Clustering by Authorship Within and Across Documents

Efstathios Stamatatos,[1] Michael Tschuggnall,[2] Ben Verhoeven,[3] Walter Daelemans,[3]
Günther Specht,[2] Benno Stein,[4] and Martin Potthast[4]

[1]University of the Aegean, Greece
[2]University of Innsbruck, Austria
[3]University of Antwerp, Belgium
[4]Bauhaus-Universität Weimar, Germany

pan@webis.de http://pan.webis.de

Abstract The vast majority of previous studies in authorship attribution assume the existence of documents (or parts of documents) labeled by authorship to be used as training instances in either closed-set or open-set attribution. However, in several applications it is not easy or even possible to find such labeled data and it is necessary to build unsupervised attribution models that are able to estimate similarities/differences in personal style of authors. The shared tasks on author clustering and author diarization at PAN 2016 focus on such unsupervised authorship attribution problems. The former deals with single-author documents and aims at grouping documents by authorship and establishing authorship links between documents. The latter considers multi-author documents and attempts to segment a document into authorial components, a task strongly associated with intrinsic plagiarism detection. This paper presents an overview of the two tasks including evaluation datasets, measures, results, as well as a survey of a total of 10 submissions (8 for author clustering and 2 for author diarization).

1 Introduction

Authorship attribution is the attempt to reveal the authors behind texts based on a quantitative analysis of their personal style. The most common scenario adopted in the vast majority of previous studies in this field assume the existence of either a closed or an open set of candidate authors and samples of their writing [19, 49]. These samples are then used as labeled data usually in a supervised classification task where a disputed text is assigned to one of the candidate authors. However, there are multiple cases where authorship information of documents either does not exist or is not reliable. In such a case *unsupervised authorship attribution* should be applied where no labeled samples are available [28].

Assuming all documents in a given document collection are single-authored, an obvious task is to group them by their author [18, 28, 46]. We call this task *author clustering* and it is useful in multiple applications where authorship information is either missing or not reliable [1]. For example, consider a collection of novels published anonymously or under an alias, a collection of proclamations by different terrorist groups, or a collection of product reviews by users whose aliases may correspond to the same person. By performing effective author clustering and combining this with

other available meta-data of the collection (such as date, aliases etc.) we can extract interesting conclusions, such as that a collection of novels published anonymously are by a single person, that some proclamations belonging to different terrorist groups are by the same authors, that different aliases of users that publish product reviews actually correspond to the same person, etc. Author clustering is strongly related to authorship verification [25, 52, 51]. Any clustering problem can be decomposed into a series of verification problems where the task is to determine whether any possible pair of documents is by the same author or not. However, some of these verification problems are strongly correlated and this information can be used to enhance the verification accuracy. For example, in a document collection of three documents d_1, d_2, and d_3, we can decompose the clustering task into three verification problems: d_1 vs. d_2, d_1 vs. d_3, and d_2 vs. d_3. However, if we manage to estimate that d_1 and d_2 are by the same author, this information can be used to enhance the verification model for both d_2 vs. d_3 and d_1 vs. d_3.

On the other hand, if the assumption of single-author documents does not hold, then unsupervised authorship attribution should attempt to decompose a given document into its authorial components [23], for example, the identification of individual contributions in collaboratively written student theses, scientific papers, or Wikipedia articles. To identify the individual authors in such and similar multi-author documents, an analysis that quantifies similarities and differences in personal style within a document should be performed to build *authorship clusters* (each cluster comprising the text fragments a specific author wrote). A closely related topic is the problem of plagiarism detection. In order to reveal plagiarized text fragments, algorithms have to designed that can deal with huge datasets to search for possible sources. This is done by so-called *external* plagiarism detectors [38] by pre-collecting data, storing it in (local) databases and/or even by performing Internet searches on the fly [33]. In addition, *intrinsic* plagiarism detection algorithms [53, 57] sidestep the problem of huge datasets and costly comparisons by inspecting solely the document in question. Here, plagiarized sections have to be identified by analyzing the writing style so as to identify specific text fragment that exhibit significantly different characteristics compared to their surroundings, which may indicate cases of text reuse or plagiarism. Although the performance of intrinsic approaches in terms of detection performance is still inferior to that of external approaches [36, 37], intrinsic methods are still important to plagiarism detection overall, e.g., to limit or pre-order the search space, or to investigate older documents where potential sources are not digitally available.

This paper reports on the PAN 2016 shared tasks on unsupervised authorship attribution, focusing on both clustering across documents (*author clustering*) and clustering within documents (*author diarization*[1]). The next section presents related work in these areas. Sections 3 and 4 describe and analyze the tasks, evaluation datasets, evaluation measures, results, and present a survey of submissions for author clustering and author diarization, respectively. Finally, Section 5 discusses the main conclusions and some directions for future research.

[1] The term "diarization" originates from the research field *speaker diarization*, where approaches try to automatically identify, cluster, and extract different (parallel) speakers of an audio speech signal like a telephone conversation or a political debate [32].

2 Related Work

This section briefly reviews the related work on author clustering and author diarization.

2.1 Author clustering

Related work to author clustering, as defined in this paper, is limited. In a pioneering study, Holmes and Forsyth [17] performed cluster analysis on the Federalist Papers. At that time, other early authorship attribution studies only depicted texts in 2-dimension plots based on a principal components analysis to provide visual inspection of clusters rather than actually performing automated clustering [5].

In an attempt to indicate similarities and differences between authors, Luyckx *et al.* [30] applied centroid clustering to a collection of literary texts. However, since only one sample of text per author was considered, their clusters comprised texts by different authors with similar style rather than different texts by the same author. Almishari and Tsudik [1] explored the linkability of anonymous reviews found on a popular review site. However, they treat this problem as a classification task rather than a clustering task.

Iqbal *et al.* [18] presented an author clustering method applied to a collection of email messages in order to extract a unique writing style from each cluster and identify anonymous messages written by the same author. Layton *et al.* [28] propose a clustering ensemble for author clustering that was able to estimate the number of authors in a document collection using the iterative positive Silhouette method. In another study, they demonstrate the positive Silhouette coefficient for author clustering analysis validation [29]. Samdani *et al.* [46] applied an online clustering method to both author-based clustering and topic-based clustering of postings in an online discussion forum and found that the order of the items was not significant for author-based clustering in contrast to the topic-based clustering.

2.2 Author diarization

The term *diarization*, as it is used throughout this paper, covers both intrinsic plagiarism detection and within-document clustering problems. Although they are closely related, many approaches exist covering the former and only a few tackling the latter. Most of the existing intrinsic plagiarism detection approaches adhere to the following scheme: (1) splitting the text into chunks, (2) calculating metrics for all chunks and, if needed, for the whole document, (3) detecting outliers and (4) applying post-processing steps. This way chunks are usually created by collecting a predefined number of characters, words, or sentences, which sum up to all possible chunks by using sliding windows with different lengths. Then, each text fragment is stylistically analyzed by quantifying different characteristics (i.e., features). Typical computations to build stylistic fingerprints include *lexical features* like character n-grams (e.g., [24, 50]), word frequencies (e.g., [16]), and average word/sentence lengths (e.g., [60]); *syntactic features* like part-of-speech (POS) tag frequencies/structures (e.g., [54]); and *structural features* like average paragraph lengths or indentation usages (e.g., [60]). Moreover, traditional IR measures like $tf \cdot idf$ are often applied (e.g., [34]).

To subsequently reveal plagiarism, outliers have to be detected. This is done either by comparing features of each chunk with those of the whole document using different distance metrics (e.g., [34, 50, 56]), by building chunk clusters and assuming each cluster to correspond to a different author (e.g., [21]), and by using statistical methods like the Gaussian normal distribution (e.g., [55]). In most of the scenarios thresholds are needed, which separate non-suspicious chunks from suspicious chunks. Finally, post-processing steps include grouping, filtering, and unifying suspicious chunks.

By comparison, there is hardly any related work on within-document clustering, while many approaches exist to separate a document into distinguishable parts. The aim of the latter is to split paragraphs by topics, which is generally referred to as *text segmentation* or *topic segmentation* [44]. In this domain, the algorithms often perform vocabulary analysis in various forms like word stem repetitions [35] or building word frequency models [45], where *"methods for finding the topic boundaries include sliding window, lexical chains, dynamic programming, agglomerative clustering, and divisive clustering"* [7].

Probably one of the first approaches that uses stylometry to automatically detect boundaries of authors of collaboratively written text has been proposed by Glover and Hirst [11]. Their main intention is not to expose authors or to gain insight into the work distribution, but to provide a methodology for collaborative authors to equalize their style in order to achieve better readability. Graham *et al.* [14] also tried to divide a collaborative text into different single-author paragraphs. Several stylometric features were employed, processed by neural networks and cosine distances, revealing that letter-bigrams lead to the best results. A mathematical approach that splits a multi-author document into single-author paragraphs is presented by Giannella [10], the first step of which is to divide a document into subsequences of consecutive sentences that are written by the same author. Roughly speaking, this is done with a stochastic generative model on the occurrences of words, where the maximum (log-joint) likelihood is computed by applying Dijkstra's algorithm on finding paths.

3 Author Clustering

In this section we report on the results of the author clustering task. In particular, we describe the experimental setup including the task definition, evaluation datasets, evaluation measures, a survey of the submitted approaches and baselines, and finally the evaluation results are analytically presented.

