR)	*0.4106*	0.7702	*0.0510*
Greek Article (GA)	0.4891	0.8005	0.1778
Greek Review (GR)	**0.5660**	**0.8326**	**0.2995**
Average	0.4678	0.7934	0.1422

6 Sensibility Assessment

To provide a fair evaluation methodology, we cannot simply compare the performance values (MAP, F_1, or Score) directly between two approaches. A leaving-one-out or cross-fold evaluation is not possible in this task. We need to estimate the underlying variability of each performance using, for instance, the bootstrap approach. In this approach, for each problem, the system must generate S new random bootstrap samples. More precisely, for each text, we will create $S = 200$ new copies having the same length. For each copy the probability of choosing one given term (word or punctuation symbol) depends on its relative frequency in the original text. This drawing is done with replacement; thus, the underlying probabilities are fixed.

Each resulting text must be viewed as a bag-of-words. As the syntax is not respected, each bootstrap text is not really readable but reflects the stylistic aspects as analyzed by the SPATIUM approach.

For each of the 200 generated collections of bootstrap samples, we have applied our approach and obtained the MAP and the BCubed F_1 values reported in Tables 4 and 5. In Table 4, the column F_1 (or MAP in Table 5) indicates the performance achieved with the original data (as presented in Table 2). Then the column labeled "$\overline{x}$" reports the mean of the F_1 (or MAP respectively) achieved with the 200 new collections, together with the limit of ± 2 standard deviations σ (last two columns) corresponding to a confidence interval of 95.4%.

As depicted in Table 4, the reported performance for the EA collection is 0.7972. With the bootstrap methodology, the 95.4% confidence interval is [0.7551; 0.8085] for this value. As one can see in Table 4 (F_1 values), the mean of the bootstrap sample is usually lower (around 2%) than the original performance values but the original performance is always within the confidence interval of the bootstrap sample. In Table 5, the difference between the original MAP performances and the mean of the bootstrap sample is larger. For the DR corpus (Table 5), the difference is rather small (around 4%) while

in the EA collection a drop of over 50% can be observed. It is known that the MAP measure is more sensitive to variations because a misclassification in the highest ranks is strongly penalized leading to a higher standard deviation.

Table 4. Results for the BCubed F_1 after applying the bootstrap estimation

Corpus	F_1	$\bar{x}$	$\bar{x} - 2\sigma$	$\bar{x} + 2\sigma$
English Article (EA)	0.7972	0.7818	0.7551	0.8085
English Review (ER)	0.7656	0.7448	0.7091	0.7805
Dutch Article (DA)	0.8387	0.8210	0.7970	0.8450
Dutch Review (DR)	0.7895	0.7699	0.7394	0.8005
Greek Article (GA)	0.8294	0.8088	0.7777	0.8399
Greek Review (GR)	0.8588	0.8452	0.8173	0.8732
Average	0.8124	0.7952	0.7659	0.8246

Table 5. Results for the MAP after applying the bootstrap estimation

Corpus	MAP	$\bar{x}$	$\bar{x} - 2\sigma$	$\bar{x} + 2\sigma$
English Article (EA)	0.1229	0.0578	0.0179	0.0978
English Review (ER)	0.0828	0.0490	0.0138	0.0842
Dutch Article (DA)	0.1981	0.1159	0.0579	0.1740
Dutch Review (DR)	0.0488	0.0466	0.0317	0.0616
Greek Article (GA)	0.3004	0.2015	0.1013	0.3017
Greek Review (GR)	0.5168	0.4279	0.3271	0.5288
Average	0.2116	0.1498	0.0916	0.2080

7 Adaptive Thresholding

To improve our knowledge, it is important to understand why and when an automatic text categorization scheme fails to provide the correct answer. Such an analysis will reveal more precisely the advantages and drawbacks of a suggested scheme. In the current context, the important question is related to the definition of a pertinent threshold in defining our limits (see Eqs. 2 to 5).

In a previous study [8], we had to classify, under the same condition, 52 excerpts of English novels containing in mean 10,000 tokens [10]. In this corpus, nine authors had written multiple texts (specifically Hardy wrote 12 texts, Conrad wrote 8, Stevenson (7), Morris (6), Orczy (6), Butler (4), Chesterton (3), Forster (3), and Tressel (3)).

When analyzing the Canberra distance between all possible pairs of texts, the global distribution is a mixture of two distributions. The first one corresponds to the distance values obtained when the two texts are written by the same author (shown in blue or white on the left part of Fig. 1). The second one results from pairs composed of two texts written by two persons (depicted in red or gray on the right part in Fig. 1). In Fig. 1, one can see these two distributions in which the three means are indicated with the vertical lines. On the left part, one can observe the mean of the correct links (denoted by

"Mean(Blue)"), the mean of the mixed distribution ("Mean(MixDist)") and on the right, the mean of the incorrect links ("Mean(Red)"). In this figure, the limit proposed in our four hints in Eqs. 2 to 5 and corresponding to the mean – 1.64 * std ("Mean – 1.64 * SD") appears with a vertical line on the left. As we can see in this figure, all distances below this limit correspond to correct pairings.

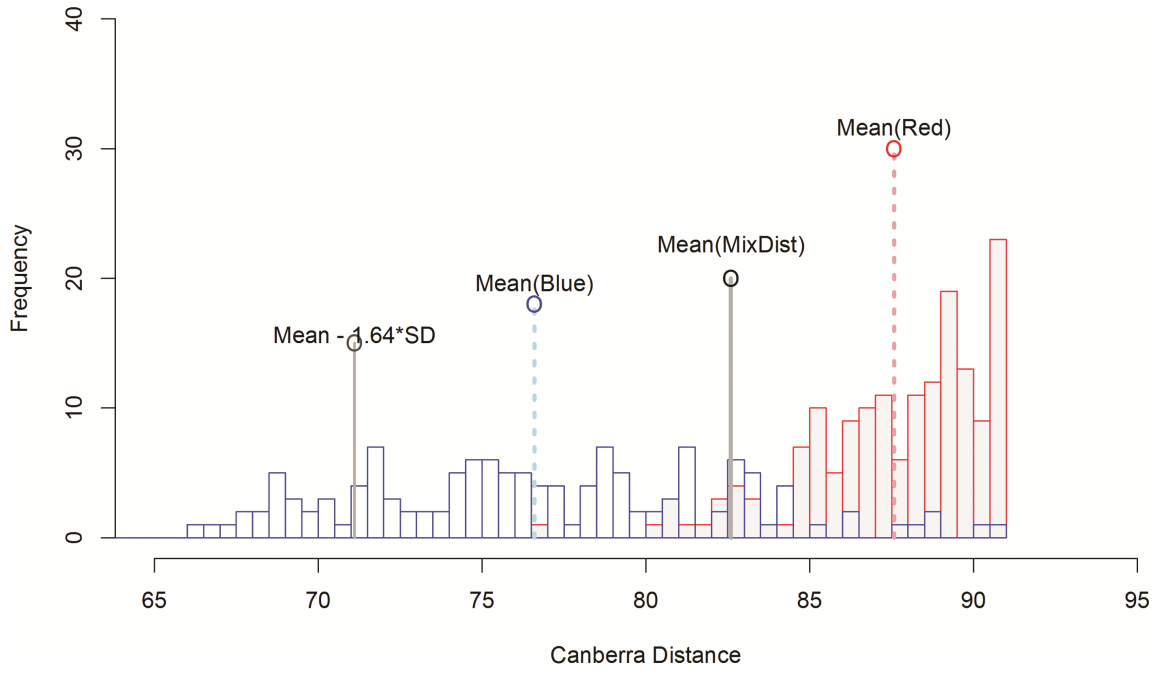


Fig. 1. Distribution of distances for a literary corpus. In white (blue) the correct links, and in grey (red) the wrong links. (Color figure online)

