Authorship Attribution By Consensus Among Multiple Features

Jagadeesh Patchala
University of Cincinnati
Cincinnati
OH, USA
jagadeesh.patchala@gmail.com

Raj Bhatnagar
University of Cincinnati
Cincinnati
OH, USA
raj.bhatnagar@uc.edu

Abstract

Most existing research on authorship attribution uses various lexical, syntactic and semantic features. In this paper we demonstrate an effective template-based approach for combining various syntactic features of a document for authorship analysis. The parse-tree based features that we propose are independent of the topic of a document and reflect the innate writing styles of authors. We show that the use of templates including sub-trees of parse trees in conjunction with other syntactic features result in improved author attribution rates. Another contribution is the demonstration that Dempster's rule based combination of evidence from syntactic features performs better than other evidence-combination methods. We also demonstrate that our methodology works well for the case where actual author is not included in the candidate author set.

1 Introduction

In the past decade authorship analysis has become a significant need due to its various application areas such as identification of authors of anonymous posts in blogs (Koppel et al., 2011; Koppel and Yeron, 2014), of emails (Patchala et al., 2015), for copyright issues, plagiarism detection, and authentication of electronic documents (Abbasi and Chen, 2005; Chaski, 2005). Typically we encounter two types of author attribution problems. The first kind is the *closed author set problem* where the test-document is written by someone within the set of candidate authors being examined. The second kind is the *semi-closed authorship problem* where the test-document may or may not be written by someone in the set of candidate authors.

The main hypothesis underlying authorship attribution methods is that the grammatical style of an author remains the same across topics and is fairly unique to each author. This uniqueness of an author is easily identifiable when considering small sets of authors, and it expectedly dilutes as the author set becomes large. It is, therefore, desirable to discover from texts more such features that are unique to an author and help increase the discriminability of authors based on their writings. The focus of this paper is on identification and integration of features to enable improved authorship attribution.

We use template based classification and an evidence combination method for exploiting information embedded in different types of syntactic features of documents. We build templates using the frequencies of sub-trees of various heights derived from the parse trees of sentences in documents. Our intuitive basis for using these new features is that they capture the grammatical style very well. We consider the frequencies of sub-trees of heights two, three, and four, as

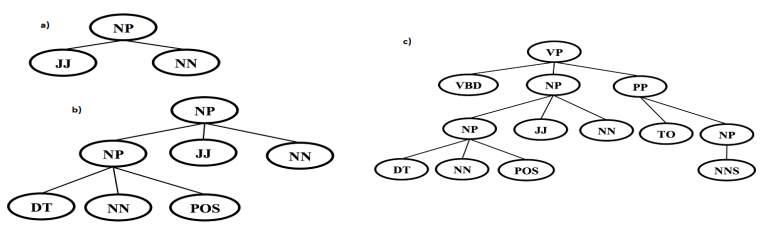

Figure 1: Height two, height three and height four sub-trees of a parse tree.

Proceedings of the 27th International Conference on Computational Linguistics, pages 2766–2777
Santa Fe, New Mexico, USA, August 20-26, 2018.

possible features to describe an author's writings. Examples of sub-trees of various heights derived from a syntactic parse tree are shown in Figure 1. In addition to these we also include three other feature types, namely, character n-grams, function words, and Part-of-Speech (POS) tags' n-grams.

We evaluate the effectiveness of each individual feature type for author attribution and then show that combining the evidence from multiple feature types enhances the performance. We show that the consensus formation approach of Dempster-Shafer (DS) framework (Dempster, 1967; Shafer, 1976) performs better than other evidence combination methods. We have used the consensus formation rule of the Dempster-Shafer theory because the six feature types that we use capture very similar and overlapping underlying information about the syntax of the text. Each feature type also has some unique insights that are not provided by the other features. We cannot combine the six features as independent pieces of evidence, and therefore seek only the consensus among individual inferences based on them. Such consensus among sources of evidence is highlighted by the Dempster's rule of combination.

Using the example in Figure 2, we illustrate the intuitive differences between the DS consensus approach, voting, and aggregation (combination) approaches. Figure 2 shows three features (columns) and three authors 'a', 'b', and 'c' (rows). Given a test document the $mass\ values$ assigned to the three authors by each feature type are shown in their respective columns.

	Height two sub-trees	Height three sub-trees	Height four sub-trees	
NULL	0	0	0	
Author 'a'	0.34	0.3	0.3	Dempster rule
Author 'b'	0.2	0.27	0.5	Combined
Author 'c'	0.36	0.33	0.1	Voting
'a', 'b', 'c'	0.1	0.1	0.1	

Figure 2: An example of Dempster's rule.

The fourth row, including all the authors, captures the uncertainty of the evidence source and this much mass is not assigned to any individual author. When an author is selected based on voting, author 'c' is elected because two out of three feature types declare author 'c' as the winner. When we use a combination approach ($0.2 + 0.27 + 0.5 = 0.97$), author 'b' is selected. When Dempster's rule is used, author 'a' is selected because the extent to which all the different features "agree" on an author is highest for author 'a'.

2 Related work

Any framework for authorship attribution has two main tasks: 1) Feature selection, and 2) A classification technique to do authorship attribution using the selected features.

