Filtering a Reference Corpus to Generalize Stylometric Representations

Julien Hay[1,2,3], Bich-Liên Doan[2,3], Fabrice Popineau[2,3] and Ouassim Ait Elhara[1]
[1]*Octopeek SAS, 95880 Enghien-les-Bains, France*
[2]*Laboratoire de Recherche en Informatique, Paris-Saclay University, 91190 Gif-sur-Yvette, France*
[3]*CentraleSupélec, Paris-Saclay University, 91190 Gif-sur-Yvette, France*

Keywords: Writing Style, Authorship Analysis, Representation Learning, Deep Learning, Filtering, Preprocessing.

Abstract: Authorship analysis aims at studying writing styles to predict authorship of a portion of a written text. Our main task is to represent documents so that they reflect authorship. To reach the goal, we use these representations for the authorship attribution, which means the author of a document is identified out of a list of known authors. We have recently shown that style can be generalized to a set of reference authors. We trained a DNN to identify the authors of a large reference corpus and then learnt how to represent style in a general stylometric space. By using such a representation learning method, we can embed new documents into this stylometric space, and therefore stylistic features can be highlighted. In this paper, we want to validate the following hypothesis: the more authorship terms are filtered, the more models can be generalized. Attention can thus be focused on style-related and constituent linguistic structures in authors' styles. To reach this aim, we suggest a new efficient and highly scalable filtering process. This process permits a higher accuracy on various test sets on both authorship attribution and clustering tasks.

1 INTRODUCTION

Among the most commonly addressed tasks in this field of authorship analysis, there is the authorship attribution and authorship verification. The authorship attribution is the process of guessing the author of documents among known authors while the authorship verification is the process of deciding whether or not a given document was written by a given author. To this end, most studies rely on feature engineering to represent the input documents in order to improve the performance of machine learning algorithms. One common way to choose these features is by assessing whether or not they can enhance the prediction accuracy. Sometimes these features intuitively belong to style such as function words (Goldstein-Stewart et al., 2009, Menon and Choi, 2011), sometimes they just correspond to common NLP features such as character n-grams (Escalante et al., 2011, Stamatatos, 2007) or distributional representations of documents (Chen et al., 2017, Gupta et al., 2019, Bagnall, 2015).

(Karlgren, 2004) defined the style as "a consistent and distinguishable tendency to make [some of these] linguistic choices". Moreover, (Karlgren, 2004) explained that "texts are much more than what they are about". Any textual difference that is not semantic nor topical belongs to stylistic choices of the author. Different expressions can have a common meaning, and can refer to the same objects and the same events, but still be made up of different words and different syntax, corresponding to the author's willingness to let a context, an orientation, sometimes an emotion be shown through (Argamon et al., 2005).

Not only is it difficult to identify precisely which characteristics fall within the scope of writing style (Bischoff et al., 2020), but it is also difficult to extract textual features that do not capture topical aspects at the same time (Stamatatos, 2018) since topical aspects allow to better predict authorship in some cases (Seroussi et al., 2014). However, under some specific cross-domain scenario (e.g. topic and genre), such features do not help, which is why recent studies propose text distortion methods that mask topic and genre terms in order to improve author analysis (Stamatatos, 2018, Stamatatos, 2017, Halvani et al., 2020).

To alleviate these issues, we proposed in (Hay et al., 2020) to train a general style model by relying on a large reference corpus in order to project unseen documents in a low dimensional stylometric space defined by reference authors. Our representation learning method proved to enhance accuracy on

Hay, J., Doan, B., Popineau, F. and Elhara, O.
Filtering a Reference Corpus to Generalize Stylometric Representations.
DOI: 10.5220/0010138802590268
In *Proceedings of the 12th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2020) - Volume 1: KDIR*, pages 259-268
ISBN: 978-989-758-474-9

the authorship clustering and the authorship attribution tasks. This led us to propose a new definition of writing style based on distributional properties.

Style appears more or less pronounced depending on the text passages, it is difficult to define it precisely and, given a document, to find a set of words (or sequence of words) that will strictly define the style of its author. The text is the combination of a shape – its style – and a content which are intertwinned thanks to the choice of specific words. Words or sequence of words in the text can rarely be denoted as belonging specifically to the style or to the content. This is why extracting style features is hard. From documents of the reference corpus, we aim to extract latent structures falling within the scope of writing style. We argue that these latent structures can be identified by DNNs, typically RNN models with attention layers which will focus on style-related terms. From a linguistic point of view, these latent structures map to lexical, syntactic or structural fragment of sentences or paragraphs.

Intuitively, when extracting a style representation of a document, we seek to focus on latent structures that will satisfy these two properties :

Intra-author Consistency. the property of being consistent in documents belonging to the same author.

Semantic Undistinguishness. the property of carrying very little information on what makes the document semantically (e.g. topics, named entities) distinguishable in the corpus.