With the PAN data, we do not observe such a clear distinction between the two distance distributions. As an example, Fig. 2 visualizes the observed distributions of distances (on a logarithmic scale for the y-axis) in the Dutch Article corpus. This collection contains 57 documents out of which 20 have a single and unique author. From the remaining 37 documents, we should create one cluster of size two, three, four, six, and seven plus three clusters each containing five texts. Therefore, a total of 152 links (that is, 76 bidirectional links) must be created, out of the possible 3,192 links (57 * 57 – 57 in total, or 1,596 bidirectional links). Figure 2 is obtained when considering all link distance values. As we can see, there is an interleaving of the correct and incorrect links and the two means are almost identical.

Some texts are generally very close to many other texts, but they don't have sets of texts which are especially close to them. This results in a series of links that should be ignored. Then, there are texts that may be very far from some texts and very close to some other texts, meaning the link distance distribution has a large variance (or standard deviation). Again, those links should not be considered for a shared authorship due to the wide spread range of values. A correct authorship link could be detected if there are texts with a few link distance values that are substantially lower with respect to the text's general link distance distribution.

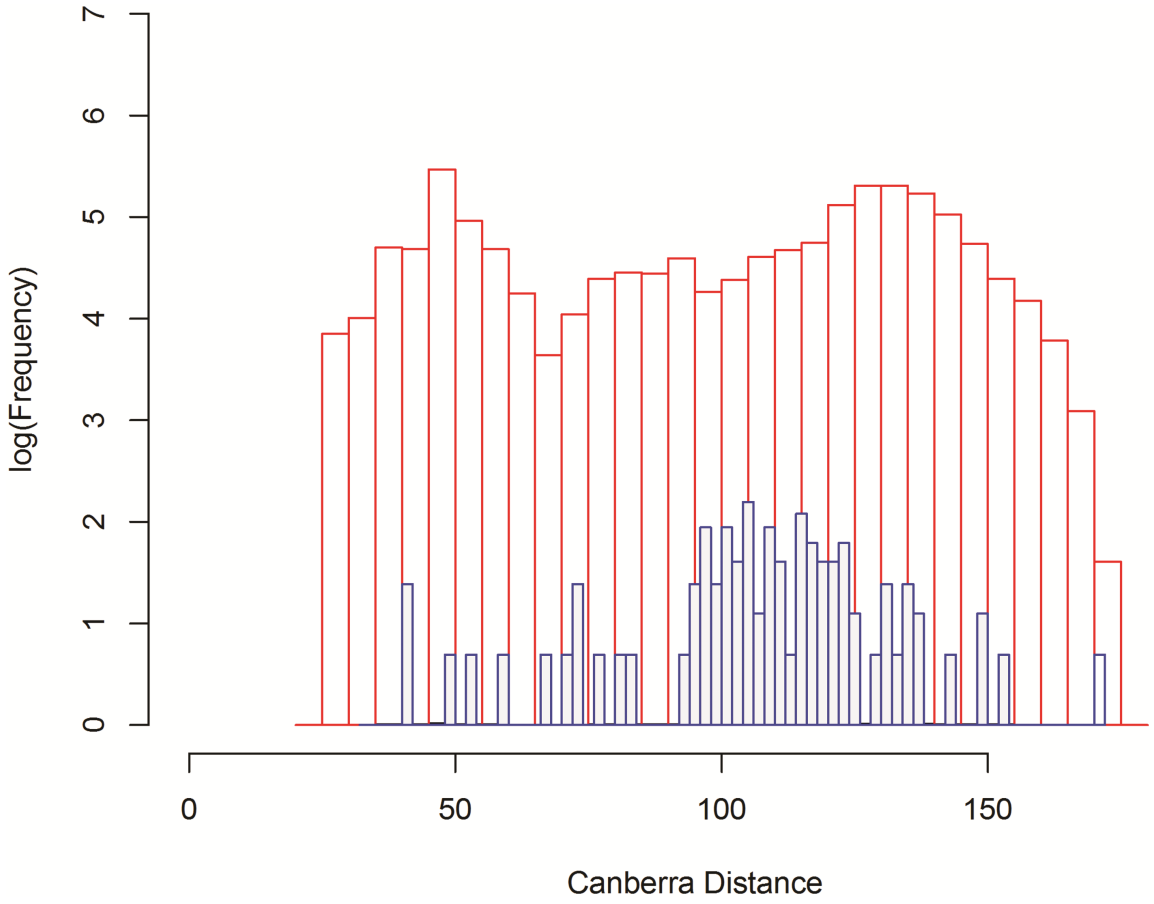


Fig. 2. Observed distribution of distance values (y-axis on a logarithmic scale) in the Dutch Article corpus (in dark (blue) the correct pairs, in white (red) incorrect pairs). (Color figure online)

As described in Sect. 4, we have two inequalities to determine when a distance value Δ_{AB} (or Δ_{BA}) can be viewed as "small" and thus hopefully reflecting a correct attribution. We use these limits to filter the distance values and to extract the more pertinent ones that are in the lower tail of a Gaussian distribution.

To generate the ranked list of links between two texts, the final attribution works as follows. After computing the distance values between all pairs of texts, we sort them from the smallest to the highest. Starting with the smallest (let's say Δ_{AB}), we also consider the opposite (Δ_{BA}). The link between the two texts is assigned in our Class 4 if the two distance values are smaller than the four limits (see Eqs. 2 to 5). If not, the link can be assigned to Class 3 (the two distances respect three limits), Class 2 (the two distances satisfy two limits), Class 1 (a single hint is available from the two distances), or Class 0 (the two distances are larger than the four limits).

To generate the final ranked list of links, we first consider Class 4. All links appearing in this group will obtain a probability of being correct between 1.0 and 0.8. For a given link, its probability depends on its position inside the Class 4. To define this position, we sort the links according to the sum of the two distance values (e.g., $\Delta_{AB} + \Delta_{BA}$) from the smallest to the largest. The smallest pair of distances will obtain the probability value of 1.0, the largest 0.8. The same sorting process is then applied for Class 3 (probability

range from 0.8 to 0.6), Class 2 (from 0.6 to 0.4), Class 1 (from 0.4 to 0.2), and Class 0 (from 0.2 to 0.0).

When inspecting the distribution over the five classes with the Dutch Articles corpus, we found no entries in Class 4 or 3, but 16 correct links in Class 2, 16 additional correct links and 32 incorrect links in Class 1. All remaining links (120 correct links, 3,008 incorrect ones) occur in Class 0.

For defining the clusters (performance measured by the BCubed F_1), we only take account of the links present in Class 4, 3 and 2. In our example with the Dutch Articles, only 16 (correct) links have been used. From them, we complement the clusters based on the present links. For example, having a link between Text C and D, and another link between Text C and F, we will generate the cluster {C, D, F}. All non-assigned texts will be considered as clusters with a single document.