2.1 Feature Selection

The crux of the authorship attribution problem is the selection of a set of features that remain stable across a large number of writings created by an author. A large variety of features built from lexical, syntactic, semantic, content, and structural properties of the texts have been used as style markers in the past (Juola, 2006; Stamatatos, 2009).

The work published in Kjell et al. (1994) is the first to use the character bigrams and trigrams for authorship analysis of $Federalist$ papers. Later, the work in (Keelj et al., 2003; Houvardas and Stamatatos, 2006; Stamatatos, 2007; Stamatatos, 2012) used character n-grams of different sizes and reported an accuracy of up to 72% with 50 authors.

Syntactic features explain the grammatical structure of sentences and are considered to be reliable style markers as they are not under the conscious control of the author (Bayyen et al., 1996). There are many studies (Argamon and Levitan, 2005; Garcia and Martin, 2007; Kestemont, 2014) that showed the effectiveness of function words as features in authorship attribution tasks. Using rewrite rules as features - Gamon (2004) reported an accuracy of 86% (author set size 3), and Patchala et al. (2015) reported 74% accuracy (30 authors). The work in Eder (2013) studied the effectiveness of POS trigrams with a set of 21 authors and reported an accuracy of 65%. The problem with POS n-grams is that, they don't provide much information about how an author forms the phrases in sentences. The features generated with partial parsing (Luyckx and Dalemens, 2005) capture only the dependency between phrases but not how each phrase is formed. Our intuition is that considering various levels of sub-trees will capture dependency between phrases and also how a phrase is formed. The authors of Kim et al. (2011) identified

- For each author
  1. Combine all the articles in training text set for each author.
  2. Generate the character trigrams from raw text and compute their frequencies.
  3. Tokenize the text into sentences.
  4. Using a parser (Klein and Manning, 2003), parse the sentences and extract the POS trigrams, function words, and height two, three, four level sub-trees, and compute the frequencies for each feature type.
  5. Sort the features within each feature type by decreasing frequencies.
  6. Choose the top n discriminating features using the entropy metric for each feature.

Figure 3: Steps to generate the features

the most discriminating sub-trees in syntactic parse-trees and reported an accuracy of 84% (for 7 authors). Even though the sub-trees we derived from syntactic trees are in the same spirit as in Kim et al. (2011), we use them in building author templates, and combine the information of these features with other syntactic feature types in a completely different and novel way.

2.2 Attribution Techniques

There are two types of classifiers used in authorship attribution: 1) Template-matching classifiers, and 2) Machine-learning based classifiers (such as SVMs).

In template-matching approaches, all the training documents by an author are merged to form a single text collection and is used to generate a feature template for each author's writing style. A test document's signature is then compared with each author's template and authorship is assigned to the author with the closest match. Using character n-grams as features and entropy as similarity measure, the work in Peng et al. (2003) has reported an accuracy of 90% (for 8 authors). Instead of using a single template for each author, the work in (Jankowska et al., 2013; Layton, 2014; Koppel et al., 2011) created multiple templates for each author by selecting subsets of features and the result from each template is combined to find the best match using voting. The work in Koppel et al. (2011) has reported precision and recall values of 90% and 43% for 1000 authors and proved that this template based approach works well even in the case of large author sets.

In machine learning based approaches each document is treated as a representative data point of an author's style in a feature space and a classifier is trained. The work in Abbasi and Chen (2005) uses a combination of lexical, structural and syntactic features, employs Decision Trees as model, and reports an accuracy of 90% (for 20 authors). A number of studies in authorship analysis (Gamon, 2004; Zheng et al., 2006; Stamatatos, 2007; Luyckx and Daelemans, 2008; Luyckx, 2011; Silva et al., 2011) have used various Support Vector Machine (SVM) based classifiers. The work in Luyckx and Daelemans (2008) uses SVM and POS n-grams to assign authorship of essays written by 145 students and reported an accuracy of 55%. Work in (Stamatatos, 2007; Stamatatos, 2012) reported an accuracy of 75% (50 authors) by using character n-grams and SVM.

3 Methodology

We address the authorship attribution problem using the template-based methodology. The main reasons for this choice include stability in the face of changing sets of authors, ability to handle semi-closed authorship problems, and ability to more effectively combine different types of syntactic feature sets.

Each syntactic feature type captures a different mix of properties of the text in a document. This necessitates inclusion of a number of different feature types to design a robust classification system. We consider six types of features: character trigrams, function words, POS trigrams, and sub-trees of height two, three and four.

The steps to generate author's template are summarized in Figure 3. Instead of including thousands of feature values into an author template, we find the most frequent and then within them the most

informative feature values of each feature type. That is, we may include some character trigrams by first selecting the most frequent ones and then within those the most informative ones as measured by their informational entropy across all the authors. Typically 800 to 1500 of each feature's values and their frequencies are included in a template. The most frequent feature items used by an author are likely to persist in their higher relative frequencies across their multiple writings. One such template for each of the six feature types is created for each author. Let us say the resulting six templates for an author are: T_1, T_2, T_3, T_4, T_5, and T_6. Each template contains the relative frequencies of occurrence of individual feature items of a feature type for an author. For example, a template may contain relative frequencies of selected 1000 character trigrams, or 1000 POS trigrams, or 1000 height two sub-trees of parse trees.