Thus, this definition, inspired by (Karlgren, 2004, Holmes, 1998), means that the style of a document is represented by linguistic structures which are consistent for individual authors (allowing their identification) but more likely semantically poor regarding the content of the document (e.g. topic, named entities). Indeed, what the document is about is a constraint that imposes on the author to use a specific vocabulary. The terms that belong to this specific vocabulary have a strong semantic value with respect to the theme of the document, and on the contrary, are less likely to convey the author's style. The representation learning method is based on identifying consistent latent structures following the *intra-author consistency* property. Next to that, the *semantic undistinguishness* is a property which can be verified by studying attention weights of a trained DNN models. Moreover, the filtering process we present in this paper aims at enforcing this property for terms the trained DNN focus on.

In this article, we seek to validate the *filtering assumption* stating that removing the most informative sentences about the identity of authors in the reference corpus (i.e. containing the most author-consistent sequences of words) allows to enhance our representation learning method in adequacy with the *semantic undistinguishness*. The most informative sentences are those containing author-specific word sequences, i.e. word sequences that are used frequently by one author and very little by the rest of the authors in the corpus. For this purpose, we propose a filtering process based on the TFIDF weighting which is designed to remove terms which are too peculiar from certain authors of the reference corpus. Targeted terms are those having a high frequency in documents of individual authors and having a low inverse document frequency, i.e. those that are rare in the corpus.

This filtering process is to be dissociated from our definition of style since it does not consist in eliminating or preserving an author's writing style. Indeed, it consists in the removal of sentences that allow easy identification of authors in the context of the authorship attribution task. In the absence of terms allowing to spot the author easily, the DNN model will be forced to focus on more subtle terms to identify the author. With the *filtering assumption*, we suppose it will allow to better learn to capture stylometric representations on the basis of reference authors.

The reference corpus is typically large and the TFIDF computation can be very time consuming when the entire vocabulary needs to be taken into account, which is one of our requirements because we want to exhaustively find the most informative terms. Moreover the reference corpus can contain very unbalanced classes (i.e. classes having a lot of documents compared to others), which can be problematic when computing TFIDF and choosing a TFIDF threshold. Thus we propose a method that alleviate these issues by making a set of balanced buckets on which we will find the most informative terms (or sequences of terms) about authors independently, then merge all these terms to process to the final corpus filtering. We will come back on the filtering process and its requirements in Section 3.

The rest of this paper is organized as follows. Section 2 gives the related work on filtering and masking methods in authorship analysis. Section 3 will formalize the *filtering assumption* and give an overview on the method we propose. Section 4 describes the implementation of the proposed filtering process. Section 5 presents the results obtained with and without the use of the filtered reference corpus on the authorship clustering and attribution tasks. Finally, in Section 6, we conduct a deep analysis on the *semantic undistinguishness* property.

2 RELATED WORK

In NLP tasks, it is common to perform a first step of text preprocessing (e.g. lemmatization and stop word removal) in order to eliminate irrelevant parts of the text or to highlight relevant features (Lourdusamy and Abraham, 2018).

In authorship analysis, (Stamatatos, 2017) introduced a text distortion method aiming to replace content word which are less frequent in the corpus by special tokens. This technique was originally used to mask frequent words and improve the accuracy of text classification (Granados et al., 2011). But for the authorship analysis, the goal is to mask topic- and genre-related words that do not express the author's writing style. The advantage of masking is that the structure of the sentences is preserved, unlike other preprocessing methods such as the removal of stop words for example. This technique has been shown to achieve better results in authorship attribution, especially in cross-domain situations when the topic or genre of the authors changes between the train set and the test set (Stamatatos, 2017, Stamatatos, 2018).

Similarly, (Halvani et al., 2020) proposed POSNoise, a preprocessing step aiming to mask topic-related text units in documents. Each topic-related text units is replaced by its part-of-speech tag. They showed that the POSNoise get higher scores than the text distorsion of (Stamatatos, 2018) in authorship verification on various datasets. The goal of these methods is to preprocess corpora in order to make documents representation of an author robust to topic and genre shifts.

The difference with the method we propose is that we do not target topic- or genre-related terms in general but terms that are too specific of an author in the reference corpus. The final goal is slightly different, we seek to make the identification of an author more difficult in order to train a DNN able to capture subtle and consistent structures in the text and not relying on overly obvious sequence of words about authorship. This method is therefore not intended to directly improve model performance in author analysis on a specific dataset with known authors but to filter a reference corpus to better capture style features of unknown authors.