To obtain a better understanding of the distance value when faced with pairs of text not written by the same author, we have inspected some examples from the English corpora. Usually, the relative frequency (or probability) differences with very frequent words such as *when*, *is*, *in*, *that*, *to*, or *it* as well as the usage of punctuation symbols can explain the decision. In other cases, the decision is mainly based on topical words like *European Union*, *wealth*, *history*, *language*, or *reader*. Therefore, using only the functional words does not seem to be an effective approach when facing short texts, as is the case with the PAN test collections.

8 Conclusion

This paper evaluates a simple unsupervised technique to solve the author clustering problem. As features to discriminate between the proposed author and different candidates, we propose using the top 200 most frequent terms (isolated words and punctuation symbols). This choice was found effective for other related tasks, such as in authorship attribution [2]. Moreover, compared to various feature selection strategies used in text categorization [15], the most frequent terms tend to select the most discriminative features when applied to stylistic studies [14]. To make the author linking decision, we propose using a simple distance measure based on the Spatium model using a variant of the L^1 norm (Canberra). This choice seems a good one compared to other possible distance functions (such as Euclidean, Cosine, or Dice) [9].

When using the CLEF PAN test collections, several parameters having a clear impact on the text style have been fixed, such as the time period, the text genre, or length of the data. This strategy tends to minimize the possible sources of variation in the corpus. The most challenging aspect of those test collections are the rather short lengths of the texts. In this context, our main objective is to present a simple and unsupervised approach without many predefined arguments.

With an adapted version of the Spatium algorithm [7], the proposed clustering system could be 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 could be clearer than when working with many features, dealing with numerous *n*-grams of letters or when combing several similarity measures. The Spatium

decision can be explained by major differences in relative frequencies of frequent words, usually corresponding to functional terms.

To improve the current version of our classifier, we need to analyze in more detail the distance measurement. The current version ignores the terms appearing once and replaces all uppercase letters with their corresponding lowercase ones. It could be checked if such decisions are pertinent when facing short texts. Moreover, we think that replacing the single link agglomerative clustering by the complete or average link will provide a more robust solution. Furthermore, such strategies will reduce the risk of the chaining effect present in the single link approach.

Acknowledgments. The authors want 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.

References

1. Amigo, E., Gonzalo, J., Artiles, J., Verdejo, F.: A comparison of extrinsic clustering evaluation metrics based on formal constraints. Inf. Retr. **12**(4), 461–486 (2009)
2. Burrows, J.F.: Delta: a measure of stylistic difference and a guide to likely authorship. Lit. Linguist. Comput. **17**(3), 267–287 (2002)
3. Craig, H., Kinney, A.F.: Shakespeare, Computers, and the Mystery of Authorship. Cambridge University Press, Cambridge (2009)
4. Hernández, D.M., Bécue-Bertaut, M., Barahona, I.: How scientific literature has been evolving over the time? A novel statistical approach using tracking verbal-based methods. In: JSM Proceedings, Section on Statistical Learning and Data Mining, Alexandria, pp. 1121–1131. American Statistical Association (2014)
5. Holmes, D.I.: The evolution of stylometry in humanities scholarship. Lit. Linguist. Comput. **13**(3), 111–117 (1998)
6. Jockers, M.L., Witten, D.M.: A comparative study of machine learning methods for authorship attribution. Lit. Linguist. Comput. **25**(2), 215–223 (2010)
7. Kocher, M., Savoy, J.: A simple and efficient algorithm for authorship verification. J. Am. Soc. Inf. Sci. Technol. **68**(1), 259–269 (2017)
8. Kocher, M., Savoy, J.: Author clustering using spatium. In: Proceedings of ACM/IEEE Joint Conference on Digital Libraries (2017, to appear)
9. Kocher, M., Savoy, J.: Distance measures in author profiling. Inf. Process. Manag. **53**(5), 1103–1119 (2017)
10. Labbé, D.: Experiments on authorship attribution by intertextual distance in English. J. Quant. Linguist. **14**(1), 33–80 (2007)
11. Layton, R., Watters, P., Dazeley, R.: Evaluating authorship distance methods using the positive silhouette coefficient. Nat. Lang. Eng. **19**, 517–535 (2013)
12. Manning, C.D., Raghaven, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
13. Savoy, J.: Estimating the probability of an authorship attribution. J. Am. Soc. Inf. Sci. Technol. **67**(6), 1462–1472 (2016)
14. Savoy, J.: Comparative evaluation of term selection functions for authorship attribution. Digit. Scholarsh. Hum. **30**(2), 246–261 (2015)

15. Sebastiani, F.: Machine learning in automatic text categorization. ACM Comput. Surv. **34**(1), 1–27 (2002)
16. Stamatatos, E., Tschuggnall, M., Verhoeven, B., Daelemans, W., Specht, G., Stein, B., Potthast, M.: Clustering by authorship within and across documents. In: Working Notes of the CLEF 2016 Evaluation Labs, CEUR Workshop Proceedings, CEUR-WS.org (2016)
17. Witten, I.H., Frank, E., Hall, M.A.: Data Mining. Practical Machine Learning Tools and Techniques. Morgan Kaufmann, Burlington (2011)
18. Zhao, Y., Zobel, J.: Searching with style: authorship attribution in classic literature. In: Proceedings of the Thirtieth Australasian Computer Science Conference, Ballarat, pp. 59–68 (2007)

Appendix B

Publications

B.1 Journal Articles

Mirco Kocher, Jacques Savoy.
Distributed Language Representation for Authorship Attribution.
In *Digital Scholarship in the Humanities*, to appear.

Mirco Kocher, Jacques Savoy.
Distance Measures in Author Profiling.
In *Information Processing and Management*, 53(5), 1103-1119, 2017.

Mirco Kocher, Jacques Savoy.
A Simple and Efficient Algorithm for Authorship Verification.
In *Journal of the American Society for Information Science and Technology*, 68(1), 259-269, 2017.

Mirco Kocher, Jacques Savoy.
Evaluation of Text Representation Schemes and Distance Measures for Authorship Linking.
In *Scientometrics (Special Issue Proposal on Scieno-Network-Mining)*, submitted.

B.2 Conference Proceedings

Mirco Kocher, Jacques Savoy.
Author Clustering with an Adaptive Threshold.
In *Jones, G. J. F., Lawless, S., Gonzalo, J., Kelly, L., Goeuriot, L., Thomas M., Cappellato, L., & Ferro, N. (Eds)*, Experimental IR Meets Multilinguality, Multimodality, and Interaction, Proceedings of the 8th International Conference of the CLEF Association, CLEF 2017, Dublin, Ireland, September 11-14, 2017, 186-198.

Mirco Kocher, Jacques Savoy.
Author Clustering Using Spatium.
Short Paper JCDL 2017, Toronto, Canada, June 19-23, 2017, ACM/IEEE, 265-268.

Mirco Kocher, Jacques Savoy.
Regroupement d'auteurs : Qui a écrit cet ensemble de romans ?
In *Nie, J.Y. & Lamprier, S.(Eds)*, Proceeding CORIA 2017, Marseille, France, March 29-31, 2017, ARIA, 311-326.