3.1 Task Definition

The requirements of an author clustering tool differ according to the application. In many cases, complete information about all available clusters may be necessary. However, especially when very large document collections are given, it may be sufficient to extract the most likely *authorship links* (pairs of documents linked by authorship). The latter case may be viewed as a retrieval task where we need to provide a list of document pairs and rank the list based on their probability to be true authorship links. In particular, we aim to study the following two application scenarios:

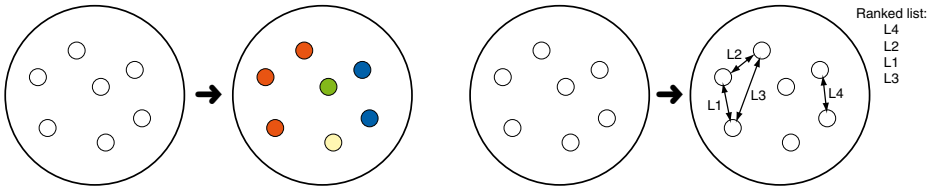

Figure 1. Examples of complete clustering (left) and authorship-link ranking (right).

- **Complete author clustering**. This scenario requires a detailed analysis where the number of different authors (k) found in the collection should be identified and each document should be assigned to exactly one of the k clusters (each cluster corresponds to a different author). In the illustrating example of Figure 1 (left), four different authors are found and the color of each document indicates its author.
- **Authorship-link ranking**. This scenario views the exploration of the given document collection as a retrieval task. It aims at establishing authorship links between documents and provides a list of document pairs ranked according to a confidence score (the score shows how likely a document pair is from the same author). In the example of Figure 1 (right), four document pairs with similar authorship are found and then these authorship-links are ranked according to their similarity.

In more detail, given a collection of (up to 100) documents, the task is to (1) identify groups of documents by the same author, and (2) provide a ranked list of authorship links (pairs of document by the same author). All documents within the collection are single-authored, in the same language, and belong to the same genre. However, the topic or text-length of documents may vary. The number of distinct authors whose documents are included in the collection is not given.

Let N be the number of documents in a given collection and k the number of distinct authors in this collection. Then, k corresponds to the number of clusters in that collection and the ratio $r = k/N$ indicates the percentage of single-document clusters as well as the number of available authorship links. If r is high then most documents in the collection belong to single-document clusters and the number of authorship links is low. If r is low then most of the documents in the collection belong to multi-document clusters and the number of authorship-links is high. In our evaluation, we examine the following selection of cases:

- $r \approx 0.9$: only a few documents belong to multi-document clusters and it is unlikely to find authorship links.
- $r \approx 0.7$: the majority of documents belong to single-document clusters and it is likely to find authorship links.
- $r \approx 0.5$: less than half of the documents belong to single-document clusters and there are plenty of authorship links.

Table 1. Evaluation dataset for the author clustering task (left=training, right=test).

ID	Language	Genre	*r*	*N*	*k*	Links	maxC	Words
001	English	articles	0.70	50	35	26	5	752.3
002	English	articles	0.50	50	25	75	9	756.2
003	English	articles	0.86	50	43	8	3	744.7
004	English	reviews	0.69	80	55	36	4	977.8
005	English	reviews	0.88	80	70	12	3	1,089.7
006	English	reviews	0.50	80	40	65	5	1,029.4
007	Dutch	articles	0.89	57	51	7	3	1,074.7
008	Dutch	articles	0.49	57	28	76	7	1,321.9
009	Dutch	articles	0.70	57	40	30	4	1,014.8
010	Dutch	reviews	0.54	100	54	77	4	128.2
011	Dutch	reviews	0.67	100	67	46	4	134.9
012	Dutch	reviews	0.91	100	91	10	3	125.3
013	Greek	articles	0.51	55	28	38	4	748.9
014	Greek	articles	0.69	55	38	25	5	741.6
015	Greek	articles	0.87	55	48	8	3	726.8
016	Greek	reviews	0.91	55	50	6	3	523.4
017	Greek	reviews	0.51	55	28	55	8	633.9
018	Greek	reviews	0.73	55	40	19	3	562.9

ID	Language	Genre	*r*	*N*	*k*	Links	maxC	Words
001	English	articles	0.71	70	50	33	5	582.4
002	English	articles	0.50	70	35	113	8	587.3
003	English	articles	0.91	70	64	7	3	579.8
004	English	reviews	0.73	80	58	30	4	1,011.2
005	English	reviews	0.90	80	72	10	3	1,030.4
006	English	reviews	0.53	80	42	68	5	1,003.7
007	Dutch	articles	0.74	57	42	24	4	1,172.1
008	Dutch	articles	0.88	57	50	8	3	1,178.4
009	Dutch	articles	0.53	57	30	65	7	945.2
010	Dutch	reviews	0.88	100	88	16	4	151.7
011	Dutch	reviews	0.51	100	51	76	4	150.3
012	Dutch	reviews	0.71	100	71	37	4	155.9
013	Greek	articles	0.71	70	50	24	4	720.5
014	Greek	articles	0.50	70	35	52	4	750.3
015	Greek	articles	0.89	70	62	9	3	737.6
016	Greek	reviews	0.73	70	51	24	4	434.8
017	Greek	reviews	0.91	70	64	7	3	428.0
018	Greek	reviews	0.53	70	37	44	4	536.9

3.2 Evaluation Datasets

A new dataset was developed for this shared task comprising several clustering problems in three languages (Dutch, English, and Greek) and two genres (articles and reviews). Each part of the dataset has been constructed as follows:

- Dutch articles: this is a collection of opinion articles from the Flemish daily newspaper *De Standaard* and weekly news magazine *Knack*. The training dataset was based on a pool of 216 articles while the test dataset was based on a separate set of 214 articles.
- Dutch reviews: this is a collection of reviews taken from the CLiPS Stylometry Investigation (CSI) corpus [59]. These are both positive and negative reviews about both real and fictional products from the following categories: smartphones, fast-food restaurants, books, artists, and movies. They are written by language students from the University of Antwerp.
- English articles: this is a collection of opinion articles published in *The Guardian* UK daily newspaper.[2] Each article is tagged with several thematic labels. The training dataset was based on articles about politics and UK while the evaluation dataset was based on articles about society.
- English reviews: this is a collection of book reviews published in *The Guardian* UK daily newspaper. All downloaded book reviews were assigned to the thematic area of culture.
- Greek articles: this is a collection of opinion articles published in the online forum *Protagon*.[3] Two separate pools of documents were downloaded, one about politics and another about economy. The former was used to build the training dataset and the latter was used for the evaluation dataset.
- Greek reviews: this is collection of restaurant reviews downloaded from the website Ask4Food.[4]

[2] http://www.theguardian.com

[3] http://www.protagon.gr

[4] https://www.ask4food.gr

For each of the above collections, we constructed three instances of clustering problems corresponding to $r \approx 0.9$, $r \approx 0.7$, and $r \approx 0.5$ for both training and test datasets. Detailed dataset statistics on each clustering problem in the training and evaluation datasets are provided in Table 1. As can be seen, there are significant differences in the produced instances. In cases where $r \approx 0.9$, the number of authorship links is small (less than 20), and when $r \approx 0.5$, the authorship links are more than 35. In some cases a low r corresponds to a relatively high maximum cluster size (maxC). English book reviews and Dutch newspaper articles are relatively long texts (about 1,000 words on average) while Dutch reviews are short texts (about 150 words on average). In all other cases average text lengths vary between 400 to 800 words. The largest collections correspond to Dutch articles (100 documents per problem instance) and in all other cases the size of collections ranges between 50 and 80 documents.

3.3 Survey of Submissions

We received 8 submissions from research teams coming from Bulgaria [61], India [26], Iran [31], New Zealand [6], Switzerland (2) [12, 22], and the UK (2) [47, 58]. Two of them have not submitted a notebook paper to describe their approach [12, 26]. The 6 remaining submissions present models that fall into two major categories. *Top-down* approaches first attempt to form clusters using a typical clustering algorithm (k-means) and then transform clusters into authorship links, assigning a score to each link [31, 47]. A crucial decision in such methods is the appropriate estimation of k, the number of clusters (authors) in a given collection. Sari and Stevenson use a k value that optimizes the Silhouette coefficient [47].

Bottom-up approaches, on the other hand, first estimate the pairwise distance of documents, estimating the scores for authorship links, and then use this information to form clusters [6, 22, 58, 61]. The distance measure that attempts to capture the stylistic similarities between documents in some cases is a modification of an authorship verification approach [6, 58]. The effectiveness of the submission of Bagnall [6], especially in the authorship-link ranking task, indicates that exploiting author verification methods is a promising direction. Another idea used by Zmiycharov *et al.* [61] is to transform the estimation of authorship link scores to a supervised learning task by exploiting the training dataset. Given that the amount of true authorship links in the training dataset is very limited in comparison to the amount of false links, this learning task suffers from the class imbalance problem. Bottom-up approaches do not explicitly estimate the number of authors in the collection (k) but they form clusters according to certain criteria. Kocher [22] group texts in one cluster when they are connected by a path of authorship links with significantly high score. A very modest strategy is used by Bagnall [6] which practically forbids clusters with more than two items to be formed. An interesting discussion on more sophisticated methods is also provided by Bagnall [6].

With respect to the stylometric features used by participants, there is no significant novelty. Some approaches are exclusively based on character-level information [6, 47], some on very frequent terms [22, 58]. Others attempt to combine traditional features like sentence length and type-token ratio with word frequencies and syntactic features like part-of-speech tag frequencies and distributions [31, 61]. Evaluation results indicate

that approaches that use homogeneous features are more effective. Sari and Stevenson [47] report that they also examined word embeddings but finally dropped these features since preliminary results on the training dataset were not encouraging.