3 THE FILTERING ASSUMPTION AND MOTIVATIONS

Let us denote $D = \{d_1, ..., d_n\}$ a set of documents and $A = \{a_1, ..., a_m\}$ a set of authors so that each document belongs to one and only one author and each author wrote at least one document. Let us denote *R-set* $= (D^r, A^r)$ the reference set with $D^r \subset D$, $A^r \subset A$, $|D^r| = n^r$, $|A^r| = m^r$ and A^r is the set of all documents authors in D^r. Both n^r and m^r are typically large. Let us denote *U-set* $= (D^u, A^u)$ a set of unseen documents and unseen authors with $D^u \subset D$, $A^u \subset A$, $|D^u| = n^u$, $|A^u| = m^u$ and A^u is the set of all the authors of the documents in D^u. $A^r \cap A^u = \emptyset$ and $D^r \cap D^u = \emptyset$.

The *style-generalization* assumption states that the projection of documents of D^u (the *U-set*) by a DNN model trained to identify authors of *R-set* documents allows to compute similarities such that similar documents from D_u are likely written by the same author. Intuitively, it states that the style of any author can be generalized on the basis of the style of reference authors. We validated this assumption by using representations from intermediate layers of DNN models trained on the *R-set* (authorship attribution task). These embeddings showed to better represent *U-sets* documents by authorship than other standard models, but also allowed to improve the performance of a SVM on the authorship attribution task. Thus, learning a DNN on a reference set allows authorship clustering in the general stylometric space that it defines (Hay et al., 2020). DNNs we implemented are a bi-LSTM network with an attention layer and a pre-trained BERT-based model fine-tuned on the *R-set* (Sanh et al., 2019). By adding a softmax layer on top of each DNN, we trained them to identify the 1200 reference authors of the *R-set*. Then, we extracted embeddings of unseen documents from the *U-sets* by taking the outputs of the attention layer of both DNNs. Both DNNs were implemented with *TensorFlow* (Abadi et al., 2015). More details are presented in (Hay et al., 2020).

In this article, we seek to validate *filtering assumption* stating that removing sentences which include too obvious terms enabling the identification of an author in the reference corpus allows to train a model that better generalizes the style and thus:

1. allows to better embed new documents with the aim of improving performance in the authorship clustering and attribution tasks ;
2. allows to focus less on semantic words but more on function words, in adequacy with the *semantic undistinguishness*.

The intuition is that the trained DNN will generalize the style by focusing on most subtle terms reflecting the author's writing style, i.e. on more frequent terms (e.g. function words) that will most likely fit the writing style of unseen authors. These terms are also terms that do not allow the document to be distinguishable in the corpus which meet the *semantic*

undistinguishness property.

In order to validate the hypothesis, we need a filtered *R-set* and the original one. We then evaluate two DNNs, each trained on a version of the *R-set*, on the authorship clustering and the authorship attribution tasks on different *U-sets*. The reference corpus needs to be large for the DNN model to capture stylometric latent structures of reference authors, thus the filtering process must allow to distribute the TFIDF computation and handle special cases such as highly unbalanced classes, i.e. classes having a lot of documents compared to others. Moreover, the filtering process needs to take into account the entire vocabulary of the reference corpus to exhaustively eliminate targeted terms to prevent inadvertently leaving overly obvious terms. Thus, in the first step, we make several buckets of documents[1] in order to distribute the computation of TFIDF weights instead of use a distributed term frequency computation based on feature hashing for instance (Weinberger et al., 2009).

Filtering the *R-set* involves three steps:

1. The generation of buckets, each having a limited number of classes (referring to author labels), a limited number of documents but with a balanced total of tokens per class.
2. For each of these buckets, the computation of the TFIDF weights of 1-grams, 2-grams and 3-grams vocabularies on class-documents[2] of each class in the bucket.
3. For each of these buckets, the extraction of the *n*-grams that are most indicative of their class, i.e. having a high TFIDF weight. We select these *n*-grams, which we call black *n*-grams, using a threshold on the TFIDF weights. We choose the threshold so that when we delete the sentences containing a black *n*-grams, a certain ratio of the sentences in the bucket is deleted. This ratio is a parameter that we define in advance.

We make every bucket balanced in order to avoid having class-documents that are too large compared to others because of the possible imbalance between classes. However, the buckets must be large enough for the vocabulary to be representative of the entire corpus. In the second step, in addition to 1-grams, we choose to also take into account 2-grams and 3-grams in order to be able to identify word sequences that expose the authorship.

We choose to eliminate entire sentences, not just the *n*-grams, in order to preserve the sentence structure. We also choose not to mask the *n*-grams to make the *R-set* and *U-sets* inputs consistent. In addition, the deletion of sentences allows to remove repeated pieces of text from certain authors, such as conditions of use or invitations to comment the article. We consider that such sentences are not relevant for the representation of style.

[1] A bucket is a subset of documents belonging to several authors in the *R-set*.

[2] The class-document of a class is the concatenation of all documents belonging to the class.

4 THE FILTERING PROCESS

The dataset we have to filter is composed of documents each belonging to its author's class. Each document is tokenized into sentences and words. At the end of this procedure, we aim to obtain a filtered *R-set*.