B.3 Evaluation Forums

Mirco Kocher, Jacques Savoy.
UniNE at CLEF 2017: Author Clustering - Notebook for PAN at CLEF 2017.
In *Capellato, L., Ferro, N., Goeuriot, L., & Mandl, T. (Eds)*, CLEF 2017 Labs Working Notes, Dublin, Ireland, September 11-14, 2017, Aachen: CEUR.

Mirco Kocher, Jacques Savoy.
UniNE at CLEF 2017: Author Profiling Reasoning - Notebook for PAN at CLEF 2017.
In *Capellato, L., Ferro, N., Goeuriot, L., & Mandl, T. (Eds)*, CLEF 2017 Labs Working Notes, Dublin, Ireland, September 11-14, 2017, Aachen: CEUR.

Mirco Kocher.
UniNE at CLEF 2016: Author Clustering - Notebook for PAN at CLEF 2016.
In *Balog, K., Capellato, L., Ferro, N., & Macdonald, C. (Eds)*, CLEF 2016 Labs Working Notes, Évora, Portugal, September 5-8, 2016, Aachen: CEUR.

Mirco Kocher, Jacques Savoy.
UniNE at CLEF 2016: Author Profiling - Notebook for PAN at CLEF 2016.
In *Balog, K., Capellato, L., Ferro, N., & Macdonald, C. (Eds)*, CLEF 2016 Labs Working Notes, Évora, Portugal, September 5-8, 2016, Aachen: CEUR.

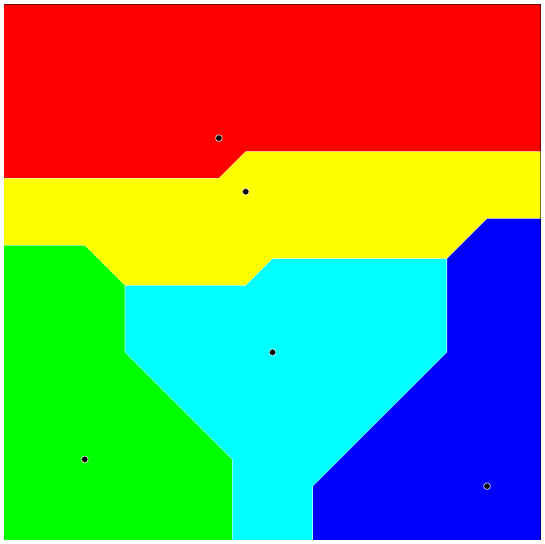
Mirco Kocher, Jacques Savoy.
UniNE at CLEF 2015: Author Identification - Notebook for PAN at CLEF 2015.
In *Capellato, L., Ferro, N., Jones, F. J. F., & Juan, E. S. (Eds)*, CLEF 2015 Labs Working Notes, Toulouse, France, September 8-11, 2015, Aachen: CEUR.

Mirco Kocher.
UniNE at CLEF 2015: Author Profiling - Notebook for PAN at CLEF 2015.
In *Capellato, L., Ferro, N., Jones, F. J. F., & Juan, E. S. (Eds)*, CLEF 2015 Labs Working Notes, Toulouse, France, September 8-11, 2015, Aachen: CEUR.

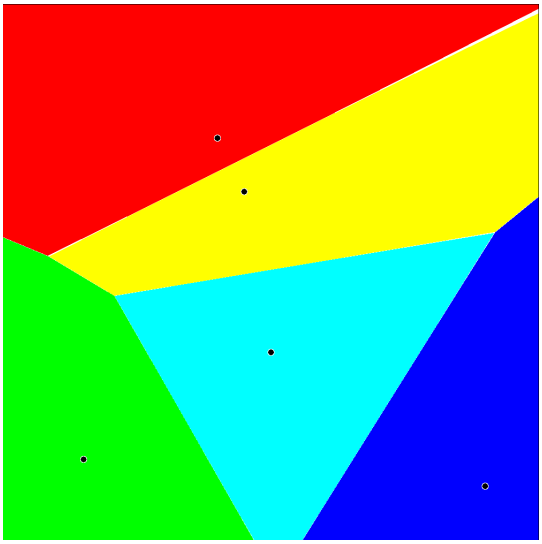
Appendix C

Voronoi Diagrams of Distance Measures

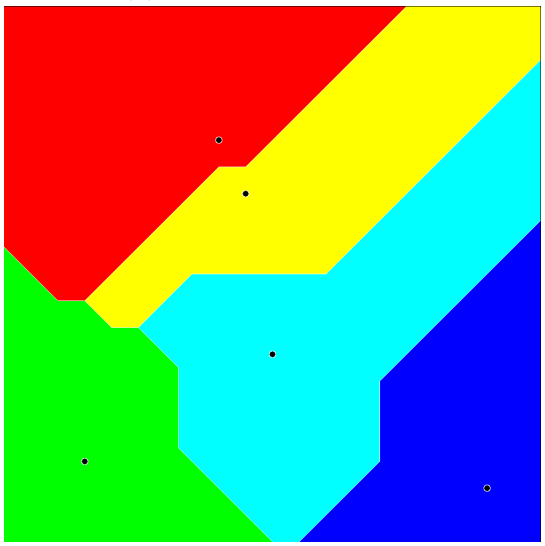
In the following figures, we have selected five dots as described in section 2.2. Every region presented with the same color represents points that are closest to one of the fixed five dots according to different distance functions.