3.4 Baselines

To be able to estimate the contribution of each submission we provide several baseline methods. First, a baseline based on random guessing (BASELINE-Random) is employed where the number of authors in a collection is randomly guessed and each document is randomly assigned to one author. In addition, an authorship link is established for any two documents belonging to the same randomly-formed cluster and a random score is assigned to that link. We provide average scores corresponding to 50 repetitions of this baseline for each clustering problem. This baseline may be seen as the lower limit of performance in the author clustering task. Moreover, for the complete author clustering task, we provide a simple baseline that considers all documents in a given collection as belonging to different authors. Thus, it forms singleton clusters (BASELINE-Singleton). Such a baseline is very hard to beat when only a few multi-item clusters exist in a large collection of documents. Actually, it guarantees a BCubed precision of 1 while BCubed recall depends on the size of clusters. When the ratio r is high, the BCubed F-score of BASELINE-Singleton will also be high. For the authorship-link ranking task, we provide another baseline method based on cosine similarity between documents. In more detail, each document is represented using the normalized frequencies of all words appearing at least 3 times in the collection (each clustering problem) and the cosine similarity between any two documents is used to estimate the score of each possible authorship link. This baseline method (BASELINE-Cosine) would be affected by topical similarities between documents.

3.5 Performance Measures

There are multiple evaluation measures available for clustering tasks. In general, a clustering evaluation measure can be *intrinsic* (when the true labels of data are not available) or *extrinsic* (when true labels of data are available). Given that the information about the true authors of document is available, our task fits the latter case. Among a variety of extrinsic clustering evaluation metrics, we opted to use *BCubed Precision*, *Recall*, and the *F-score*. The latter has been found to satisfy several formal constraints including cluster homogeneity, cluster completeness, and the *rag bag* criterion (where multiple unrelated items are merged into a single cluster) [2]. Let d_i be a document in a collection ($i = 1, ..., N$). Let $C(d_i)$ be the cluster d_i is put into by a clustering model and $A(d_i)$ be the true author of d_i. Then, given two documents of the collection d_i and d_j, a correctness function can be defined as follows:

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

The BCubed precision of a document d_i is the proportion of documents in the cluster of d_i (including itself) by the same author of d_i. Moreover, BCubed recall of d_i is the

proportion of documents by the author of d_i that are found in the cluster of d_i (including itself). Let C_i be the set of documents in the cluster of d_i and A_i be the set of documents in the collection by the author of d_i. BCubed precision and recall of d_i are then defined as follows:

$$\text{precision}(d_i) = \frac{\sum_{d_j \in C_i} \text{correct}(d_i, d_j)}{|C_i|}, \qquad \text{recall}(d_i) = \frac{\sum_{d_j \in C_i} \text{correct}(d_i, d_j)}{|A_i|}.$$

Finally, the overall BCubed precision and recall for one collection is the average of precision and recall of documents in the collection, whereas the BCubed F-score is the harmonic mean of BCubed precision and recall:

$$\text{BCubed precision} = \frac{1}{N}\sum_{i=1}^{N} \text{precision}(d_i), \qquad \text{BCubed recall} = \frac{1}{N}\sum_{i=1}^{N} \text{recall}(d_i),$$

$$\text{BCubed F} = 2 \times \frac{\text{BCubed precision} \times \text{BCubed recall}}{\text{BCubed precision} + \text{BCubed recall}}.$$

Regarding the authorship-link ranking task, we use *average precision* (AP) to evaluate submissions. This is a standard scalar evaluation measure for ranked retrieval results. Given a ranked list of authorship links for a document collection, average precision is the average of non-interpolated precision values at all ranks where true authorship links were found. Let L be the set of ranked links provided by a submitted system and T the set of true links for a given collection. If l_i is the authorship link at i-th position of L then a relevance function, precision at cutoff i in the ranked list, and AP are defined as follows:

$$\text{relevant}(i) = \begin{cases} 1 & \text{if } l_i \in T \\ 0 & \text{otherwise,} \end{cases} \qquad \text{precision}(i) = \frac{\sum_{j=1}^{i} \text{relevant}(j)}{i},$$

$$\text{AP} = \frac{\sum_{i=1}^{|L|} \text{precision}(i) \times \text{relevant}(i)}{|T|}.$$

It is important to note that AP does not punish verbosity, i.e., every true link counts even if it is at a very low rank. Therefore, by providing all possible authorship links one can attempt to maximize AP. In order to show how effective a system is in top-ranked predictions, we also provide *R-precision* (RP) and *P@10*, which are defined as follows:

$$\text{R} - \text{precision} = \frac{\sum_{i=1}^{R} \text{relevant}(i)}{R}, \qquad \text{P@10} = \frac{\sum_{i=1}^{10} \text{relevant}(i)}{10},$$

where R is the number of true authorship links. Focusing on either the top R or the top 10 results, these metrics ignore all other answers.

For multiple instances of author clustering problems, mean scores of all the above measures are used to evaluate the overall performance of submissions in all available collections. Finally, submissions are ranked according to Mean F-score (MF) and Mean Average Precision (MAP) for complete author clustering and authorship-link ranking, respectively.

Table 2. Overall evaluation results in author clustering (mean values for all clustering problems).

Participant	Complete clustering			Authorship-link ranking			Runtime
	B3 F	B3 rec.	B3 prec.	MAP	RP	P@10	
Bagnall	**0.822**	0.726	0.977	**0.169**	**0.168**	**0.283**	63:03:59
Gobeill	0.706	0.767	0.737	0.115	0.131	0.233	00:00:39
Kocher	**0.822**	0.722	**0.982**	0.054	0.050	0.117	00:01:51
Kuttichira	0.588	0.720	0.512	0.001	0.010	0.006	00:00:42
Mansoorizadeh *et al.*	0.401	0.822	0.280	0.009	0.012	0.011	00:00:17
Sari & Stevenson	0.795	0.733	0.893	0.040	0.065	0.217	00:07:48
Vartapetiance & Gillam	0.234	**0.935**	0.195	0.012	0.023	0.044	03:03:13
Zmiycharov *et al.*	0.768	0.716	0.852	0.003	0.016	0.033	01:22:56
BASELINE-Random	0.667	0.714	0.641	0.002	0.009	0.013	–
BASELINE-Singleton	0.821	0.711	**1.000**	–	–	–	–
BASELINE-Cosine	–	–	–	0.060	0.074	0.139	–

Table 3. Evaluation results (mean BCubed F-score) for the complete author clustering task.

Participant	Overall	Articles	Reviews	English	Dutch	Greek	$r\approx0.9$	$r\approx0.7$	$r\approx0.5$
Bagnall	**0.822**	**0.817**	**0.828**	**0.820**	**0.815**	0.832	0.931	0.840	**0.695**
Kocher	**0.822**	**0.817**	0.827	0.818	**0.815**	**0.833**	**0.933**	**0.843**	0.690
BASELINE-Singleton	0.821	**0.819**	0.823	**0.822**	**0.819**	0.822	**0.945**	0.838	0.680
Sari & Stevenson	0.795	0.789	0.801	0.784	0.789	0.813	0.887	0.812	0.687
Zmiycharov *et al.*	0.768	0.761	0.776	0.781	0.759	0.765	0.877	0.777	0.651
Gobeill	0.706	0.800	0.611	0.805	0.606	0.707	0.756	0.722	0.639
BASELINE-Random	0.667	0.666	0.667	0.668	0.665	0.667	0.745	0.678	0.577
Kuttichira	0.588	0.626	0.550	0.579	0.584	0.601	0.647	0.599	0.519
Mansoorizadeh *et al.*	0.401	0.367	0.435	0.486	0.256	0.460	0.426	0.373	0.403
Vartapetiance & Gillam	0.234	0.284	0.183	0.057	0.595	0.049	0.230	0.241	0.230

Table 4. Evaluation results (MAP) for the authorship-link ranking task.

Participant	Overall	Articles	Reviews	English	Dutch	Greek	$r\approx0.9$	$r\approx0.7$	$r\approx0.5$
Bagnall	**0.169**	**0.174**	**0.163**	**0.126**	**0.109**	**0.272**	**0.064**	**0.186**	**0.257**
Gobeill	0.115	0.119	0.110	0.097	0.079	0.168	0.040	0.105	0.198
BASELINE-Cosine	0.060	0.063	0.057	0.053	0.053	0.074	0.019	0.054	0.107
Kocher	0.054	0.047	0.061	0.032	0.044	0.085	0.042	0.058	0.063
Sari & Stevenson	0.040	0.033	0.047	0.009	0.042	0.069	0.017	0.041	0.062
Vartapetiance & Gillam	0.012	0.010	0.014	0.014	0.006	0.016	0.010	0.008	0.017
Mansoorizadeh *et al.*	0.009	0.013	0.004	0.006	0.010	0.010	0.002	0.009	0.014
Zmiycharov *et al.*	0.003	0.002	0.004	0.001	0.000	0.009	0.002	0.003	0.004
BASELINE-Random	0.002	0.002	0.001	0.001	0.002	0.002	0.001	0.001	0.002
Kuttichira	0.001	0.002	0.001	0.001	0.002	0.001	0.001	0.002	0.001

3.6 Evaluation Results

Following the practice of previous PAN editions, each participant submitted their software to the TIRA experimentation platform where they were also able to run their software on training and evaluation datasets [13, 39]. We then reviewed the participants' runs and provided feedback in cases when a software did not complete its run successful. Although the participants could examine various versions of their software, only one run was considered in the final evaluation. Table 2 shows the overall results for both complete clustering and authorship-link ranking on the evaluation dataset. All evaluation measures are averaged over the 18 evaluation problems. The runtime of each submission is also provided.