Algorithm 1 allows to make buckets of documents with a sufficient number of tokens yet balanced per class. This algorithm takes as input a *TokensCount-structure* (abbreviated *TC-struct*) *r* (for "remaining") which map classes to the identifiers of its documents with the number of tokens of the document. The parameter *r* is thus an initial *TC-struct* containing the whole corpus. The algorithm also takes a predefined number *maxT* denoting the maximum number of tokens each bucket can contain. The algorithm returns a list of *TC-struct* in the variable $buckets$ on which we will extract black *n*-grams. The norm of a *TC-struct*, for instance $|r|$, denotes the total number a tokens it contains.

Algorithm 1: Documents distribution.

```
1:  procedure DocDist(r : TC, maxT : integer, vr : float)
2:      s ← new empty TC-struct
3:      buckets ← ∅
4:      while r is not empty do
5:          bucket ← makeBucket(r, s, maxT)
6:          ok ← isValidBucket(bucket, vr)
7:          changed = false
8:          if ok then
9:              newR, newS ← copy of r, s
10:             Adding bucket's ids in newS
11:             Removing bucket's ids from newR
12:             changed ← |newR| − |r| ≠ 0
13:             if changed then
14:                 buckets ← buckets ∪ {bucket}
15:                 r, s ← newR, newS
16:         if ¬(ok ∧ changed) then
17:             r ← prune(r, bucket)
18:     return buckets
19: end procedure
```

The function *makeBucket* selects documents in r as well as in s. It returns a *TC-struct* which correspond to a bucket. The *TC-struct s* (for "selected") is intended to remember which documents were selected by previous iterations. We use s because it is sometimes necessary to select already selected documents in order to balance the bucket. The function first chooses a certain number of classes by following two heuristics:

1. prioritizing the selection of r classes having the fewest tokens ;
2. when necessary, adding classes of s by prioritizing the classes with the most tokens in order to facilitate subsequent balancing.

The selection of documents from each class is then carried out randomly with several trials prioritizing the selection of r documents. We retain the selection of documents with a number of tokens closest to $maxT$.

The function *isValidBucket* line 6 checks the balance of the current bucket. It returns $false$ when one of the class-documents in the current bucket has too many or too few tokens compared to other class-documents. When calling *DocDist*, we set the parameter *vr*. This parameter is a variation ratio allowing to calculate the range of tokens count each class-documents must contain for the bucket to be valid. The range is calculated on the basis of the average tokens per class-document and the variation ratio *vr*. The function *prune* line 17 removes the longest document and the shortest document in the class that has the largest deviation from the average in the current *bucket*. This ensures the convergence of the algorithm by preventing the selection of documents that do not allow a proper balancing of the bucket. This pruning is performed if no documents of r have been removed or the current bucket is invalid.

Algorithm 2: Black n-grams generation.

```
1: procedure GENBLACKNG(b, minN, maxN, d)
2:     cd ← generate class-documents of b
3:     weights, cumD ← new dictionaries
4:     for n ← minN to maxN do
5:         weights[n] ← tfidf(cd, n)
6:         cumD[n] ← compute the CumDist
7:     return bnDicho(d, cumD, weights)
8: end procedure
```

For each bucket we then generate a list of black n-grams that will allow to remove a predefined ratio of sentences[3] of the bucket. Algorithm 2 gives the pseudocode of the black n-grams generation process. Its parameters are a bucket, the n-grams range (from 1 to 3 in our case) as well as a deletion ratio indicating the proportion of sentences that black n-grams have to remove. First, line 2, we generate class-documents of the bucket which correspond to a concatenation of documents per class. Thus the variable *cd* is a list of class-documents that are equal in number to the number of classes in the bucket b. In order to find sentences to filter, we keep the sentence level tokenization as well as the word level tokenization. A class-document is thus a list of sentences made up of tokens. Line 5, we generate TFIDF weights of all n-grams of cd. Line 6, we generate the cumulative distribution function of sentences TFIDF weights. The TFIDF weight of a sentence is the maximum weights of its n-grams. The function $f : \mathbb{R} \to \mathbb{N}$ represents a discretized approximation of the cumulative distribution function:

$$x \mapsto |\{s : s \in S, \text{TFIDF}_{max}(s) >= x\}| \qquad (1)$$

with TFIDF_{max} the function returning the maximum TFIDF weight of a sentence and S the set of all sentences of the bucket.

To extract all black ngrams, we need to search the TFIDF weight threshold so that each n-grams with a TFIDF weight higher or equal allow to remove a ratio d of sentences in the bucket. The goal is is to search the threshold y such that:

$$y = \arg\min_x(\text{abs}(f(x) - d.|S|)) \qquad (2)$$

The use of cumulative distribution functions allows to make the computational complexity of the threshold search constant because it only depends on the discretisation of x we choose in advance.