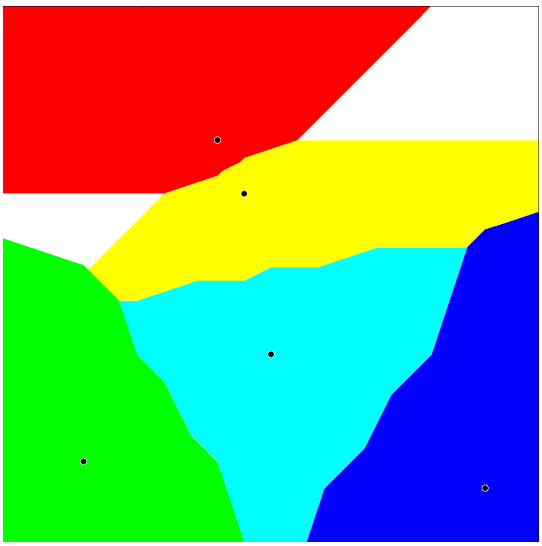
(a) Manhattan distance



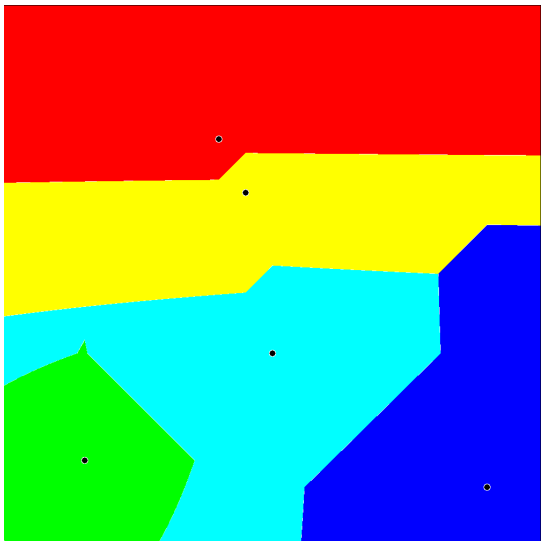
(b) Euclidean distance



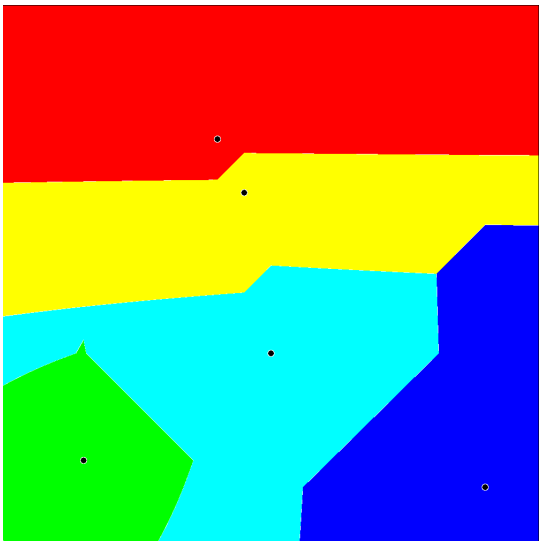
(c) Chebyshev distance



(d) Average distance

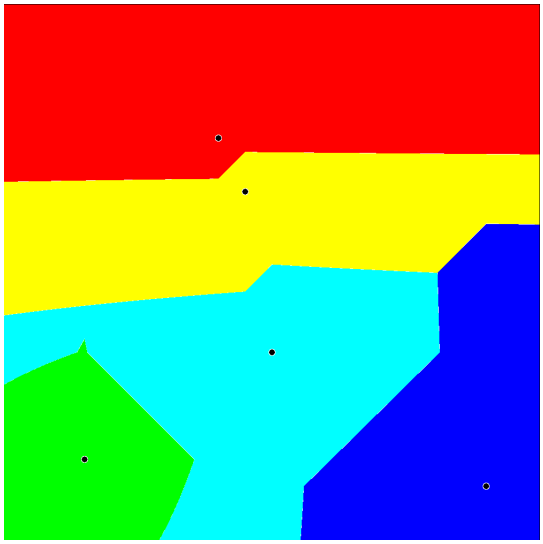


(e) Sørensen distance

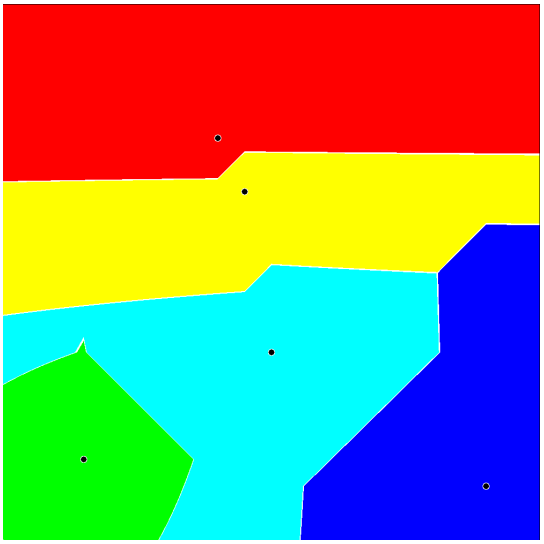


(f) Soergel distance

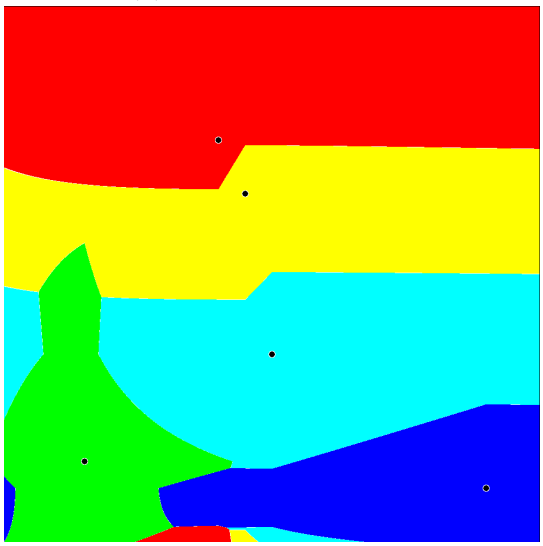
Figure C.1 – Voronoi Diagram using different distances.