A more detailed view of the results for the complete clustering task is shown in Table 3. Participants and baseline methods are ranked according to overall BCubed F-score, while partial results are also given for the available genres (articles or reviews), languages (English, Dutch, or Greek), and the (approximate) value of r (0.9, 0.7, or

Table 5. Left table: Number of clusters detected by each participant in the evaluation dataset. Number of documents (N) and authors (k) per are also given. Right table: Number of authorship links detected by each participant in the evaluation dataset. Number of true links and maximum links per problem are also given.

ID	N	k	Bagnall	Gobeill	Kocher	Kuttichira	Mansoorizadeh *et al.*	Sari and Stevenson	Vartapetiance and Gillam	Zmiycharov *et al.*
001	70	50	70	61	68	36	20	60	1	59
002	70	35	70	54	68	36	20	60	1	63
003	70	64	70	56	68	36	20	60	1	60
004	80	58	80	77	78	36	25	70	1	73
005	80	72	79	78	78	36	31	70	1	74
006	80	42	78	78	77	36	29	70	1	71
007	57	42	54	50	55	36	1	48	42	47
008	57	50	55	48	55	36	11	48	39	49
009	57	30	56	49	55	36	2	48	46	49
010	100	88	99	28	97	36	20	90	28	84
011	100	51	96	23	98	36	20	90	25	86
012	100	71	98	29	98	36	20	90	33	80
013	70	50	69	61	68	36	20	60	1	55
014	70	35	70	63	68	36	20	60	1	59
015	70	62	70	66	66	36	20	60	1	58
016	70	51	56	29	67	36	20	60	1	58
017	70	64	59	23	68	36	20	60	1	58
018	70	37	58	31	67	36	20	60	1	53

ID	true links	max links	Bagnall	Gobeill	Kocher	Kuttichira	Mansoorizadeh *et al.*	Sari and Stevenson	Vartapetiance and Gillam	Zmiycharov *et al.*
001	33	2415	2415	2415	2415	68	170	14	526	19
002	113	2415	2415	2415	2415	57	189	11	529	18
003	7	2415	2415	2415	2415	67	262	13	611	16
004	30	3160	3160	3160	3160	120	605	23	2705	11
005	10	3160	3160	3160	3160	126	614	18	2750	9
006	68	3160	3160	3160	3160	88	605	21	2691	10
007	24	1596	1596	1596	1596	52	1596	11	36	18
008	8	1596	1596	1596	1596	42	475	11	40	23
009	65	1596	1596	1596	1596	51	1486	30	21	24
010	16	4950	4950	4950	4950	214	323	11	94	79
011	76	4950	4950	4950	4950	261	464	14	107	98
012	37	4950	4950	4950	4950	229	297	13	91	97
013	24	2415	2415	2415	2415	62	288	12	616	94
014	52	2415	2415	2415	2415	114	444	13	642	104
015	9	2415	2415	2415	2415	70	335	13	833	95
016	24	2415	2415	2415	2415	108	335	14	954	36
017	7	2415	2415	2415	2415	96	932	23	865	30
018	44	2415	2415	2415	2415	87	859	23	1134	51

0.5). As can be seen, the BASELINE-Singleton method is only narrowly beaten by two submissions. Both of these submissions were better than BASELINE-Singleton in handling reviews and Greek documents and, quite predictably, when r is lower than 0.9. The approach of Bagnall [6] is slightly better than Kocher's [22] (overall F-score 0.8223 vs. 0.8218). In terms of efficiency, Kocher's approach is much more faster than Bagnall's. In general when r decreases (i.e., more multi-item clusters are available), the performance of all submissions is negatively affected.

On the other side of the table, there are 3 submissions with overall F-score less than BASELINE-Random mainly because they failed to accurately predict the number of clusters in each problem. Table 5 (left) shows the number of clusters formed by each participant per problem together with the number of documents and number of true clusters (authors) per problem. As can be seen, the approach of Kuttichira *et al.* [26] always guesses the same number of clusters while the approaches of Mansoorizadeh *et al.* [31] and Vartapetiance and Gillam [58] tend to predict 20 and 1 clusters per problem, respectively. On the other hand, the successful approaches of Bagnall [6] and Kocher [22] resemble BASELINE-Singleton by being modest in forming clusters with more than one document.

Table 4 shows the performance results for authorship-link ranking. Participants and baseline methods are ranked by their overall MAP score while partial results for genre, language, and r value are also given. Roughly half of participants achieve better results on articles and the other half perform better on reviews. The Greek part seems to be easier in comparison to the English and Dutch parts. Finally, the performance of all

submissions is improved when r decreases and more authorship links are available. Only two approaches are better than BASELINE-Cosine. This is surprising given that this baseline approach is not sophisticated and does not attempt to explore stylistic information. On the other side of the table, a couple of submissions were less effective than or very close to BASELINE-Random.

Table 5 (right) shows the number of authorship links detected by each submission per evaluation problem. Some participants chose to report all possible authorship links attempting to maximize MAP. However, it should be noted that the approaches of Bagnall [6] and Gobeill [12], which achieve the best MAP score, are also the winners with respect to the measures RP and P@10 as shown in Table 2. It is also remarkable that the submission of Sari and Stevenson [47] detects only a few authorship links but still achieves a relatively high P@10 score.

4 Author Diarization

This section presents the task of author diarization. More specifically, it includes task definitions and evaluation datasets, a survey of submissions and the baselines, and it describes the evaluation results in detail.

4.1 Task Definition

The author diarization task continues the previous PAN tasks from 2009-2011 on intrinsic plagiarism detection [43, 36, 37]. As already pointed out, the task is extended and generalized by introducing within-document author clustering problems. Following the methodology of intrinsic approaches, any comparison with external sources are disallowed for all subtasks. In particular, the author diarization task consists of the following three subtasks:

A) *Traditional intrinsic plagiarism detection*. Assuming a major author who wrote at least 70% of a document, this subtask is to find the remaining text portions written by one or several others.
B) *Diarization with a given number of authors*. The basis for this subtask is a document which has been composed by a known number of authors with the goal to group the individual text fragments by authors, i.e., build author clusters.
C) *Unrestricted diarization*. As a tightening variant of the previous scenario, the number of collaborating authors is not given as an input variable for the this subtask. Thus, before/during analyzing and attributing the text, also the correct number of clusters, i.e., writers, has to be guessed.

To ensure consistency throughout the subtasks and also to emphasize the similarity between them, Task A also requires to construct author clusters. In this special case, there exactly two clusters exist: one for the main author and one for the intrusive fragments. The participants were free to create more than one cluster for the latter, e.g., to create a cluster for each intrusive text fragment. For all three subtasks, training datasets (see Section 4.2) were provided in order to allow for adjusting and tuning the developed algorithms prior to the submission.

Table 6. Parameters for generating the datasets.

Parameter	Subtask A	Subtask B	Subtask C
number of authors	2-10	2-10	2-10
total number of words (max.)	3500	7000	7000
authorship boundaries	sentence/paragraph	sentence/paragraph	sentence/paragraph
main author contribution	70-99%	–	–
intrusive sections	1-20	–	–
author distribution	–	uniformly/randomly	uniformly/randomly

Table 7. Dataset statistics.

Parameter	Range	Subtask A		Subtask B		Subtask C	
		train	test	train	test	train	test
#documents		65	29	55	31	54	29
#authors	2-4	77%	72%	33%	35%	39%	38%
	5-7	20%	21%	36%	35%	39%	31%
	8-10	3%	7%	31%	29%	22%	31%
#words	<1000	31%	28%	5%	6%	9%	14%
	1000-2000	46%	41%	16%	29%	24%	21%
	2000-3000	17%	17%	16%	23%	22%	21%
	3000-4000	6%	14%	25%	19%	13%	24%
	≥ 4000	–	–	36%	19%	31%	17%
authorship boundaries	sentence	49%	48%	56%	45%	52%	48%
	paragraph	51%	52%	44%	55%	11%	52%
intrusive sections	0-10%	42%	38%	–	–	–	–
	10-20%	48%	48%	–	–	–	–
	20-30%	11%	14%	–	–	–	–
#intrusive sections	1-5	83%	72%	–	–	–	–
	6-10	15%	21%	–	–	–	–
	>10	2%	7%	–	–	–	–
author distribution	uniformly	–	–	69%	48%	41%	48%
	randomly	–	–	31%	52%	59%	52%

4.2 Evaluation Datasets

For all subtasks, distinct training and test datasets have been provided, which are all based on the Webis Text Reuse Corpus 2012 (Webis-TRC-12) [41]. The original corpus contains essays on 150 topics used at the TREC Web Tracks 2009-2011 (e.g., see [8]). The essays were written by (semi-)professional writers hired via crowdsourcing. For each essay, a writer was assigned a topic (e.g., "Barack Obama: write about Obama's family"), then asked to use the ChatNoir search engine [40] to retrieve relevant sources of information, and to compose an essay from the search results, reusing text from the retrieved web pages. All sources of the resulting document were annotated, so that the origin of each text fragment is known.