When using multiple n-grams vocabularies, this step will remove more than the ratio d of sentences because sentence removals are independents. Thus we use a dichotomic search line 7 to find a new deletion ratio between 0 and d. After finding this new ratio and corresponding TFIDF weight thresholds, we extract all black n-grams. For the final step, we merge all black n-grams of each bucket. Thus we obtain a dictionary mapping each class to a set of its black n-grams coming from one or more buckets. For each document in the *R-set*, we remove sentences having a black n-gram associated to the class of the document.

[3]Sentences that are removed from the bucket are sen-

Sarah Kaplan@washingtonpost.com

March for Science participants walk along Constitution Avenue to the Capitol on April 22. (Astrid Riecken for The Washington Post). [...] Conceived in the wake of President Trump's inauguration, and galvanized by his efforts to slash environmental protections and cut the federal budget [...] So we want to know: What has changed for you since the March for Science? Have you altered anything about your life or work? Have your colleagues? Do you feel part of a "global movement"? Do you think the March for Science achieved its goals? Please let us know what you think using the form below.

Adrian Dater@denverpost.com

[...] He has been at The Denver Post hockey columnist and sports feature writer. [...] I would caution against everyone getting their hopes up , though. This is Peter Forsberg after all. But he looked MUCH better at practice today than two days ago. I know he worked with doctors on adjustments to the brace on his right foot in the last two days. [...] Comments are moderated and may not appear immediately.

3030784@blogger.com

Hey I am Josh Debner and I am the kid who sits home and does nothing on fri. nights [...] with my family FUN FUN FUN :-/ two weeks in a row now. [...] I had ALL my academy classes... It is block scheduling so having 4 45min classes is much better than 2 1.5 hour ones... I found since i was absent I don't have to make up a lab YAHOO. Hmmm Bus ride home ate lunch. [...] Lauren and Steph went to movies with Mike and Joe.. yea they had fun. Busy day tomarrow. [...]

Figure 1: Filtered sentences of three sample documents.

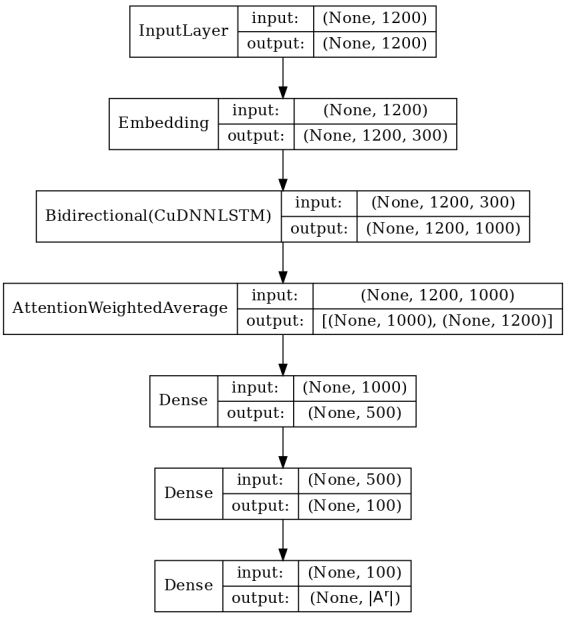

Figure 2: Flow graph of the *SNA* model.

5 EXPERIMENTATION

For this experiment, we used a *R-set* of newspaper and blogs articles[4]. The *R-set* is composed of approximately 3.3 millions of documents and 1200 different classes representing all authors. The minimum number of document per class is 100 and the maximum is 30000. We gathered documents of *The Blog Authorship Corpus* (Schler et al., 2006), ICWSM datasets (Burton et al., 2009, Burton et al., 2011) and news collected for this study. For each article we have the domain name of the source website and we extracted authors from the html content. Online newspapers also showed to have their own consistent writing style (Chakraborty et al., 2016, Dickson and Skole, 2012, Weir, 2009, Cameron, 1996). The style of online newspaper is called journalese with factual analysis, quotes, clickbait trends, etc. Blog articles also have their own style with authors self mentions, personal anecdotes, etc. So in case no author is extracted from the articles or the author has written very few articles, we consider the label to be the online newspapers domain name.

We generated the filtered *R-set* with the method presented in Section 4. The deletion ratio we have chosen for removing sentences is 0.3 and the variation ratio *vr* is 0.05. We chose to remove 30% of the sentences from the reference corpus because we consider it a reasonable trade-off. Thus, corpus filtering can have a significant impact during the training phase, but avoids the elimination of too many sentences that may convey the authors' style. Figure 1 shows filtered sentences of three sample documents in red color. Green sentences are sentences that we kept. Underlined words are words appearing in a black *n*-gram related to the class of the document on top of each text snippet. As we can see, some words are related to the online newspaper such as "*Washington Post*" and "*The Denver Post*". Sentences appearing a lot in documents of an author such as "*Comments are moderated and may not appear immediately*" are automatically removed by the filtering process, thus the process also reduce noise of the corpus for irrelevant sentences. Some *n*-grams are specific to the author such as "*Lauren and Steph*", even common words having specific spelling such as "*tomarrow*".