Given a test document, six signatures referred to as: S_1, S_2, S_3, S_4, S_5, and S_6, which are similar in structure to the six templates generated for authors are generated. We use the Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951) to compute the dissimilarity values between a test document's six signatures and the corresponding six templates for an author. This gives us six different divergence values for each document-author pair.

We now consider the decision rules for assigning an author to a test document. We examine the cases of making a decision for: (i) the closed author set using only one feature type (ii) the semi-closed author set using only one feature type; (iii) the closed author set using all the six feature types; and (iv) the semi-closed author set using all the six feature types. Our decision making strategies for these four cases are as follows.

Case-1: Compute the divergence value between the test document's selected feature signature, S_m, and each author's template for the m^{th} feature type, T_m. Let us say, D(S_m , T_m(i)) is the divergence value for the m^{th} feature type for the i^{th} author's template. We find the k^{th} author for whom this divergence value is the smallest and assign him/her the authorship.

Case-2: We use the intuitive idea that if the k^{th} author is the true author then the value D(S_m , T_m(k)) must be significantly smaller than the average of D(S_m , T_m(i)) values for all the authors in the pool. We capture this notion by requiring that the k^{th} author is assigned to a document only when the z-score of D(S_m , T_m(k)) is -2.0 or smaller when compared to the values D(S_m , T_m(i)), for all authors else we announce that the true author is not in the candidate set.

Case-3: In this case we seek to combine information obtained from all six feature types and obtain consensus among the inferences arrived by each of them individually. We use Dempster's rule from Theory of Evidence to combine the evidences. Let us say there are k authors in the pool of known authors and for the m^{th} feature type we compute the k divergence values as Div(m,i) = D(S_m , T_m(i)) where i takes the value from 1 to k. Now such a divergence vector is generated for each of the six feature types.

To apply the Dempster's Rule we need to convert these divergence values for authors (for a fixed feature type) to a $mass\ assignment$. As part of this conversion process we have to group all those authors who have very similar divergence values for a feature type into a single group. Given a vector of k divergence values (one feature type), we first normalize the divergence values by scaling them as follows:

$$NormDiv(i,j) = \frac{(max_j(Div(i,j)) - Div(i,j))}{(max_j(Div(i,j)) - (min_j(Div(i,j)))}. \quad (1)$$

Here, i represents the author number, j represents the feature type and $Div(i,j)$ denotes the divergence value of i^{th} author for j^{th} feature type. These scaled values are such that the author with the template closest to the document signature gets a value of 1 and the author with the template farthest from the document gets a value of 0. If two or more authors have very similar scaled scores (say 1.0 and 0.98) then they are equally likely to be the authors of the document (as per the feature type under consideration).

Therefore, as the next step of this process we form groups of authors from the perspective of this feature type. For each author in the candidate set, we identify the authors whose scaled scores are within the threshold δ and include all these authors to form an author set. The score of the author group is set equal to the average of the scaled values of the authors contained in the group. We then remove the duplicate author groups, if any are formed, and normalize all the group scores (for the author sets,

for each feature type) so that all the scores (for each feature type) add up to 1.0 as per the DS theory requirement of a mass assignment. We call these scores as mass values. Now, we have generated one mass assignment function for each feature type.

For example, say, for eight authors (a, b, c, d, e, f, g, h) the scaled values are: (1, 0.97, 0.94, 0.92, 0.87, 0.81, 0.73, 0). The mass assignment for these eight authors is shown in Table 1.

Author groups	Scaled scores	Average	mass values
<a,b >	<1, 0.97 >	0.985	0.157
<a, b, c, d >	<1, 0.97, 0.94, 0.92 >	0.957	0.153
<b, c, d >	<0.97, 0.94, 0.92 >	0.943	0.151
<b, c, d, e >	<0.97, 0.94, 0.92, 0.87 >	0.925	0.148
<d, e >	<0.92, 0.87 >	0.895	0.143
<f >	<0.81 >	0.81	0.129
<g >	<0.73 >	0.73	0.116
<h >	<0 >	0	0

Table 1: Mass assignment process using eight authors and δ value 0.05

We repeat this process for the divergence vector of each feature type and generate mass assignments for groups of authors. Then, the mass assignments for each feature type are combined using Dempster's rule of combination. Given two mass assignments m_1, m_2, the combined mass assignment m_{12} is computed as follows:

$$m_{12}(A) = \frac{1}{1-K} \sum_{B \cap C = A \neq \emptyset} m_1(B) m_2(C), \quad K = \sum_{B \cap C = \emptyset} m_1(B) m_2(C) \tag{2}$$

Here, K is the measure of the amount of conflict between two mass assignments and A, B, C represent subsets of authors.

After computing the consensus mass assignment from six feature types we compute the *plausibility* values of each candidate author being the true author of the test document.

The plausibility pl(A) is the sum of all the masses of sets 'B' that intersect the set of interest-'A' and is defined as follows:

$$pl(A) = \sum_{B | B \cap A \neq \emptyset} m(B) \tag{3}$$

We then select the author with the highest plausibility value as the true author of the test document.