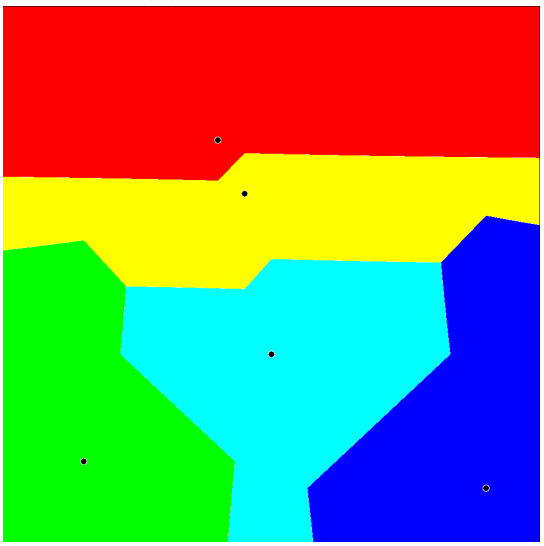
(a) Kulcynski distance



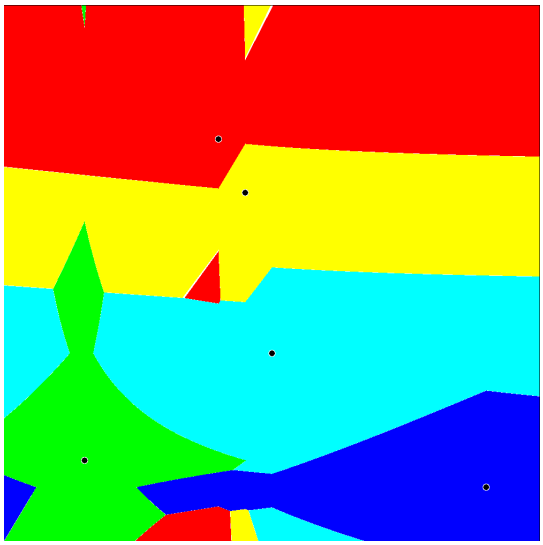
(b) Motyka distance



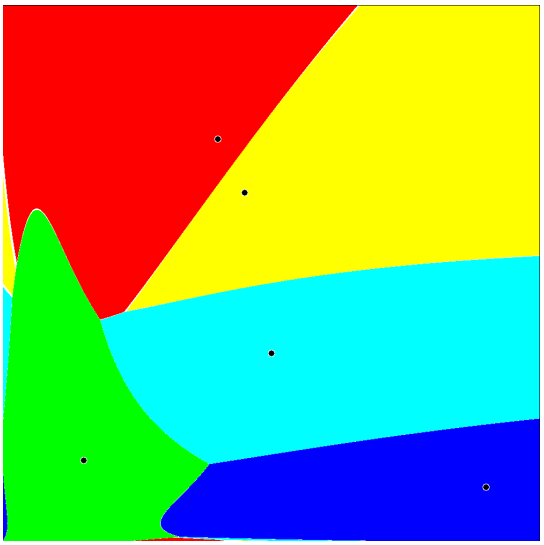
(c) Canberra distance



(d) Lorentzian distance

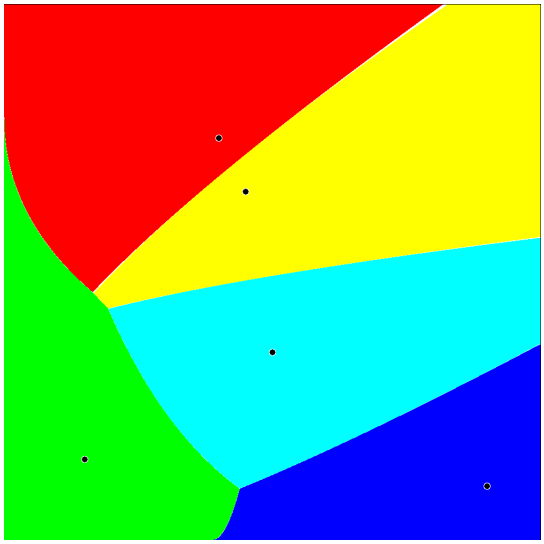


(e) Wave-Hedges distance

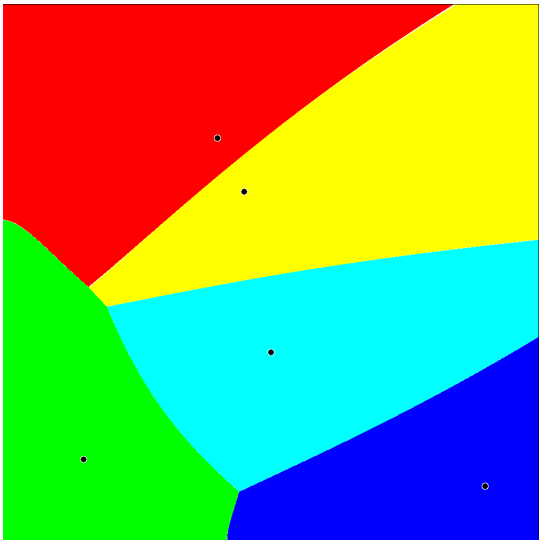


(f) Clark distance

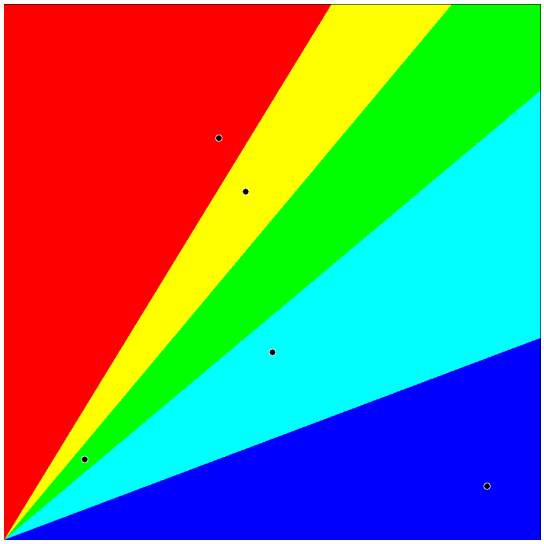
Figure C.2 – Voronoi Diagram using different distances.