We use 117 different *U-sets*. We recall that *U-sets* are test sets with unseen documents belonging to unseen authors. These datasets each have 50 authors and 50 documents per author. Datasets *BlogCorpus* and *LiveJournal*, 10 in total, gather documents with labels referring to authors of blog articles. Datasets *WashingtonPost*, *Breitbart*, *BusinessInsider*, *CNN*, *GuardianUK*, *TheGuardian* and *NYTimes* gather documents with author labels, each of these author wrote for the corresponding online newspaper. Datasets *NewsID*, 100 in total, include both documents with author labels and online newspaper labels.

In order to validate the *filtering assumption*, we propose to evaluate two DNNs that share the same

tences that contain a black *n*-gram.

[4]Datasets and code are available at https://github.com/hayj/AuthFilt

Table 1: Impact of the filtering on the authorship clustering (*DavB* and *SimRank* metrics) and attribution tasks (*Acc* metric). The first part of the table (three rows) corresponds to the scores of the *SNA* model trained on the raw *R-set* and the second part (next three rows) corresponds to the scores of the *SNA* model trained on the filtered *R-set*. Each column shows the scores obtained on different *U-sets*.

R-set filtering	Metric	*NewsID (100)*	*BlogCorpus (5)*	*LiveJournal (5)*	*WashingtonPost (1)*	*Breitbart (1)*	*BusinessInsider (1)*	*CNN (1)*	*GuardianUK (1)*	*TheGuardian (1)*	*NYTimes (1)*
Filt.	↓ ***DavB***	3.55	4.29	5.58	7.09	5.6	6.06	6.16	7.79	**4.96**	5.91
⌜	↑ ***SimRank***	0.55	0.39	0.40	0.33	0.37	0.36	0.36	0.30	0.43	0.38
⌜	↑ ***Acc***	0.64	0.50	0.43	0.39	0.40	0.40	0.38	**0.76**	0.36	0.48
Filt.	↓ ***DavB***	**3.35**	**4.13**	**5.23**	**6.88**	**5.07**	**5.86**	**5.93**	**7.15**	5.08	**5.5**
⌜	↑ ***SimRank***	**0.63**	**0.42**	**0.43**	**0.35**	**0.40**	**0.40**	**0.40**	**0.34**	**0.46**	**0.40**
⌜	↑ ***Acc***	**0.69**	**0.57**	**0.47**	**0.42**	**0.48**	**0.45**	**0.42**	0.73	**0.43**	**0.55**

architecture but trained on different versions of the *R-set*. The first one is a DNN trained on the original *R-set* while the second one is trained on the filtered *R-set*. These DNNs models are then evaluated on a variant of the authorship clustering (internal evaluation) and the authorship attribution task. We implemented the *SNA* model (*Stylometric Neural Attention*) which is a bi-directionnal LSTM with an attention layer mainly based on the architecture proposed by (Zhou et al., 2016). Inputs of the DNN are the 300 dimensions *GloVe 840B* (Pennington et al., 2014) word embeddings. We only kept 1200 first words of documents and padded too long documents to 1200 using a specific token. The first layer of the *SNA* model is the bi-directionnal LSTM with 500 units. Since style will not be carried by whole documents, we introduce an attention layer that focus on some words in the document. We added two dense layers with 500 units. The last layer is a softmax layer, each dimension will correspond to an author in A^r. The loss function is the multi-class log loss for the 1200 classes in the *R-set*. We set dropouts of each layers to 0.2. We early stopped the training of both DNNs when no accuracy increase was observed on a validation set. We kept the best models. For both models, the learning time was about one week on a *NVIDIA TITAN V* GPU (12GB memory). Figure 2 gives the flow graph of the *SNA* model. Vector representations of documents are generated using both *SNA* models. For generalization purposes, we do not take the softmax layer as the vector representation of *U-set* documents but the output of the attention layer having less dimensions. The choice of the layer was experimentally validated on a validation set.

We first assess stylometric representation of *U-sets* documents on a variant of the authorship clustering task. Given vector representation of all documents from a *SNA* model and there ground truth labels, we assess how well documents of an author are close to other document of the same author. Thus we assess the quality of representations of documents in their ability to represent the authorship of documents. For this, we use the well-known metric *Davies-Bouldin Index* (abbreviated *DavB*) as well as *SimRank*, a metric introduced in (Hay et al., 2020). *SimRank* is based on nDCG (Järvelin and Kekäläinen, 2002) which assess a ranking quality. For the *SimRank* metric, the rankings of vector representations are computed using the cosine similarity. Next, we assess stylometric representations on the authorship attribution task. We train a linear SVM classifier model on 80% of each *U-set* with vector representations as input data. The score corresponds to the accuracy of predicting the right author label on the 20% remaining data. The model choice and its hyperparameters are grid-searched on a validation *U-set*.