Case-4: For this situation we follow the same process as in case-3 and then determine if the author with the highest plausibility has a plausibility value greater than 0.5, and its z-score among plausibility values for all authors is larger than 2.0. If not, we say that the true author is not in the given author set.

4 Dataset Description

News Articles' Dataset: We selected 7 columnists from the New York Times newspaper and 3 from the Guardian newspaper and extracted 50 articles for each author. To make each author's writings topic independent we selected the topics of sports, education, immigration, elections, health and politics and made sure that each author has written articles on at least 4 of these topics in our collection. These articles consists of, on an average, 800 words per document.

Reuter_50_50 Dataset: *Reuter_50_50* dataset (Houvardas and Stamatatos, 2006; Stamatatos, 2007; Bache and Lichman, 2013) has 50 authors and 5,000 documents. The training corpus has 2,500 documents and test corpus has 2,500 documents (50 documents for each author in train and test) with an average of 500 words per document.

Blogs Dataset: We used the *blogs* dataset (Schler et al., 2006; Koppel et al., 2011) and selected the top 100 authors based on the number of posts. We selected 10 posts for each author as test documents and rest as training documents. Each posting has around 200 words on average.

5 Results and Analysis

We have used the traditional metrics - accuracy, precision, and recall for measuring the performance of our author attribution study.

We have used the Naïve Bayes, Support Vector Machine (SVM), Voting, and the method proposed in Koppel et al. (2011) which we call as 'Koppel 2011' to compare the performance of proposed features types and methodology. Matlab's Naïve Bayes classifier with multinomial distribution and LIBSVM one-vs-one classifier with RBF kernel are used for Naïve Bayes and SVM based classification, and parameters are optimized in each case.

In case of attributing an author based on all six feature types, the method labeled 'Combined (C)' is performed by concatenating all the six feature templates of an author into one large template and then making the decisions using the divergence of similarly concatenated test signatures. The voting method labeled 'Voting' is executed by assigning the authorship to the author selected by maximum number of feature types. The method with label 'Koppel 2011' uses the approach discussed in Koppel et al. (2011). The purpose of these methods is to contrast their result with proposed Dempster's combination rule based method (DS combined).

5.1 Closed author set using only one feature type

The main reason to assign authorship considering each feature type (section 3: case-1) separately is to compare and contrast their individual performances. For each feature type, we kept the number of features - 'n' to be constant at 1,500 and reported the accuracies in Figure 4. This figure shows the accuracy of author attribution using four different classifiers: our proposed method KL-Divergence, Naïve Bayes, SVM, and Koppel 2011.

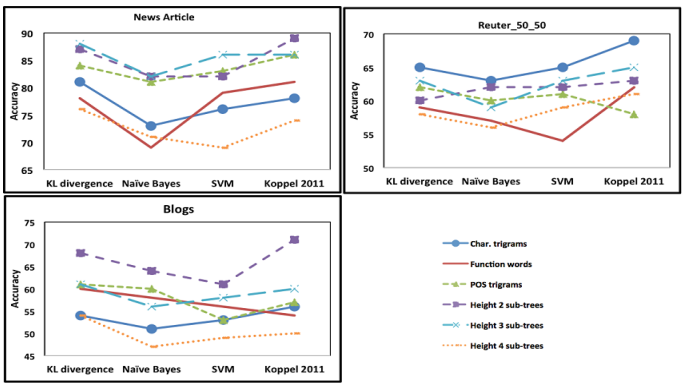

Figure 4: Accuracy values for closed-author set using one feature type.

Here, we can see that the frequencies of sub-trees of height two, and three, and POS trigrams - yielded greater accuracy in *news articles* and *blogs* datasets irrespective of the classification algorithm used. In *Reuter_50_50* dataset, character trigrams gave a slightly better accuracy compared to other features. A common intuition with character n-grams is that they capture the thematic information along with the author style (Stamatatos, 2012). Many authors in *Reuter_50_50* dataset have written articles on a particular subtopic with a high occurrence of a set of words specific to the topic. This makes it easy for character trigrams to distinguish different authors. Parse trees frequencies work better for the News Articles dataset because each author has written articles on different topics and the feature is topic independent.

5.1.1 Effect of feature set size

For this analysis (Figure 5), we kept the training and test document sets fixed, and gradually increased the number of best feature items (number of best character 3-grams, or height 2 sub-trees etc.) included in each template. Most of the feature types attained maximum accuracy when 1000 to 2000 feature items are included and are stable in performance when few hundred feature items are added.

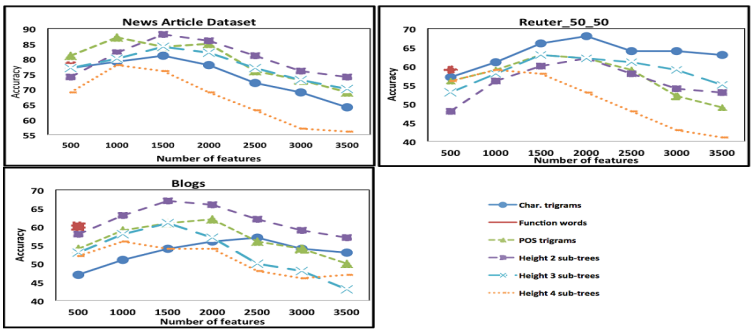

Figure 5: Effect of feature set size.