From these documents, assuming that each distinct source represents a different author, the respective datasets for all subtasks have been randomly generated by varying several parameters as shown in Table 6. Beside the number of authors and words, also authorship boundary types have been altered to be on sentence or paragraph levels: i.e., authors may switch either after or even within sentences or only after whole paragraphs (separated by one or more line breaks). For the diarization datasets, the authorship distribution has been configured to either be uniformly distributed (each author contributed approximately the same amount) or randomly distributed (resulting in contributions like: authors $(A, B, C, D) \rightarrow (94, 3, 2, 1)\%$). As the original corpus has

already been partly used and published, the test documents are created from previously unpublished documents only. Table 7 shows statistics of the generated datasets.

4.3 Survey of Submissions

We received software submissions from two teams, both solving all three subtasks. This section summarizes the main principles of these approaches.

Approach of Kuznetsov *et al.* [27]. The authors describe an algorithm that operates on all three subtasks with only slight modifications. At first, the text is split into sentences, and for each sentence selected stylometric features, including word and n-gram frequencies ($n = \{1, 3, 4\}$) are calculated. A *relational* frequency is calculated by comparing the features of the selected sentence with the global document features which results in three features for each measure (and n): a 5%, 50% and 95% percentile. Additionally, features such as sentence length, punctuation symbol counts, and selected part-of-speech (POS) tag frequencies are calculated. The extracted features serve as input for a classifier, namely an implementation of Gradient Boosting Regression Trees [9], which outputs a model, i.e., an author style function. For the prediction of the label of a sentence (plagiarized, non-plagiarized), also nearby sentences are included in the decision. Finally, outliers are detected by defining a threshold, which compares to the degree of mismatch with the main author style that is calculated by the classifier. To train the classifier, the PAN 2011 dataset for intrinsic plagiarism detection was used [37].

To solve the diarization task for known numbers of authors, the algorithm is slightly modified. Instead of finding outliers on a threshold basis, a segmentation is calculated by using a Hidden Markov Model with Gaussian emissions [20]. Finally, the number of authors is estimated to solve Task C by computing segmentations for $\#authors = [2, ..20]$ and measuring the clusterings' discrepancy.

Approach of Sittar *et al.* [48]. Reflecting the similarity between all three subtasks, the submitted algorithm of this team represents an "all-in-one" solution for all three tasks by calculating clusters. Like the previous approach, it is based on analyzing features on individual sentences. A total of 15 lexical metrics are extracted, including character and word counts, average word length, and ratios of digits, letters, or spaces. With these features, a distance between every pair of sentences is calculated using the *ClustDist* metric [15]. Using these distances, i.e., a feature vector consisting of the distances to all other sentences, K-Means is applied to generate clusters.

For tackling the respective subtasks, the only modification is the predefined number of clusters that is given to K-Means. In case of the plagiarism detection task, the number of clusters is set to two (one for the main author and one for the intrusive authors). For the diarization tasks, the number of clusters is set to the given corresponding authors (Task B) or randomly assigned (Task C). As a final optimization step, also the grouping of sentences has been evaluated. The distances are not calculated for single sentences only, but also for sentence groups. The authors report that the best results on the provided training dataset was achieved by using sentence groups of size 7 (Task A) and 5 (Task B, and C). Consequently, this configuration has been used for the test dataset as well.

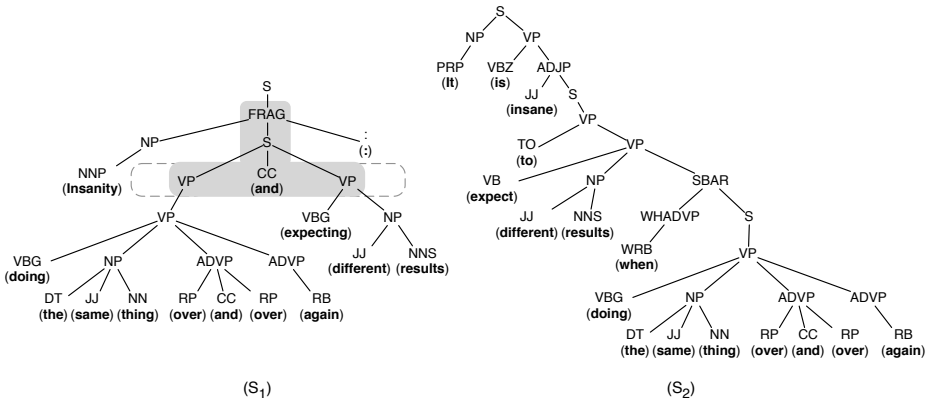

Figure 2. PQPlagInn Baseline: Parse Trees of Sentences S_1 and S_2.

4.4 Baselines

To quantify the performance of the submitted approaches, baselines for each subtask have been computed. The baseline for Task A is the PQPlagInn approach [56]. It is the best working variant of the PlagInn algorithm [55] and operates solely on the grammar syntax of authors. The main idea is that authors differ in their way of constructing sentences, and that these differences can be used as style markers to identify plagiarism. For example, Figure 2 shows the syntax trees (parse trees) of the Einstein quote *"Insanity: doing the same thing over and over again and expecting different results" (S_1)* and the slightly modified version *"It is insane to expect different results when doing the same thing over and over again" (S_2)*. It can be seen that the trees differ significantly, although the semantic meaning is the same. To quantify such differences of parse trees, the concept of pq-grams is used [4]. In a nutshell, pq-grams can be seen as "n-grams for trees" since they represent structural parts of the tree, where p defines how much nodes are included vertically, and q defines the number of nodes to be considered horizontally. The set of possible pq-grams serve as the feature vectors that are compared with the global document's features by using sliding windows and a selected distance metric. Finally, suspicious sentences are found using (several) thresholds and applying a filtering/grouping algorithm [55]. As a baseline for Task A, the PQPlagInn algorithm is used in an unoptimized version, i.e., optimized for the PAN 2011 dataset [37] and not considering the specifications of the current dataset. For example, the facts that the main author contribution is at least 70%, or that it can be assumed implicitly that no document is plagiarism-free are disregarded. Thus, the performance of PQPlagInn should give a stable orientation, but still provide room for improvement.

As author diarization is tackled for the first time at PAN 2016 there exist, to the best of our knowledge, no comparable algorithms for this specific task, a random baseline has been created for Tasks B and C. This has been done by dividing the document into n parts of equal length, and assigning each part to a different author. Here, n is set to the exact number of authors for Task B, and randomly chosen for Task C.

Table 8. Intrinsic plagiarism detection results (Task A).

Rank	Team	Micro-averaged perf.			Macro-averaged perf.		
		Recall	Precision	F	Recall	Precision	F
1	Kuznetsov *et al.*	**0.19**	**0.29**	**0.22**	**0.15**	**0.28**	**0.17**
-	BASELINE	0.16	0.25	0.18	**0.15**	0.24	0.16
2	Sittar *et al.*	0.07	0.14	0.08	0.10	0.14	0.10

4.5 Evaluation Results

The participants submitted their approaches as executable software to TIRA [13, 39], where they were executed against the test datasets hosted there. Performances on the provided training data were visible immediately to participants, whereas results on the test data were revealed only after the submission deadline only. This section details the evaluation results of the submitted approaches for each subtask.

Task A: Intrinsic Plagiarism Detection Results The performance of the intrinsic plagiarism detection subtask has been measured with the metrics proposed in by Potthast *et al.* [42]. The proposed micro-averaged variants of the metrics incorporate the length of each plagiarized section, whereas the macro-averaged variants do not. To illustrate the difference, consider the following situation: a document contains two plagiarized sections, but the second one is very short compared to the first one. If an algorithm finds 100% of the first section, but misses the second one, the macro-F would be only 50%, whereas the micro-F can easily exceed 90% (depending on section lengths). Favoring coverage of different sources and conforming with the previous PAN plagiarism detection tasks, the final ranking is based on the macro-averaged scores.

Table 8 shows the final results. Kuznetsov *et al.* [27] exceed the baseline, achieving a macro-F of 0.17. Interestingly, the approach of Sittar *et al.* [48] achieves better macro-averaged scores than micro-averaged ones, whereas this is the other way around for the other approaches which is closer to our expectation. In what follows, detailed analyses depending on different dataset parameters of the test datasets are presented. Figure 3 (top left) shows the F-scores over ranges of documents lengths in terms of number of words. While there are no notable differences for documents with less than 2000 words, Kuznetsov *et al.* achieve a peak performance of over 0.4 for longer documents (2000-2500 words). The results with respect to the authorship border types are depicted in Figure 3 (bottom left). As expected, all approaches perform better on documents having intrusive sections only on paragraph boundaries, and not between or even within sentences. In Figure 3 (top right), the results per percentage of intrusive text is shown. With the exception of the two documents containing 20-25% intrusive text—which are obviously difficult to to identify—the chart reveals a steady increase of performance with the percentage of intrusive text. For documents with a very high percentage of intrusive text, Kuznetsov *et al.* achieved an F-score of nearly 0.6, and also the other approaches perform best on those documents. These performances correspond to the diarization results presented later, and emphasize once again that the latter tasks can also be seen as diarization tasks, e.g., with authorship contributions like $(A, B) \rightarrow (70, 30)\%$. As can be seen in Figure 3 (bottom right), the performances of Sittar *et al.* and the baseline do not change significantly with the number of intrusive