Table 1 shows the results of these experiments. On the left side of the table, the first column indicates whether the *U-set* is filtered or not, thus the first three rows of the table are scores of the *SNA* model trained on the raw *R-set* and the next three rows are scores of the *SNA* model trained on the filtered *R-set*. The second column tells the metrics used: *SimRank* and *DavB* for the authorship clustering and *Acc* for the authorship attribution. For the majority of *U-sets*, the *SNA* model trained on the filtered *R-set* scores higher. Hence both experiments validate the *filtering assumption*.

The filtering process allows an accuracy gain of $\sim$5% on the authorship attribution task by averaging on all test sets categories (columns). For the clustering metrics, the filtering process allows a $\sim$3.5% gain

Table 2: Impact of the filtering on *TFIDF focus* scores. The first row of scores corresponds to *TFIDF focus* scores obtained by the *SNA* model trained on the raw *R-set* and the second by the *SNA* model trained on the filtered *R-set*. Each column shows the *TFIDF focus* scores obtained on different *U-sets*.

***R-set* filtering**	***NewsID (100)***	***BlogCorpus (5)***	***LiveJournal (5)***	***WashingtonPost (1)***	***Breitbart (1)***	***BusinessInsider (1)***	***CNN (1)***	***GuardianUK (1)***	***TheGuardian (1)***	***NYTimes (1)***
¬ Filt.	0.006	0.14	0.12	0.26	0.50	0.57	0.44	0.55	0.37	0.38
Filt.	**0.005**	**0.11**	**0.09**	**0.22**	**0.39**	**0.46**	**0.35**	**0.44**	**0.30**	**0.32**

on the *SimRank* metric. The *DavB* index on the filtered *R-set* get 0.3 points less which corresponds to an improvement of $\sim$5%.

6 UNDISTINGUISHNESS

The *semantic undistinguishness* suggests that style-related linguistic structures tend to carry little information on the content, the topic, the entities, etc. On the other hand, terms with a high semantic value that will identify, for instance, a topic, are those allowing the document to be distinguishable in a corpus. Intuitively, by filtering too much informative words that are related to topics and semantic words consistent for an author, the DNN, during the training phase, will focus less on semantic words but more on function words. This intuition echoes that of (Stamatatos, 2018) with the text distortion method of hiding less frequent words to better identify authors in cross-domain scenarios.

In our case, the DNN will generalize style representations when embedding documents of unknown authors who use a different vocabulary and write on different topics compared to reference authors. Therefore the filtering of the reference corpus can help in the representation of the style by being more in adequacy with the second property of the writing style: the *semantic undistinguishness*.

The TFIDF weighting is a well established method to estimate how important a word is to a document in a corpus. Thus, in order to quantitatively assess the *semantic undistinguishness* of both *SNA* models, we propose a measure based on the TFIDF weighting. The *TFIDF focus* measure allows to compute how well attentions of the model focus on words having lower TFIDF weights:

$$\text{TFIDFFocus}(A,T) = \frac{\text{Tr}(A.T^{\intercal})}{d} \quad (3)$$

A is the attention matrix of size $w \times d$. w is the number of words in a document that we set to 1200 and d is the number of documents. Each line of the matrix corresponds to the attention weights in the *SNA* model for a document in a given *U-set*. An attention vector of a single document is normalized so that the weights sum to 1. The same goes with the normalized TFIDF matrix T of size $w \times d$ computed on the given *U-set*.

Table 2 shows *TFIDF focus* of both *SNA* models on same *U-sets* as the previous experiment. It validates our intuition by showing that the *SNA* model trained on the filtered *R-set* focuses more on terms with low TFIDF weights than the other model.

7 CONCLUSION

The purpose of these experiments is to validate the *filtering assumption* stating that filtering the most informative sentences about authorship allow our representation learning method to better generalize stylometric representations of unseen documents on the basis of reference authors. First we compared two DNNs models , one trained on a reference corpus and another on the same corpus but filtered. The results obtained validated the assumption. The filtering process gained us about 5% on the authorship attribution task and the authorship clustering aiming to assess the quality of documents stylometric representations.

Moreover, we assessed the effect of the filtering of the reference corpus on the adequacy of trained models with the *semantic undistinguishness* which state that style-related latent structures are those which do not make the document distinguishable in the corpus and are more likely to be function words. We showed that the filtering process allows to focus more attention on these terms.

The proposed filtering process offers the scalability properties needed to process the large corpora required to capture style features. In addition it allows to efficiently remove the most informative sentences about the identity of authors according to a predefined deletion ratio. In perspective, we plan to improve the

proposed method by testing different parameters such as the deletion ratio and by using other approaches such as unmasking (Koppel et al., 2007).