Afterward, the performance decreased as more feature items are added. The results presented in (Stamatatos, 2007; Stamatatos, 2012) suggest considering at least a feature set size of 3000 - 5000 for character trigrams in *Reuter_50_50* dataset. In their formulation they include the 5000 most frequent character trigrams in the feature set whereas in our formulation we select the top '*n*' discriminating features based on informational entropy. From Figure 5, we can observe that there is no overall consensus on ideal size for feature sets. However, one can achieve a reasonable accuracy by considering the most discriminating features numbering between 1000 and 2000. This is a very wide range. For larger numbers it seems the curse of dimensionality sets in and the extra feature items, that are less discriminating, add noise to the author attribution process. Fewer feature items have lower accuracy because they may not have enough information in them to make best decisions.

5.1.2 Effect of training data size

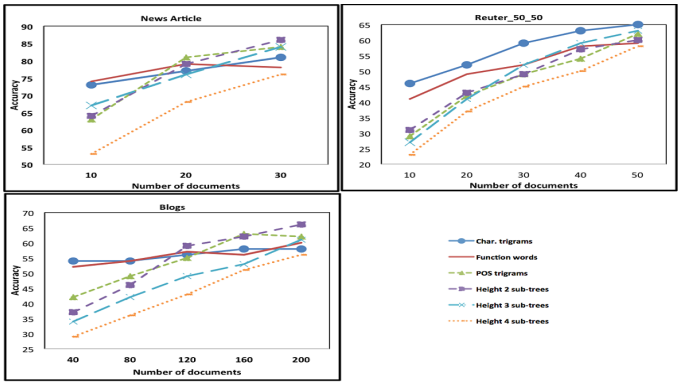

Figure 6: Effect of training data size.

To analyze the effect of limited training data, we started with ten documents and increased the number of documents in multiples of ten for *news article* and *Reuter_50_50* dataset. In *blogs* dataset, we started with 40 documents and increased the size in multiples of 40. Each document in *news article* dataset is approximately twice the size of the document in *Reuter_50_50* and *blogs* dataset. The test set is kept constant for each author and the attribution accuracy results are shown in Figure 6. There is a large difference in accuracy between the character trigrams and other syntactic features especially with limited training data and this difference narrows once we start increasing the number of training documents. The function words and character 3-grams templates have the best performance when the training dataset is very small. So, these features are very effective even with very small training set sizes. We also notice that, expectedly, as the number of training documents increases the performance for each feature type improves consistently. At the higher end of the number of training documents, subtrees of height two became the best feature types for the News Articles and the Blogs datasets and kept improving their performance even for the Reuters dataset.

5.2 Semi-closed author set using only one feature type

Here, we use the approach described in Case-2 of the methodology section (Section 3). In addition to the test documents in the *news articles* dataset, we added 200 randomly selected documents from the *Reuter_50_50* dataset to the *news article* test set. Similarly, we added 200 test documents from *news article* dataset to *Reuter_50_50* dataset. For *blogs* dataset, we randomly selected 500 documents written by the authors who are not in the selected sample dataset and added them to the test set. The feature set size '*n*' is kept constant at 1500 items and z-score threshold for making a decision is selected as -1.5 for all the datasets.

In Table 2, we show the precision and recall values, measured individually for each feature type. We can observe the similar behavior as in the closed author set case. That is, grammar based features gave better performance in *news article* and *blogs* datasets and character trigrams gave higher performance in *Reuter_50_50* dataset. We observe that as the number of authors increases the precision and recall values decline. They are the highest for the News articles dataset (10 authors), followed by Reuters (50 authors) and then the lowest for the Blogs dataset (100 authors). Also, the recall is consistently smaller than the precision because for the semi-closed case an author is announced only if he/she stands significantly apart compared to other authors.

		Character trigrams	Function words	POS trigrams	Height 2 sub-trees	Height 3 sub-trees	Height 4 sub-trees
News article	prec.	69	72	71	77	74	59
	rec.	51	47	56	54	58	43
Reuter_50_50	prec.	64	66	57	54	57	51
	rec.	56	53	49	53	51	49
Blogs	prec.	41	43	46	47	46	44
	rec.	36	41	43	41	41	37

Table 2: Precision and Recall for semi-closed authorship using individual features (Case-2)

		Combined (C)	Voting	Koppel 2011	SVM	DS combined(3)	DS combined(6)
News Article	prec.	77±4.4	69±2.9	80±3.4	76±3.8	78±4.5	88±3.0
	rec.	73±5.1	71±2.7	83±2.9	72±4.4	73±3.2	82±2.0
Reuter_50_50	prec.	56±3.9	64±3.2	64±4.2	63±3.5	67±4.0	73±3.7
	rec.	61±4.8	59±3.6	62±2.6	64±3.1	62±4.6	67±5.0
Blogs	prec.	58±4.1	54±3.8	62±3.9	57±3.6	57±3.4	68±2.4
	rec.	61±3.2	52±3.4	63±3.4	54±3.9	56±2.2	65±1.1

Table 3: Precision and Recall values using all feature types (Case-3)