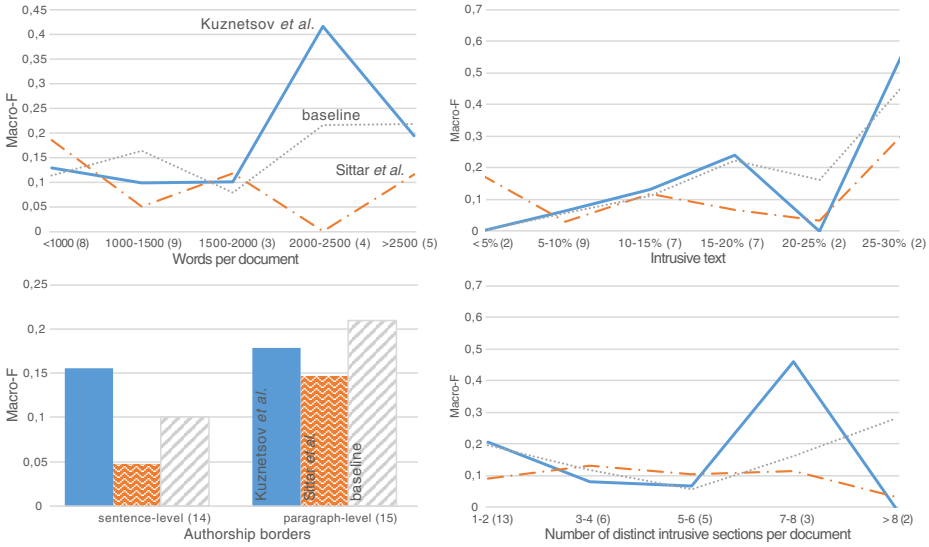

Figure 3. Subtask A: Macro-F performances with respect to words per document (top left), authorship borders (bottom left), percentage of intrusive text (top right), and number of intrusive sections (bottom right). Brackets denote the number of applicable documents.

sections. Solely Kuznetsov *et al.* achieves a peak performance on the three documents having seven or eight intrusive sections. Finally, Table 9 shows the three best results on individual problem instances. Kuznetsov *et al.* achieve a top performance of 0.94 with perfect precision. Sittar *et al.* reach an F-score of 0.58 on the `problem-9` document, which is also among the best three for all approaches.

Tasks B and C: Diarization Results The diarization subtasks have been measured with the BCubed clustering metrics [3], as they reflect the clustering nature on the one hand, and are also used for the evaluations of the PAN 2016 across-document clustering problems on the other hand (see Section 3). Table 10 shows the respective results for the Tasks B and C. They reveal that the tasks are hard to tackle, as none of the participants surpasses the random baseline, neither for Task B nor C. Nevertheless, Kuznetsov *et al.* achieves the highest precision for both subtasks, meaning that characters grouped together really belong together (with an accuracy significantly beyond random guessing).

As excepted, the results for an unknown number of authors (Task C) are slightly below the results where the exact number of authors are known beforehand (Task B). An exception is Sittar *et al.*'s approach, whose results are the opposite of the expectation. To investigate possible upsides and downsides of the approaches, detailed evaluations depending on parameters of the test datasets have been conducted. Figure 4 (top) shows the performance scores with respect to document length. While performances for Task B are quite stable, it can be seen that they decrease with the number of words when the number of authors had to be estimated. The results with respect to the number of corresponding authors are presented in Figure 4 (middle). Here also the scores follow no recognizable pattern for Task B, but become lower with the

Table 9. Best results per problem instance of Task A.

Instance	Authors	Words	Border	Intrusive text		Macro-averaged perf.		
				%	Sections	Recall	Precision	F
problem-3	8	2952	sent	29.78	7	0.88	1.00	0.94
problem-6	3	2375	par	16.37	2	0.69	1.00	0.82
problem-9	2	956	par	28.44	2	0.64	1.00	0.78

(a) Kuznetsov *et al.*

Instance	Authors	Words	Border	Intrusive text		Macro-averaged perf.		
				%	Sections	Recall	Precision	F
problem-9	2	956	par	28.44	2	0.45	0.85	0.58
problem-23	5	1259	par	19.35	4	0.50	0.42	0.46
problem-20	3	1870	par	4.10	6	0.38	0.34	0.36

(b) Sittar *et al.*

Instance	Authors	Words	Border	Intrusive text		Macro-averaged perf.		
				%	Sections	Recall	Precision	F
problem-9	2	956	par	28.44	2	0.82	1.00	0.90
problem-6	3	2375	par	16.37	2	0.81	0.50	0.62
problem-3	8	2952	sent	29.78	7	0.47	0.50	0.48

(c) Baseline

Table 10. Diarization results (Tasks B and C).

Number of authors	Rank	Team	BCubed		
			Recall	Precision	F
known (Task B)	-	BASELINE	**0.67**	0.52	**0.58**
	1	Kuznetsov *et al.*	0.46	**0.64**	0.52
	2	Sittar *et al.*	0.47	0.28	0.32
unknown (Task C)	-	BASELINE	**0.62**	0.56	**0.52**
	1	Kuznetsov *et al.*	0.42	**0.64**	0.48
	2	Sittar *et al.*	0.47	0.31	0.35

number of authors that corresponded in Task C. Remarkably, Kuznetsov *et al.* significantly exceeds the baseline for documents with two to four authors. As depicted in Figure 4 (bottom), Tasks B and C reveal similar results depending on the distribution of the corresponding authors. The results for randomly distributed contributions (e.g., $(A, B, C) \rightarrow (80, 7, 13)\%$) are generally better than those for uniformly distributed contributions (e.g., $(A, B, C) \rightarrow (33, 32, 35)\%$). An explanation for this outcome may be that the submitted approaches are designed for, or originate from intrinsic plagiarism detection, focusing on finding outliers. In case of the diarization problems, this seems not to be a good choice, especially when the contributions among authors are equally distributed, i.e., when there are no "outliers". Finally, also for these subtasks, the borders between authorships have been altered, i.e., either within sentences, at the end of sentences, or after paragraphs only. In contrast to Task A, there were no significant differences in performances for Tasks B and C with respect to this parameter. Although the baseline could not be exceeded on the whole dataset, Table 11 underlines

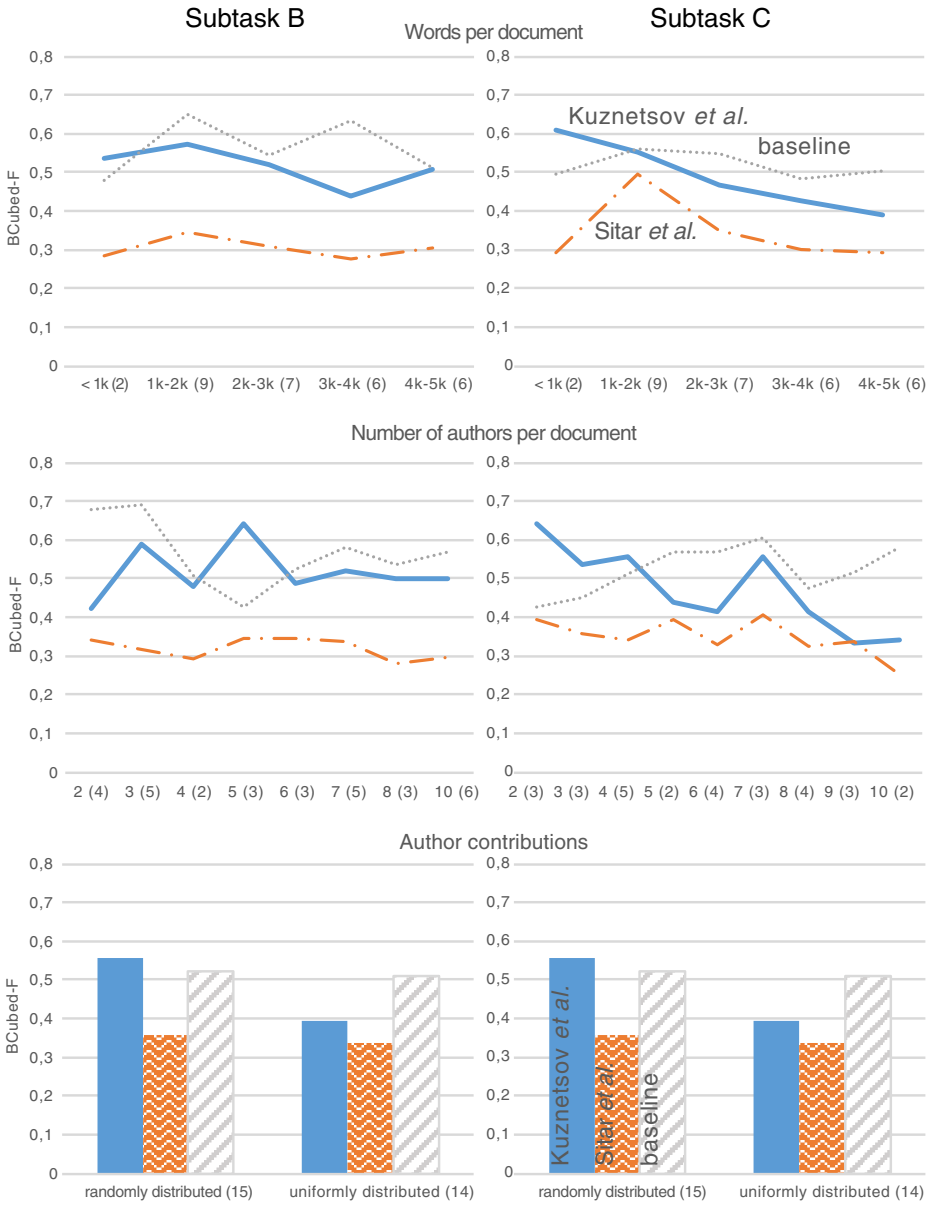

Figure 4. Subtask B (left) and Subtask C (right): performance with respect to words per document, number of authors per document, and author contributions.

that the approaches nevertheless produce very good results on individual instances. On `problem-12`, Kuznetsov *et al.* achieve a BCubed F-score of 0.88 for Task B. Remarkably, the score of a document containing ten different authors is among the best

Table 11. Best results per problem instance of Tasks B and C.

