Register & Genre Seminar: Towards Intrinsic Plagiarism Detection

Oren Halvani

Institut für Sprach- und Literaturwissenschaft,

Technische Universität Darmstadt,

64289, Darmstadt

Oren@Halvani.de

Abstract

Thanks to the internet the amount of information nowadays has grown drastically worldwide during the last two decades. According to [14]: "The Amount of Digital Information Reached 281 Exabytes (281 Billion Gigabytes)." Furthermore it can be observed that a huge percentage of all these information are represented through text, e.g. websites, e-Mails, office documents, e-books, etc. As a logical consequence one can infer that many distinct authors have produced these texts. However, very often documents do not contain any explicit information about their authors and therefore determining the true copyright holder manually is very hard.

To cope with this challenge, so-called "Authorship Attribution Methods" have been developped over a number of years with the intention to guess the most likely author for a given text. Unfortunately in some cases it is not possible to apply these methods as, for instance, when there is a lack of reference material (e.g. in the forensics field) or quite the contrary, when there are too many refrences (e.g. in the internet) which makes comparison analyses fairly impossible. For other scenarios there might be no interest in uncovering the true identity of the unknown author. Instead, one is curious about the fact if a given text has been produced by one or several authors.

If a document consists of textual material from at least two authors, where only one is explicit declared and no associated citations have been found then, this act may refer to the term of "plagiarism". The discipline that seeks to recognize plagiarism, when no further material is available besides the given document itself, is called "Intrinsic Plagiarism Detection" and is strongly related to the discipline of authorship verification.

In this paper, I work out the fascinating and highly complex discipline of Intrinsic Plagiarism Detection. I explain the essential foundations and present several state-of-the-art approaches, that have been proposed by the most important researchers in this field.

Keywords: intrinsic plagiarism detection, authorship verification, outlier detection, register, genre, style, stylometry.

1 Introduction

Thanks to the Internet's possibilities the process of information procurement has been improved extremely during the last years. A huge amount of information that is accessable worldwide over the Internet is represented through text. Due to its nature, textual information is easy to use and reuse, where the latter one can be understood as modifying a source text in order to embedd the result into other texts.

Reusing text is a indispensable prerequisite in our everyday life. Website operators for instance, often reuse textual content from other websites and combine it with their own material. As long as the "borrowed" text is correctly cited and copyright issues are strictly abided and respected, the

act of reusing text can be treated as legally. Unfortunately this is not always the case. Very often entire texts or small parts are reused by individuals for their own purposes, without giving credit to the original authors. From the legal point of view, this act refers to the term of plagiarism, a term which has been mentioned recently often in the media (e.g. the Guttenberg affair). Plagiarism is a serious academic offense, which seems to become more and more ordinarily in our everyday life as Stein et al. reveals: *"A recent large-scale study on 18,000 students by McCabe reveals that about 50% of the students admit to plagiarize from extraneous documents"* [35].

But what do we mean by the term of plagiarism? Although its wide variety of possible definitions, I believe that the following definition might best fit in terms of text plagiarism: *"Plagiarism is the act, intentional or otherwise, of copying or borrowing words or ideas without properly acknowledging the original source"* [23].

According to Ben-Dror et al. [10] plagiarism can be splitted up into *"five levels, or degrees, of misconduct, ranging from the most serious (Level One) to the least serious (Level Five)*":

- *"**Level One:** The uncredited verbatim copying of a full paper, or the verbatim copying of a major portion (greater than half of the original paper)"*. [10]
- *"**Level Two:** The uncredited verbatim copying of a large portion (less than half of the original paper)"*. [10]
- *"**Level Three:** The uncredited verbatim copying of individual elements (e.g., paragraphs, sentences, figures)"*. [10]
- *"**Level Four:** The uncredited improper paraphrasing of pages or paragraphs"*. [10]
- *"**Level Five:** The credited verbatim copying of a major portion of a paper without clear delineation (e.g., quotes or indents)"*. [10]

Note: due to a strong influence by [25] and [30], this paper is mainly focusing on "Level Two" and "Level Three".