5.3 Closed author set using all the six feature types

We consider feature item set sizes between 500 - 3000, in multiples of 500, and the average values for precision and recall along with their standard deviations are show in Table 3. Here, the threshold δ for forming the groups of authors for the Dempster's rule combination method, is kept constant at 0.05. We observed that for δ values between 0.025 and 0.075 there is negligible change in performance in all the datasets. To show that the sub-tree features derived from the parse trees capture some additional author discriminating information that is not captured by the char. trigrams, func. words, and POS trigrams, we performed the following experiment. First we used the three feature types (func. words, char. trigrams, POS trigrams) and calculated the precision and recall values using our DS combined approach (*DS combined(3)*). Then we considered all six feature types, and calculated the precision and recall values (*DS combined(6)*). There is a significant increase in precision and recall values when we include the sub-tree features for consensus formation compared to only combining the three non-tree feature types. This validates our hypothesis that the sub-tree feature types capture some additional information that is not captured by the other three feature types. We also can see that the performance improves significantly when inferences from multiple features types are combined using various aggregation methods; and the best performance for all datasets is obtained when Dempster's rule is used.

Robustness Analysis:

The main aim of our test described in this section is to analyze how our *DS combined* method performs when small changes are introduced in the test documents. For each test document in an author's test set we

		Original	Adding 50 words	Adding 100 words
News Article	prec.	88±3.0	86±4.3	83±3.8
	rec.	82±2.1	80±3.9	77±5.7
Reuter_50_50	prec.	73±3.7	71±2.8	65±2.7
	rec.	66±5.0	64±3.7	60±3.6
Blogs	prec.	68±2.4	61±2.1	54±2.4
	rec.	65±1.1	58±2.4	51±2.3

Table 4: Precision and Recall values with slight changes to test documents

added 50 random words of text taken from a different author within the same dataset. We repeated this in all the three datasets and used feature set sizes between 500 - 3000, in multiples of 500. The results obtained are shown in Table 4. The δ value is kept constant at 0.05 for all the datasets. The *Blogs* dataset is very sensitive to changes in test documents as the test document size is very small (200 words). There is very small drop in precision and recall values in other datasets when we added 50 random words to a test document. However, when we added 100 random words to each test document, performance decreased more in *Reuter_50_50* dataset but not that much in *news article* dataset. This is because the average test document size is small (500 words) in the former dataset and the average size in *news article* dataset is 800 words.

To study the effect of candidate author set size on *DS combined* approach, we kept the feature set size constant at 1500 items and analyzed the performance by gradually increasing the author set size. For each author set size, we repeated the experiment ten times by randomly selecting the authors and the average precision and recall values are reported in Figure 7. As expected, both the precision and recall values slowly decrease with an increase in the author set size.

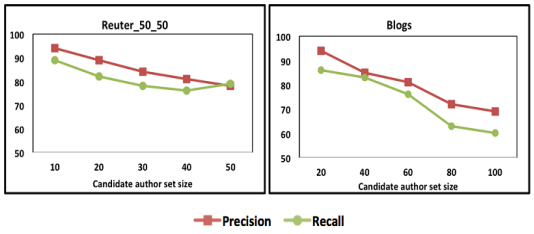

Figure 7: Precision and recall values for different candidate author set sizes.

5.4 Semi-closed author set using all the six feature types

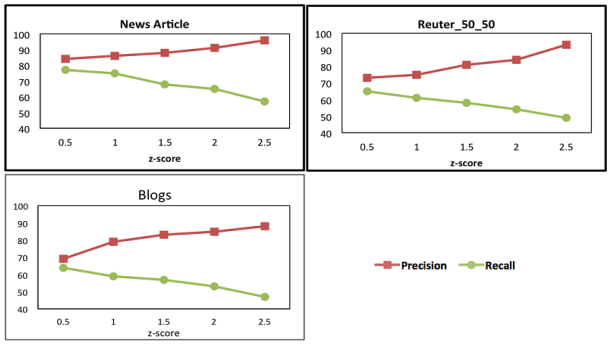

Figure 8: Precision and recall values for different z-score cutoff values.

For this test, we have used the same datasets as described in Section 5.2 and used the approach described in case-4 of the methodology section. The precision and recall values for various z-score cut off values are shown in Figure 8. In all datasets the precision values do not decline, but rise slightly, as the z-score cutoff goes up to 2.5. This is because a larger z-score cutoff means that the winning author must have a significantly higher plausibility value compared to all the other authors. Therefore, an author is announced for a document only if he/she is the clear winner compared to all the others. The recall values reduce somewhat with increasing z-score cutoff and this is because the authors who do not stand out very strongly compared to the other authors do not get attributed to the documents.

6 Conclusion

In the above discussion we have shown two main contributions of our proposed methodology for author attribution of documents. The first is that templates formed from the sub-tree frequencies in the parse tree of an author's text provide very valuable insights about one's writing style. The second main contribution is the demonstration that information embedded in different types of syntactic features is best combined using Dempster's rule compared to some other methods of information fusion. We have also shown successfully that our proposed approach works well for both - 'closed' and 'semi-closed' cases. We have shown the robustness of our proposed approach from various perspectives and the best results obtained using Dempster's Rule based combination of all six features are of significantly high quality.