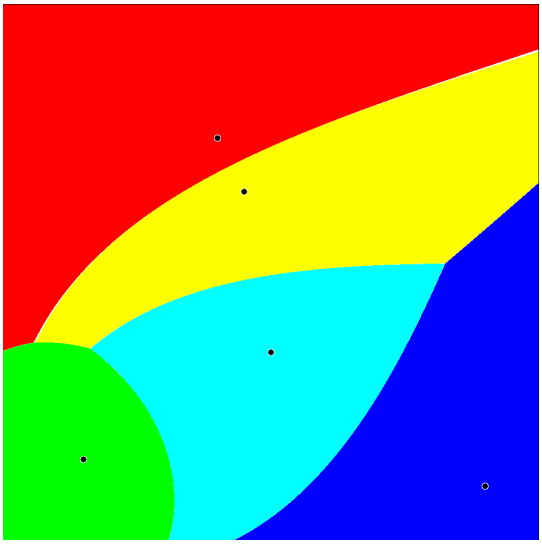
(a) Matusita distance



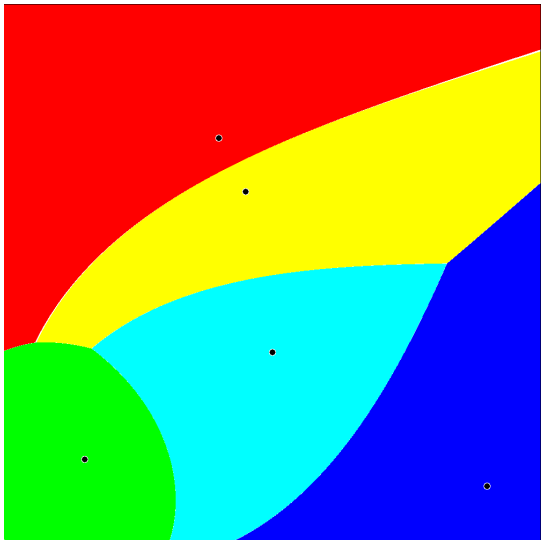
(b) Chi-Square (χ^2) distance



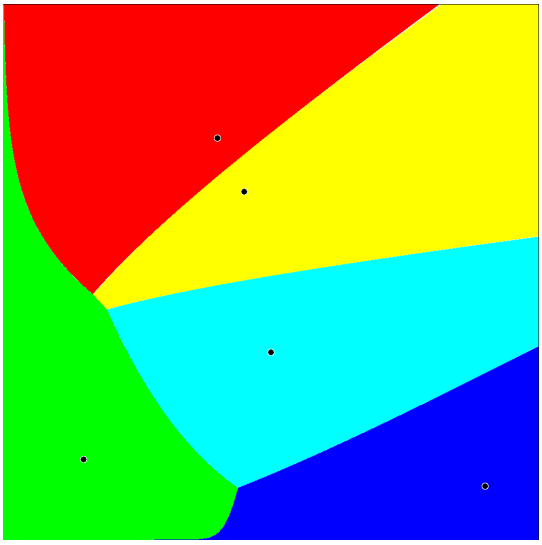
(c) Cosine distance



(d) Jaccard distance

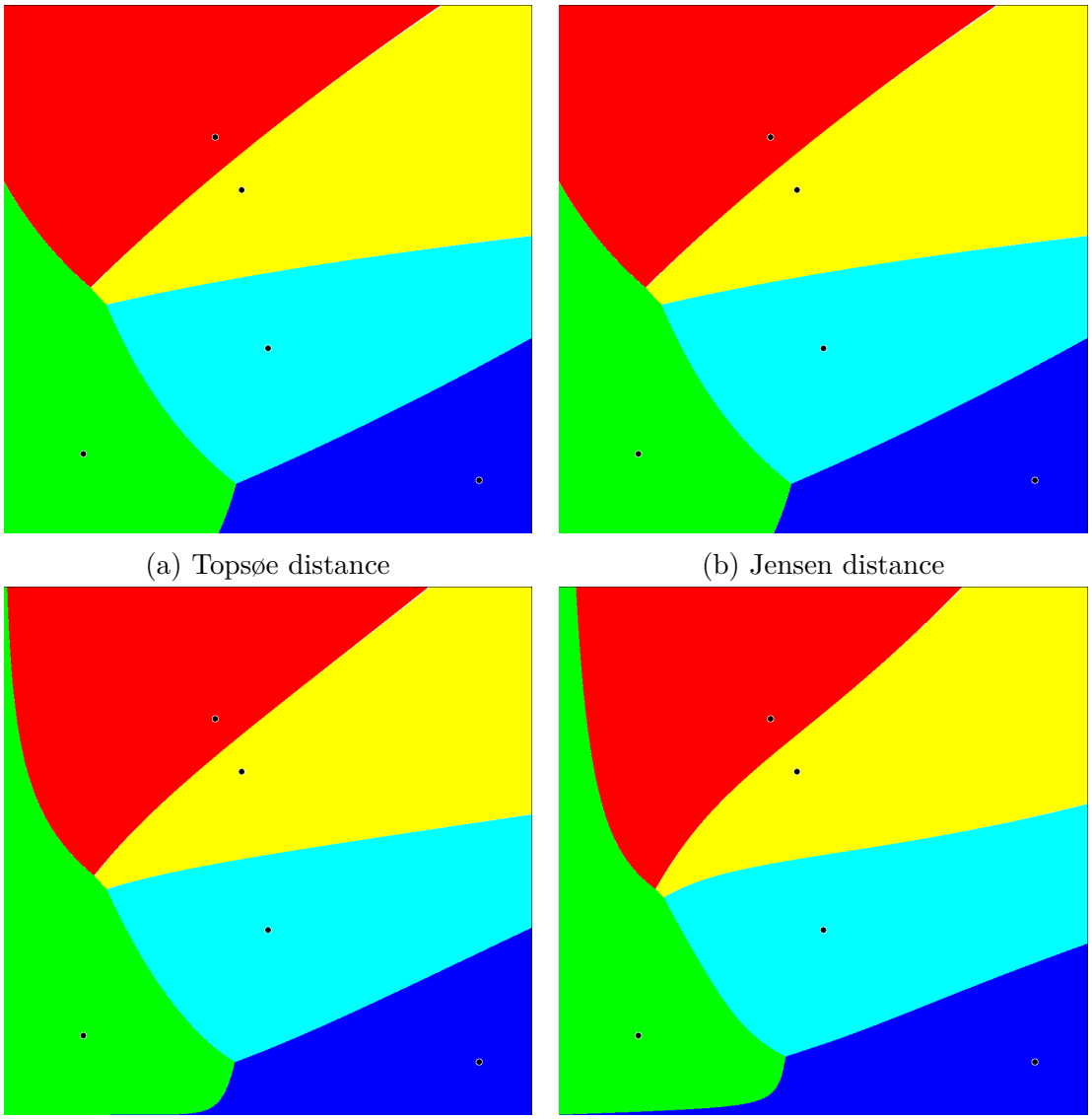


(e) Dice distance



(f) JDivergence distance

Figure C.3 – Voronoi Diagram using different distances.



(a) Topsøe distance

(b) Jensen distance

(c) Taneja distance

(d) Kumar-Johnson distance

Figure C.4 – Voronoi Diagram using different distances.

Bibliography

[1] Almishari, M., and Tsudik, G. Exploring Linkability of User Reviews. In *Computer Security - ESORICS 2012 - 17th European Symposium on Research in Computer Security, Pisa, Italy, September 10-12, 2012. Proceedings* (2012), S. Foresti, M. Yung, and F. Martinelli, Eds., vol. 7459, Springer, pp. 307–324.

[2] Amigó, E., Gonzalo, J., Artiles, J., and Verdejo, F. A Comparison of Extrinsic Clustering Evaluation Metrics Based on Formal Constraints. *Information Retrieval 12*, 4 (2009), 461–486.

[3] Argamon, S., Koppel, M., Pennebaker, J. W., and Schler, J. Automatically Profiling the Author of an Anonymous Text. *Communications of the ACM 52*, 2 (2009), 119–123.

[4] Baayen, H. R. *Analysis Linguistic Data: A Practical Introduction to Statistics Using R*. Cambridge University Press, 2008.

[5] Bengio, Y. Learning Deep Architectures for AI. *Foundations and Trends in Machine Learning 2*, 1 (2009), 1–127.

[6] Bessi, A., Coletto, M., Davidescu, G. A., Scala, A., Caldarelli, G., and Quattrociocchi, W. Science vs Conspiracy: Collective Narratives in the Age of Misinformation. *PLOS ONE 10*, 2 (02 2015), 1–17.

[7] Blackshaw, P. *Satisfied Customers Tell Three Friends, Angry Customers Tell 3,000 - Running a Business in Today's Consumer-Driven World*. Crown Business, 2008.

[8] Burrows, J. 'Delta': a Measure of Stylistic Difference and a Guide to Likely Authorship. *Literary and Linguistic Computing 17*, 3 (2002), 267–287.

[9] Burrows, J. All the Way Through: Testing for Authorship in Different Frequency Strata. *Literary and Linguistic Computing 22*, 1 (2007), 27–47.

[10] Chaski, C. Empirical Evaluations of Language-Based Author Identification Techniques. *International Journal of Speech Language and the Law 8*, 1 (2001), 1–65.

[11] Craig, H., and Kinney, A. *Shakespeare, Computers, and the Mystery of Authorship*. Cambridge University Press, 2009.

[12] Del Vicario, M., Bessi, A., Zollo, F., Petroni, F., Scala, A., Caldarelli, G., Stanley, H. E., and Quattrociocchi, W. The Spreading of Misinformation Online. *Proceedings of the National Academy of Sciences 113*, 3 (2016), 554–559.

[13] Duda, R. O., Hart, P. E., and Stork, D. G. *Pattern Classification, 2nd Edition.* Wiley, 2001.

[14] Goldberg, Y. Neural Network Methods for Natural Language Processing. *Synthesis Lectures on Human Language Technologies 10*, 1 (2017), 1–309.

[15] Goodfellow, I. J., Bengio, Y., and Courville, A. C. *Deep Learning.* Adaptive computation and machine learning. MIT Press, 2016.

[16] Grieve, J. Quantitative Authorship Attribution: An Evaluation of Techniques. *Literary and Linguistic Computing 22*, 3 (2007), 251–270.

[17] Holmes, D. I. The Evolution of Stylometry in Humanities Scholarship. *Literary and Linguistic Computing 13*, 3 (1998), 111–117.

[18] Hoover, D. L. Delta Prime? *Literary and Linguistic Computing 19*, 4 (2004), 477–495.

[19] Jockers, M. L., and Witten, D. M. A Comparative Study of Machine Learning Methods for Authorship Attribution. *Literary and Linguistic Computing 25*, 2 (2010), 215–223.

[20] Kešelj, V., Peng, F., Cercone, N., and Thomas, C. N-Gram-Based Author Profiles for Authorship Attribution. In *Proceedings of the Conference Pacific Association for Computational Linguistics, PACLING* (2003), vol. 3, pp. 255–264.

[21] Kjell, B. Authorship Determination Using Letter Pair Frequency Features with Neural Network Classifier. *Literary and Linguistic Computing 9*, 2 (1994), 119–124.