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Argamon, S., Dhawle, S., Koppel, M., and Pennebaker, J. W. (2005). Lexical predictors of personality type. In *Proceedings of the Joint Annual Meeting of the Interface and the Classification Society of North America*.

Bagnall, D. (2015). Author identification using multi-headed recurrent neural networks. *CoRR*, abs/1506.04891.

Bischoff, S., Deckers, N., Schliebs, M., Thies, B., Hagen, M., Stamatatos, E., Stein, B., and Potthast, M. (2020). The importance of suppressing domain style in authorship analysis.

Burton, K., Java, A., Soboroff, I., et al. (2009). The icwsm 2009 spinn3r dataset. In *Third Annual Conference on Weblogs and Social Media (ICWSM 2009)*.

Burton, K., Kasch, N., and Soboroff, I. (2011). The icwsm 2011 spinn3r dataset. In *Proceedings of the Annual Conference on Weblogs and Social Media (ICWSM 2011)*.

Cameron, D. (1996). Style policy and style politics: a neglected aspect of the language of the news. *Media, Culture & Society*, 18(2):315–333.

Chakraborty, A., Paranjape, B., Kakarla, S., and Ganguly, N. (2016). Stop clickbait: Detecting and preventing clickbaits in online news media. In *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 9–16.

Chen, Q., He, T., and Zhang, R. (2017). Deep learning based authorship identification.

Dickson, P. and Skole, R. (2012). *Journalese: A Dictionary for Deciphering the News*. Marion Street Press.

Escalante, H. J., Solorio, T., and Montes-y Gómez, M. (2011). Local histograms of character n-grams for authorship attribution. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 288–298, Portland, Oregon, USA. Association for Computational Linguistics.

Goldstein-Stewart, J., Winder, R., and Sabin, R. (2009). Person identification from text and speech genre samples. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 336–344, Athens, Greece. Association for Computational Linguistics.

Granados, A., Cebrian, M., Camacho, D., and d. B. Rodriguez, F. (2011). Reducing the loss of information through annealing text distortion. *IEEE Transactions on Knowledge and Data Engineering*, 23(7):1090–1102.

Gupta, S. T., Sahoo, J. K., and Roul, R. K. (2019). Authorship identification using recurrent neural networks. In *Proceedings of the 2019 3rd International Conference on Information System and Data Mining*, ICISDM 2019, pages 133–137, New York, NY, USA. ACM.

Halvani, O., Graner, L., Regev, R., and Marquardt, P. (2020). An improved topic masking technique for authorship analysis.

Hay, J., Doan, B.-L., Popineau, F., and Ait Elhara, O. (2020). Representation learning of writing style. In *(to appear) Proceedings of the 6th Workshop on Noisy User-generated Text (W-NUT 2020)*.

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

Järvelin, K. and Kekäläinen, J. (2002). Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446.

Karlgren, J. (2004). The wheres and whyfores for studying text genre computationally. In *Workshop on Style and Meaning in Languange, Art, Music and Design. National Conference on Artificial Intelligence*.

Koppel, M., Schler, J., and Bonchek-Dokow, E. (2007). Measuring differentiability: Unmasking pseudonymous authors. *J. Mach. Learn. Res.*, 8:1261–1276.

Lourdusamy, R. and Abraham, S. (2018). A survey on text pre-processing techniques and tools. *International Journal of Computer Sciences and Engineering*, 6(3):148–157.

Menon, R. and Choi, Y. (2011). Domain independent authorship attribution without domain adaptation. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pages 309–315, Hissar, Bulgaria. Association for Computational Linguistics.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Schler, J., Koppel, M., Argamon, S., and Pennebaker, J. (2006). Effects of age and gender on blogging. In *Computational Approaches to Analyzing Weblogs - Papers from the AAAI Spring Symposium, Technical Report*, volume SS-06-03, pages 191–197.

Seroussi, Y., Zukerman, I., and Bohnert, F. (2014). Authorship attribution with topic models. *Computational Linguistics*, 40(2):269–310.

Stamatatos, E. (2007). Author identification using imbalanced and limited training texts. In *18th International*

Workshop on Database and Expert Systems Applications (DEXA 2007), pages 237–241.

Stamatatos, E. (2017). Authorship attribution using text distortion. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1138–1149, Valencia, Spain. Association for Computational Linguistics.

Stamatatos, E. (2018). Masking topic-related information to enhance authorship attribution. *Journal of the Association for Information Science and Technology*, 69(3):461–473.

Weinberger, K., Dasgupta, A., Langford, J., Smola, A., and Attenberg, J. (2009). Feature hashing for large scale multitask learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 1113–1120, New York, NY, USA. Association for Computing Machinery.

Weir, A. (2009). Article drop in english headlinese. *London: University College MA thesis*.

Zhou, P., Shi, W., Tian, J., Qi, Z., Li, B., Hao, H., and Xu, B. (2016). Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212, Berlin, Germany. Association for Computational Linguistics.