References

Abbasi, A., and Chen, H. 2005. Applying Authorship Analysis to Extremist-Group Web Forum Messages. *IEEE Intelligent Systems* 20(5): 67-75.

Argamon, S., Koppel, M., and Avneri, G. 1998, Routing documents according to style, *First International workshop on innovative information systems*, pp. 85-92.

Argamon, S., Levitan, S. 2005, Measuring the usefulness of function words for authorship attribution. *In Proceedings of the 2005 ACH/ALLC Conference*.

Argamon, S., Whitelaw, C., Chase, P., Raj, H. S., Garg, N., and Levitan, S. 2007. Stylistic text classification using functional lexical features: Research Articles. *Journal of American Society for Information Science and Technology*, 58(6): 802-822.

Axelsson, M. W., 2000. USE-the Uppsala Student English corpus: an instrument for needs analysis. *ICAME journal* 24:155-157.

Bache, K. and Lichman, M. 2013. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

Baayen, H., Van Halteren, H., and Tweedie, F. (1996). Outside the cave of shadows: Using syntactic annotation to enhance authorship attribution. *Literary and Linguistic Computing*, 11(3): 121-132.

Bayyen, R. H., Halteren, H. V., Neijt, A., and Tweedie, F. J. 2002. An experiment in authorship attribution. *In Proceedings of the 6th International Conference on Statistical Analysis of Textual Data*, pp. 29-37.

Chaski, C.E. 2005, Who's At The Keyboard? Authorship Attribution in Digital Evidence Investigations. *International Journal of Digital Evidence*, 4(1): 1-13.

Dempster, A. P. 1967. Upper and lower probabilities induced by a multivalued mapping. *The Annals of Mathematical Statistics*, 38(2): 325-339.

Eder, M. 2013 Does size matter? Authorship attribution, small samples, big problem. *Literary and Linguistic Computing*: fqt066.

Galitsky, B. 2013. Machine learning of syntactic parse trees for search and classification of text. *Engineering Applications of Artificial Intelligence*, 26(3): 1072-1091.

Gamon, M. 2004. Linguistic correlates of style: authorship classification with deep linguistic analysis features. *In Proceedings of the 20th international conference on Computational Linguistics (COLING '04). Association for Computational Linguistics*, Stroudsburg, PA, USA, Article 611.

Garca, A. M., and Martin, J. C. (2007). Function words in authorship attribution studies. *Literary and Linguistic Computing*, 22(1), 49-66.

Houvardas, J., and Stamatatos, E. 2006. N-gram feature selection for authorship identification. *In Artificial Intelligence: Methodology, Systems, and Applications*. Springer Berlin Heidelberg, pp. 77-86.

Jankowska, M., Keelj, V., and Milios, E. 2013. Ensembles of Proximity-Based One Class Classifiers for Author Verification ? Notebook for PAN at CLEF 2014. *In Cappellato, L., Ferro, N., Halvey, M., and Kraaij, W. (eds.). CLEF 2014 Labs and Workshops, Notebook Papers. CEUR Workshop Proceedings (CEUR-WS.org)*,ISSN 1613-0073.

Juola, P. 2006. Authorship attribution. *Found. Trends Inf. Retr*, 1(3): 233-334.

Keelj, V., Peng, F., Cercone, N., and Thomas, C. 2003. N-gram-based author profiles for authorship attribution. *In Proceedings of the conference pacific association for computational linguistics*, PACLING, Vol. 3, pp. 255-264.

Kestemont, M. 2014. Function Words in Authorship Attribution From Black Magic to Theory?. *EACL 2014*, pp. 59-66.

Kim, S., Kim, H., Weninger, T., Han, J., and Kim, H. D. 2011. Authorship classification: a discriminative syntactic tree mining approach. *In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. pp. 455-464.

Kjell, B., Woods, W. A., and Frieder, O. 1994. Discrimination of authorship using visualization. *Information processing and management*, 30(1), 141-150.

Klein, D., and Manning, D. C. 2003. Accurate Unlexicalized Parsing. *Proceedings of the 41st Meeting of the Association for Computational Linguistics*, pp. 423-430.

Koppel, M., and Schler, J. 2003. Exploiting stylistic idiosyncrasies for authorship attribution. *In Proceedings of IJCAI'03 Workshop on Computational Approaches to Style Analysis and Synthesis*, pp. 69-72.

Koppel, M., Schler, J., and Zigdon, K. 2005. Automatically determining an anonymous author's native language. *In Proceedings of the 2005 IEEE international conference on Intelligence and Security Informatics* (ISI'05), Paul Kantor, Gheorghe Muresan, Fred Roberts, Daniel D. Zeng, and Fei-Yue Wang (Eds.). Springer-Verlag, Berlin, Heidelberg, pp. 209-217.

Koppel, M., Akiva, N., and Dagan, I. 2006. Feature instability as a criterion for selecting potential style markers: Special Topic Section on Computational Analysis of Style. *Journal of American Society for Information Science and Technology*, 57(11):1519-1525.

Koppel, M., Schler, J., Argamon, S. 2011. Authorship attribution in the wild. *Language Resources and Evaluation*, 45(1): 83-94.