[22] Kocher, M. UniNE at CLEF 2015: Author Profiling. In *Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015.* (2015), L. Cappellato, N. Ferro, G. J. F. Jones, and E. SanJuan, Eds., vol. 1391, CEUR-WS.org.

[23] Kocher, M. UniNE at CLEF 2016: Author Clustering. In *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, Évora, Portugal, 5-8 September, 2016.* (2016), K. Balog, L. Cappellato, N. Ferro, and C. Macdonald, Eds., vol. 1609, CEUR-WS.org, pp. 895–902.

[24] Kocher, M., and Savoy, J. UniNE at CLEF 2015: Author Identification. In *Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015.* (2015), L. Cappellato, N. Ferro, G. J. F. Jones, and E. SanJuan, Eds., vol. 1391, CEUR-WS.org.

[25] Kocher, M., and Savoy, J. UniNE at CLEF 2016: Author Profiling. In *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, Évora, Portugal, 5-8 September, 2016.* (2016), K. Balog, L. Cappellato, N. Ferro, and C. Macdonald, Eds., vol. 1609, CEUR-WS.org, pp. 903–911.

[26] Kocher, M., and Savoy, J. UniNE at CLEF 2017: Author Clustering. In *Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum, Dublin, Ireland, September 11-14, 2017.* (2017), L. Cappellato, N. Ferro, L. Goeuriot, and T. Mandl, Eds., vol. 1866, CEUR-WS.org.

[27] Kocher, M., and Savoy, J. UniNE at CLEF 2017: Author Profiling Reasoning. In *Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum, Dublin, Ireland, September 11-14, 2017.* (2017), L. Cappellato, N. Ferro, L. Goeuriot, and T. Mandl, Eds., vol. 1866, CEUR-WS.org.

[28] Koppel, M., Schler, J., and Bonchek-Dokow, E. Measuring Differentiability: Unmasking Pseudonymous Authors. *Journal of Machine Learning Research 8* (2007), 1261–1276.

[29] Labbé, C., and Labbé, D. A Tool for Literary Studies: Intertextual Distance and Tree Classification. *Literary and Linguistic Computing 21*, 3 (2006), 311–326.

[30] Labbé, D. Experiments on Authorship Attribution by Intertextual Distance in English. *Journal of Quantitative Linguistics 14*, 1 (2007), 33–80.

[31] Ledger, G., and Merriam, R. Shakespeare, Fletcher, and the Two Noble Kinsmen. *Literary and Linguistic Computing 9*, 3 (1994), 235–248.

[32] Love, H. *Attributing Authorship: An Introduction.* Cambridge University Press, 2002.

[33] Manning, C. D., Raghavan, P., and Schütze, H. *Introduction to Information Retrieval.* Cambridge University Press, 2008.

[34] Mcnamee, P., and Mayfield, J. Character N-Gram Tokenization for European Language Text Retrieval. *Information Retrieval 7*, 1-2 (2004), 73–97.

[35] Mehri, A., Darooneh, A. H., and Shariati, A. The Complex Networks Approach for Authorship Attribution of Books. *Physica A: Statistical Mechanics and its Applications 391*, 7 (2012), 2429–2437.

[36] Munroe, R. *Thing Explainer: Complicated Stuff in Simple Words.* Hodder & Stoughton, 2015.

[37] Olsson, J. *Forensic Linguistics: Second Edition: An Introduction To Language, Crime and the Law.* Bloomsbury Academic, 2008.

[38] Patrick, J. Authorship Attribution. *Foundations and Trends in Information Retrieval 1*, 3 (2006), 233–334.

[39] Peñas, A., and Rodrigo, Á. A Simple Measure to Assess Non-Response. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA* (2011), D. Lin, Y. Matsumoto, and R. Mihalcea, Eds., The Association for Computer Linguistics, pp. 1415–1424.

[40] Pennebaker, J. *The Secret Life of Pronouns: What Our Words Say About Us.* Bloomsbury USA, 2011.

[41] Potthast, M., Hagen, M., and Stein, B. Author Obfuscation: Attacking the State of the Art in Authorship Verification. In *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, Évora, Portugal, 5-8 September, 2016.* (2016), pp. 716–749.

[42] Rosso, P., Rangel, F. M., Potthast, M., Stamatatos, E., Tschuggnall, M., and Stein, B. Overview of PAN'16 - New Challenges for Authorship Analysis: Cross-Genre Profiling, Clustering, Diarization, and Obfuscation. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction - 7th International Conference of the CLEF Association, CLEF 2016, Évora, Portugal, September 5-8, 2016, Proceedings* (2016), pp. 332–350.

[43] Savoy, J. Comparative Evaluation of Term Selection Functions for Authorship Attribution. *Digital Scholarship in the Humanities 30*, 2 (2015), 246–261.

[44] Savoy, J. Estimating the Probability of an Authorship Attribution. *Journal of the Association for Information Science and Technology 67*, 6 (2016), 1462–1472.

[45] Savoy, J. Text Representation Strategies: An Example with the State of the Union Addresses. *Journal of the Association for Information Science and Technology 67*, 8 (2016), 1858–1870.

[46] Savoy, J. Analysis of the Style and the Rhetoric of the American Presidents Over Two Centuries. *To appear* (2017).

[47] Schler, J., Koppel, M., Argamon, S., and Pennebaker, J. W. Effects of Age and Gender on Blogging. In *Computational Approaches to Analyzing Weblogs, Papers from the 2006 AAAI Spring Symposium, Technical Report SS-06-03, Stanford, California, USA, March 27-29, 2006* (2006), Association for the Advancement of Artificial Intelligence, pp. 199–205.

[48] Solan, L. M., and Tiersma, P. M. Author Identification in American Courts. *Applied Linguistics 25*, 4 (2004), 448–465.

[49] Stamatatos, E. A Survey of Modern Authorship Attribution Methods. *Journal of the Association for Information Science and Technology 60*, 3 (2009), 538–556.

[50] Stamatatos, E., Tschuggnall, M., Verhoeven, B., Daelemans, W., Specht, G., Stein, B., and Potthast, M. Clustering by Authorship Within and Across Documents. In *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, Évora, Portugal, 5-8 September, 2016.* (2016), K. Balog, L. Cappellato, N. Ferro, and C. Macdonald, Eds., vol. 1609 of *CEUR Workshop Proceedings*, CEUR-WS.org, pp. 691–715.

[51] Stover, J. A., Winter, Y., Koppel, M., and Kestemont, M. Computational Authorship Verification Method Attributes a New Work to a Major

2nd Century African Author. *Journal of the Association for Information Science and Technology 67*, 1 (2016), 239–242.

[52] Tausczik, Y. R., and Pennebaker, J. W. The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods. *Journal of Language and Social Psychology 29*, 1 (2010), 24–54.

[53] Thisted, R., and Efron, B. Did Shakespeare Write a Newly-Discovered Poem? *Biometrika 74*, 3 (1987), 445–455.

[54] Zipf, G. K. *The Psychobiology of Language: An Introduction to Dynamic Philology.* M.I.T. Press, Cambridge, Mass., 1935.

[55] Zipf, G. K. *Human Behaviour and the Principle of Least Effort: an Introduction to Human Ecology.* Addison-Wesley, 1949.