					BCubed		
Instance	**Authors**	**Words**	**Border**	**Distribution**	**Recall**	**Precision**	**F**
problem-12 (B)	3	1217	par	uniform	0.88	0.88	0.88
problem-6 (B)	3	1245	sent	random	0.87	0.75	0.81
problem-11 (B)	10	3029	par	random	0.66	0.93	0.78
problem-17 (C)	2	548	sent	random	0.79	0.68	0.73
problem-25 (C)	2	838	sent	random	0.63	0.70	0.67
problem-27 (C)	4	2916	par	random	0.70	0.60	0.64

(a) Kuznetsov *et al.*

					BCubed		
Instance	**Authors**	**Words**	**Border**	**Distribution**	**Recall**	**Precision**	**F**
problem-4 (B)	6	6252	sent	random	0.84	0.42	0.56
problem-28 (B)	5	1512	sent	uniform	0.91	0.34	0.49
problem-2 (B)	2	1109	par	random	0.38	0.51	0.43
problem-13 (C)	2	1533	sent	uniform	1.00	0.44	0.61
problem-3 (C)	8	1768	par	random	0.61	0.52	0.56
problem-12 (C)	7	1750	sent	random	0.93	0.39	0.55

(b) Sittar *et al.*

three, with an F-score of 0.78 at a precision of 0.93. The best result of Sittar *et al.* is on `problem-13` with a BCubed F-score of 0.61 on the designated more difficult Task C.

5 Discussion

For the first time, PAN 2016 focused on unsupervised authorship attribution, an under-explored line of research that is associated with important applications. Two main problems were studied: clustering by authorship across documents and clustering by authorship within documents. In general, these are quite challenging tasks that are hard to model, and the performance of submitted approaches, in many cases very close or inferior to simple baseline methods, indicates that there is a lot of space for improvement.

The author clustering task introduced authorship-link ranking as a separate retrieval problem in unsupervised authorship attribution. This problem is useful when huge document collections are available and the main task is to help human experts to closely examine specific cases, the most probable authorship links. The best results were achieved by a modification of the winner approach at the PAN 2015 authorship verification task [6]. This indicates that authorship verification and author clustering are strongly related tasks and the expertise gained in one field can help providing reliable solutions to the other. Moreover, we introduced the ratio r that represents both the quantity of authorship links and the number of single-item clusters. It has been shown that when r is high, a naive baseline approach that assigns each document to a separate cluster is hard to beat. It is expected that if a method is able to estimate the r value in a given collection, then it is more likely to provide reliable answers. The author clustering problem can become even more challenging if we drop the assumption that all documents within a collection belong to the same genre. In that case, it would be extremely difficult to separate stylistic similarities and differences that are caused by genre or the personal

style of authors. In addition, in most of the clustering problems provided at PAN 2016, the documents within a collection fall into the same general thematic area. Although the specific topics of documents differ, it would be even more challenging if the thematic area of documents would vary.

The author diarization task focused on the problem of clustering by authorship within documents. A traditional intrinsic plagiarism detection subtask was used as an entry point, keeping up with previous PAN events. Moreover, to generalize the problem, designated subtasks have been added that deal with the decomposition of multi-author documents, where the number of corresponding authors was either given or had to be estimated. Both submitted approaches tackle all subtasks and rely on an analysis on the sentence-level, extracting lexical and syntactic features and feeding them to different machine learning techniques. One of the approaches exceeds the baseline for intrinsic plagiarism detection, whereas a random baseline for the novel subtasks focusing on clustering of text by authors could not be outperformed. The results of the diarization task underline once again that intrinsic plagiarism detection represents a difficult problem, and that clustering by authorship within documents seems to be even harder. A possible explanation of the low scores for the latter problem is that the approaches only modify intrinsic plagiarism detection algorithms. It can be assumed that by tailoring algorithms to author clustering within documents, results can be improved significantly. Moreover, as the author diarization task was held the first time at PAN 2016 receiving only two submissions, future PAN labs may attract more participants that help narrowing the gap.

Acknowledgements

We thank the participating teams of this shared task. Our special thanks go to Adobe Systems Inc. for sponsoring the event.

Bibliography

[1] Almishari, M., Tsudik, G.: Exploring linkability of user reviews. In: Computer Security, ESORICS 2012, pp. 307–324. Springer (2012)

[2] Amigó, E., Gonzalo, J., Artiles, J., Verdejo, F.: A comparison of extrinsic clustering evaluation metrics based on formal constraints. Information Retrieval 12(4), 461–486 (2009)

[3] Amigó, E., Gonzalo, J., Artiles, J., Verdejo, F.: A comparison of extrinsic clustering evaluation metrics based on formal constraints. Information retrieval 12(4), 461–486 (2009)

[4] Augsten, N., Böhlen, M., Gamper, J.: The pq-Gram Distance between Ordered Labeled Trees. ACM Transactions on Database Systems (TODS) (2010)

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

[6] Bagnall, D.: Authorship Clustering Using Multi-headed Recurrent Neural Networks. In: CLEF 2016 Working Notes. CEUR Workshop Proceedings, CLEF and CEUR-WS.org (2016)

[7] Choi, F.Y.: Advances in Domain Independent Linear Text Segmentation. In: Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference. pp. 26–33. Association for Computational Linguistics (2000)
[8] Clarke, C.L., Craswell, N., Soboroff, I., Voorhees, E.M.: Overview of the TREC 2009 web track. Tech. rep., DTIC Document (2009)
[9] Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Annals of statistics pp. 1189–1232 (2001)
[10] Giannella, C.: An improved algorithm for unsupervised decomposition of a multi-author document. Technical Papers, The MITRE Corporation (February 2014)
[11] Glover, A., Hirst, G.: Detecting stylistic inconsistencies in collaborative writing. In: The New Writing Environment, pp. 147–168. Springer (1996)
[12] Gobeill, J.: Submission to the Author Clustering Task at PAN-2016. http://www.uni-weimar.de/medien/webis/events/pan-16 (2016), HES-SO
[13] Gollub, T., Stein, B., Burrows, S.: Ousting Ivory Tower Research: Towards a Web Framework for Providing Experiments as a Service. In: Hersh, B., Callan, J., Maarek, Y., Sanderson, M. (eds.) 35th International ACM Conference on Research and Development in Information Retrieval (SIGIR 12). pp. 1125–1126. ACM (Aug 2012)
[14] Graham, N., Hirst, G., Marthi, B.: Segmenting documents by stylistic character. Natural Language Engineering 11(04), 397–415 (2005)
[15] Guthrie, D.: Unsupervised Detection of Anomalous Text. Ph.D. thesis, University of Sheffield (2008)
[16] Holmes, D.I.: The evolution of stylometry in humanities scholarship. Literary and Linguistic Computing 13(3), 111–117 (1998)
[17] Holmes, D., Forsyth, R.: The federalist revisited: New directions in authorship attribution. Literary and Linguistic Computing 10(2), 111–127 (1995)
[18] Iqbal, F., Binsalleeh, H., Fung, B.C.M., Debbabi, M.: Mining writeprints from anonymous e-mails for forensic investigation. Digital Investigation 7(1-2), 56–64 (2010)
[19] Juola, P.: Authorship Attribution. Foundations and Trends in Information Retrieval 1, 234–334 (2008)
[20] Keogh, E., Chu, S., Hart, D., Pazzani, M.: Segmenting time series: A survey and novel approach. Data mining in time series databases 57, 1–22 (2004)
[21] Kestemont, M., Luyckx, K., Daelemans, W.: Intrinsic Plagiarism Detection Using Character Trigram Distance Scores. In: Notebook Papers of the 5th Evaluation Lab on Uncovering Plagiarism, Authorship and Social Software Misuse (PAN). Amsterdam, The Netherlands (September 2011)
[22] Kocher, M.: UniNE at CLEF 2016: Author Clustering. In: CLEF 2016 Working Notes. CEUR Workshop Proceedings, CLEF and CEUR-WS.org (2016)
[23] Koppel, M., Akiva, N., Dershowitz, I., Dershowitz, N.: Unsupervised decomposition of a document into authorial components. In: Lin, D., Matsumoto, Y., Mihalcea, R. (eds.) Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics. pp. 1356–1364 (2011)
[24] Koppel, M., Schler, J., Argamon, S.: Computational methods in authorship attribution. Journal of the American Society for Information Science and Technology 60(1), 9–26 (2009)
[25] Koppel, M., Winter, Y.: Determining if two documents are written by the same author. Journal of the American Society for Information Science and Technology 65(1), 178–187 (2014)
[26] Kuttichira, D., Krishnan, K.G., Pooja, A., Kumar, M.A., Mahalakshmi, S.: Submission to the Author Clustering Task at PAN-2016. http://www.uni-weimar.de/medien/webis/events/pan-16 (2016), Amrita Vishwa Vidyapeetham