Koppel, M., and Yaron, W. 2014. Determining if two documents are written by the same author. *Journal of the Association for Information Science and Technology*, 65(1): 178-187.

KULLBACK, S. and LEIBLER, R.A. 1951. On Information and Sufficiency, *Ann. Math. Statist.*, 22: 79-86.

Layton, R. 2014. A Simple Local n-gram Ensemble for Authorship Verification ? Notebook for PAN at CLEF 2014. *In Cappellato, L., Ferro, N., Halvey, M., and Kraaij, W. (eds.). CLEF 2014 Labs and Workshops, Notebook Papers. CEUR Workshop Proceedings (CEUR-WS.org)*,ISSN 1613-0073.

Luyckx, K., and Daelemans, W. 2005. Shallow text analysis and machine learning for authorship attribution. *In Proceedings of the 15th meeting of Computational Linguistics in the Netherlands*, pp. 149-160. Utrecht, Netherlands: LOT.

Luyckx, K., and Daelemans, W. 2008. Authorship attribution and verification with many authors and limited data. *In Proceedings of the 22nd International Conference on Computational Linguistics* -Volume 1, pp. 513-520. Association for Computational Linguistics.

Luyckx, K. 2011. Scalability issues in authorship attribution. *ASP/VUBPRESS/UPA*.

McCarthy, P.M., Lewis, G.A., Dufty, D.F., and McNamara, D.S. 2006. Analyzing writing styles with coh-metrix. *In Proceedings of the Florida Artificial Intelligence Research Society International Conference*, pp. 764-769.

Mosteller, F., and Wallace, D. L. 1963. Inference in an authorship problem: A comparative study of discrimination methods applied to the authorship of the disputed Federalist Papers. *Journal of the American Statistical Association*, 58(302), 275-309.

Nektaria, P., and Stamatatos, E. 2014. A Profile-Based Method for Authorship Verification. *Artificial Intelligence: Methods and Applications*. Springer International Publishing, pp. 313-326.

Patchala, J., Bhatnagar, R., and Gopalakrishnan, S. 2015. Author Attribution of Email Messages Using Parse-Tree Features. *In Machine Learning and Data Mining in Pattern Recognition*. Springer International Publishing, pp. 313-327.

Peng, F., Schuurmans, D., Wang, S., and Keselj, V. 2003. Language independent authorship attribution using character level language models. *In Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics - Volume 1* (EACL '03), Vol. 1. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 267-274.

Raghavan, S., Kovashka, A., and Mooney, R. 2010. Authorship attribution using probabilistic context-free grammars. *In Proceedings of the ACL 2010 Conference Short Papers (ACLShort '10)*. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 38-42.

Schler, J., Koppel, M., Argamon, S., and Pennebaker, J. 2006. Effects of age and gender on blogging. *In AAAI Spring Symposium*, 06(03): 191-197.

Shafer, G. 1976. *A Mathematical Theory of Evidence*, Princeton: Princeton University Press.

Shane B., Matt, P., and David, Y. 2012. Stylometric analysis of scientific articles, *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics*: Human Language Technologies, June 03-08, Montreal, Canada.

Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., and Chanona-Hernndez, L. 2014. Syntactic N-grams as machine learning features for natural language processing. *Expert Systems with Applications*, 41(3): 853-860.

Silva, R. S., Laboreiro, G., Sarmento, L., Grant, T., Oliveira, E., and Maia, B. 2011. twazn me!!! ;(automatic authorship analysis of micro-blogging messages. *In Proceedings of the 16th international conference on Natural language processing and information systems*, Springer-Verlag, Berlin, Heidelberg, pp. 161-168.

Stamatatos, E. 2007. Author Identification Using Imbalanced and Limited Training Texts, *In Proc. of the 4th International Workshop on Text-based Information Retrieval*, September 3-7; Regensburg, Germany.

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. 2012. On the robustness of authorship attribution based on character n-gram features. *Journal of Law and Policy*, 21: 421.

Stamatatos, E., Daelemans, W., B., Verhoeven, B., Stein, B., Potthast, M., Juola, P., Sanchez-Perez, M. A., and Barrn?Cedeo, A. 2014. Overview of the Author Identification Task at PAN 2014.*Working Notes for CLEF 2014 Conference*, pp. 877-897.

Tan, R. H. R., and Tsai, F. S. 2013. Authorship Identification for Online Text, *In 2013 International Conference on Cyberworlds*, pp.155-162.

Zečević, A. 2011. N-gram Based Text Classification According To Authorship. *In Student Research Workshop*, pp. 145-149.

Zhao, Y., and Zobel, J. 2005. Effective and scalable authorship attribution using function words. *In Information Retrieval Technology*, pp. 174-189, Springer Berlin Heidelberg.

Zhao, Y. and Zobel, J. 2007. Searching with style: authorship attribution in classic literature. *In Proceedings of the thirtieth Australasian conference on Computer science* (ACSC '07), Gillian Dobbie (Ed.). Australian Computer Society, Inc., Darlinghurst, Australia, 62: 59-68.

Zheng, R., Li, J., Chen, H., and Huang, Z. 2006. A framework for authorship identification of online messages: Writing-style features and classification techniques. *Journal of American Society for Information Science and Technology*, 57(3): 378-393.