[27] Kuznetsov, M., Motrenko, A., Kuznetsova, R., Strijov, V.: Methods for Intrinsic Plagiarism Detection and Author Diarization. In: Working Notes Papers of the CLEF 2016 Evaluation Labs. CEUR Workshop Proceedings, CLEF and CEUR-WS.org (Sep 2016)
[28] Layton, R., Watters, P., Dazeley, R.: Automated unsupervised authorship analysis using evidence accumulation clustering. Natural Language Engineering 19, 95–120 (2013)
[29] Layton, R., Watters, P., Dazeley, R.: Evaluating authorship distance methods using the positive silhouette coefficient. Natural Language Engineering 19, 517–535 (2013)
[30] Luyckx, K., Daelemans, W., Vanhoutte, E.: Stylogenetics: Clustering based stylistic analysis of literary corpora. In: Workshop Toward Computational Models of Literary Analysis (2006)
[31] Mansoorizadeh, M., Aminiyan, M., Rahguy, T., Eskandari, M.: Multi Feature Space Combination for Authorship Clustering. In: CLEF 2016 Working Notes. CEUR Workshop Proceedings, CLEF and CEUR-WS.org (2016)
[32] Miro, X.A., Bozonnet, S., Evans, N., Fredouille, C., Friedland, G., Vinyals, O.: Speaker diarization: A review of recent research. Audio, Speech, and Language Processing, IEEE Transactions on 20(2), 356–370 (2012)
[33] Niezgoda, S., Way, T.P.: Snitch: A software tool for detecting cut and paste plagiarism. In: Proceedings of the 37th Technical Symposium on Computer Science Education (SIGCSE). pp. 51–55. ACM, Houston, Texas, USA (March 2006)
[34] Oberreuter, G., L'Huillier, G., Ríos, S.A., Velásquez, J.D.: Approaches for Intrinsic and External Plagiarism Detection. In: Notebook Papers of the 5th Evaluation Lab on Uncovering Plagiarism, Authorship and Social Software Misuse (PAN). Amsterdam, The Netherlands (September 2011)
[35] Ponte, J.M., Croft, W.B.: Text Segmentation by Topic. In: Research and Advanced Technology for Digital Libraries, pp. 113–125. Springer (1997)
[36] Potthast, M., Barrón-Cedeño, A., Eiselt, A., Stein, B., Rosso, P.: Overview of the 2nd International Competition on Plagiarism Detection. In: Braschler, M., Harman, D., Pianta, E. (eds.) Working Notes Papers of the CLEF 2010 Evaluation Labs (Sep 2010), http://www.clef-initiative.eu/publication/working-notes
[37] Potthast, M., Eiselt, A., Barrón-Cedeño, A., Stein, B., Rosso, P.: Overview of the 3rd International Competition on Plagiarism Detection. In: Notebook Papers of the 5th Evaluation Lab on Uncovering Plagiarism, Authorship and Social Software Misuse (PAN). Amsterdam, The Netherlands (September 2011)
[38] Potthast, M., Gollub, T., Hagen, M., Kiesel, J., Michel, M., Oberländer, A., Tippmann, M., Barrón-Cedeno, A., Gupta, P., Rosso, P., et al.: Overview of the 5th international competition on plagiarism detection. In: Notebook Papers of the 9th Evaluation Lab on Uncovering Plagiarism, Authorship and Social Software Misuse (PAN). Valencia, Spain (September 2013)
[39] Potthast, M., Gollub, T., Rangel, F., Rosso, P., Stamatatos, E., Stein, B.: Improving the Reproducibility of PAN's Shared Tasks: Plagiarism Detection, Author Identification, and Author Profiling. In: Kanoulas, E., Lupu, M., Clough, P., Sanderson, M., Hall, M., Hanbury, A., Toms, E. (eds.) Information Access Evaluation meets Multilinguality, Multimodality, and Visualization. 5th International Conference of the CLEF Initiative (CLEF 14). pp. 268–299. Springer, Berlin Heidelberg New York (Sep 2014)
[40] Potthast, M., Hagen, M., Stein, B., Graßegger, J., Michel, M., Tippmann, M., Welsch, C.: ChatNoir: A Search Engine for the ClueWeb09 Corpus. In: Hersh, B., Callan, J., Maarek, Y., Sanderson, M. (eds.) 35th International ACM Conference on Research and Development in Information Retrieval (SIGIR 12). p. 1004. ACM (Aug 2012)
[41] Potthast, M., Hagen, M., Völske, M., Stein, B.: Crowdsourcing Interaction Logs to Understand Text Reuse from the Web. In: Fung, P., Poesio, M. (eds.) Proceedings of the

51st Annual Meeting of the Association for Computational Linguistics (ACL 13). pp. 1212–1221. Association for Computational Linguistics (Aug 2013), http://www.aclweb.org/anthology/P13-1119
[42] Potthast, M., Stein, B., Barrón-Cedeño, A., Rosso, P.: An evaluation framework for plagiarism detection. In: Proceedings of the 23rd international conference on computational linguistics: Posters. pp. 997–1005. Association for Computational Linguistics (2010)
[43] Potthast, M., Stein, B., Eiselt, A., Barrón-Cedeño, A., Rosso, P.: Overview of the 1st International Competition on Plagiarism Detection. In: Stein, B., Rosso, P., Stamatatos, E., Koppel, M., Agirre, E. (eds.) SEPLN 09 Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse (PAN 09). pp. 1–9. CEUR-WS.org (Sep 2009), http://ceur-ws.org/Vol-502
[44] Reynar, J.C.: Topic segmentation: Algorithms and applications. IRCS Technical Reports Series p. 66 (1998)
[45] Reynar, J.C.: Statistical Models for Topic Segmentation. In: Proc. of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics. pp. 357–364 (1999)
[46] Samdani, R., Chang, K.W., Roth, D.: A discriminative latent variable model for online clustering. In: Proceedings of The 31st International Conference on Machine Learning. pp. 1–9 (2014)
[47] Sari, Y., Stevenson, M.: Exploring Word Embeddings and Character N-Grams for Author Clustering. In: CLEF 2016 Working Notes. CEUR Workshop Proceedings, CLEF and CEUR-WS.org (2016)
[48] Sittar, A., Iqbal, R., Nawab, A.: Author Diarization Using Cluster-Distance Approach. In: Working Notes Papers of the CLEF 2016 Evaluation Labs. CEUR Workshop Proceedings, CLEF and CEUR-WS.org (Sep 2016)
[49] Stamatatos, E.: A Survey of Modern Authorship Attribution Methods. Journal of the American Society for Information Science and Technology 60, 538–556 (2009)
[50] Stamatatos, E.: Intrinsic Plagiarism Detection Using Character n-gram Profiles. In: Notebook Papers of the 5th Evaluation Lab on Uncovering Plagiarism, Authorship and Social Software Misuse (PAN). Amsterdam, The Netherlands (September 2011)
[51] Stamatatos, E., Daelemans, W., Verhoeven, B., Juola, P., López-López, A., Potthast, M., Stein, B.: Overview of the author identification task at PAN 2015. In: Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015. (2015)
[52] Stamatatos, E., Daelemans, W., Verhoeven, B., Stein, B., Potthast, M., Juola, P., Sánchez-Pérez, M.A., Barrón-Cedeño, A.: Overview of the author identification task at PAN 2014. In: Working Notes for CLEF 2014 Conference, Sheffield, UK, September 15-18, 2014. pp. 877–897 (2014)
[53] Stein, B., Lipka, N., Prettenhofer, P.: Intrinsic plagiarism analysis. Language Resources and Evaluation 45(1), 63–82 (2011)
[54] Tschuggnall, M., Specht, G.: Countering Plagiarism by Exposing Irregularities in Authors' Grammar. In: Proceedings of the European Intelligence and Security Informatics Conference (EISIC). pp. 15–22. IEEE, Uppsala, Sweden (August 2013)
[55] Tschuggnall, M., Specht, G.: Detecting Plagiarism in Text Documents Through Grammar-Analysis of Authors. In: Proceedings of the 15th Fachtagung des GI-Fachbereichs Datenbanksysteme für Business, Technologie und Web (BTW). pp. 241–259. LNI, GI, Magdeburg, Germany (March 2013)
[56] Tschuggnall, M., Specht, G.: Using Grammar-Profiles to Intrinsically Expose Plagiarism in Text Documents. In: Proc. of the 18th Conf. of Natural Language Processing and Information Systems (NLDB). pp. 297–302 (2013)

[57] Tschuggnall, M., Specht, G.: Automatic decomposition of multi-author documents using grammar analysis. In: Proceedings of the 26th GI-Workshop on Grundlagen von Datenbanken. CEUR-WS, Bozen, Italy (October 2014)
[58] Vartapetiance, A., Gillam, L.: A Big Increase in Known Unknowns: from Author Verification to Author Clustering. In: CLEF 2016 Working Notes. CEUR Workshop Proceedings, CLEF and CEUR-WS.org (2016)
[59] Verhoeven, B., Daelemans, W.: Clips stylometry investigation (csi) corpus: A Dutch corpus for the detection of age, gender, personality, sentiment and deception in text. In: Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014). Reykjavik, Iceland (2014)
[60] Zheng, R., Li, J., Chen, H., Huang, Z.: A framework for authorship identification of online messages: Writing-style features and classification techniques. Journal of the American Society for Information Science and Technology 57(3), 378–393 (2006)
[61] Zmiycharov, V., Alexandrov, D., Georgiev, H., Nakov, P., Kiprov, Y., Georgiev, G., Koychev, I.: Authorship-link Ranking and Complete Author Clustering. In: CLEF 2016 Working Notes. CEUR Workshop Proceedings, CLEF and CEUR-WS.org (2016)