A lot of effort has been done so far in order to fight plagiarism. In former times, when computers where not available, each single plagiarism case was examined individually by responsible experts. But due to the fact that manual plagiarism detection is highly time-consuming, current research projects attempts to automate the detection process. Typically one can distinguish between two areas of plagiarism detection:

- **External Plagiarism Detection (EPD):** *"External plagiarism detection deals with the problem of finding plagiarized passages in a suspicious document based on a reference corpus"*. [22]
- **Intrinsic Plagiarism Detection (IPD):** *"Intrinsic plagiarism detection does not use external knowledge and tries to identify discrepancies in style within a suspicious document"*. [22]

While research is performed largely for EPD [29], the much more complicated area of IPD requires more research work to be carried out. There are several reasons that makes IPD difficult to handle (rather than EPD), for instance:

- No reference documents are available besides the given document itself.
- No further possibilies to uncover plagiarism besides detecting suspicious text parts, which significantly differs from the rest of the document. (How can we define "significantly"?)
- Even if suspicious text parts are found, there is still no guarantee that these parts are truely plagiarized.

Despite of its complex nature, IPD becomes increasingly important, due to the fact that on the one hand, reference material is not always available or on the other hand, that the amount of references is too large (e.g. the extremely great number of websites on the internet). To cope with such extreme cases, IPD can serve as a helpful option to assist experts.

The rest of this paper is structured as follows. The next section describes the differences between IPD and Authorship Verification. Section three explains briefly the idea of "text layers" followed by section four, which aims to distinguish between the terms of Register, Genre, Style and Stylometry. After that, section five dives more into the

depths of features, which reflects the core concept of IPD. In section six the term "Similarity metric" is explained, due to its excessive usage within the IPD discipline. Section seven elaborates on some general problems within IPD, while section eight surveys some already existing approaches in that discipline (particularly their technical aspect) and also their applicability on the PAN'09 corpus, which is then explored more in detail in section nine. Finally, the discussion is given in section ten.

2 IPD vs. Authorship Verification

The disciplines of IPD and Authorship Verification are strongly related to each other. This section elaborates briefly the statement.

In [33] the authors characterize the Authorship Verification problem as follows: *"In an authorship verification problem one is given writing examples from an author A, and one is asked to determine whether or not each text in fact was written by A"*. This characterization implies the main idea of IPD, which is, to identify breach of style within the given document. Finding such style discrepancies increases the assumption that the document might contain plagiarized text elements and hence, was probably not written by only one author. In contrast to that, if (reliable) IPD methods have been applied on the same document and were not able to find any suspicious style changings then, this would imply that the writing style is unique and therefore only one author was involved in the generation of the document (rather than several authors).

From the technical view, both disciplines are "one-class classification" problems, where *"a one-class classification problem defines a target class for which a certain number of examples exist"* [34]. In addition to that, the same authors explain that IPD can be described *"as a more general form of the authorship verification problem"* [34].

Nevertheless, there is also (at least) one important difference between both disciplines. In contrast to Authorship Verification, IPD is not addressing the question who actually has written a given document. In other words, the technical background of both might be the same, while the context or more precisely the field of application differs.

3 Text layers

Before introducing the technical aspect of IPD one must first understand an important fact about text and its underlying structure. Text is almost always organized in many so-called "text layers" (or just layers), where each layer has its own specific function. The following table enumerates some of the most important layers:

Grapheme layer:	This is a text.
Symbol layer:	T h i s i s a t e x t .
Token layer:	[This] [is] [a] [text.]
Phoneme layer:	/ðɪs ɪz ə tɛkst/
Part-Of-Speach layer:	This/DT is/VBZ a/DT text/NN ./.
Constituent layer:	(This (is (a (text)))).

Table 1: A few samples of text layers

Note: there are many more layers existing as those listed above, but to simplify matters this study concentrates only on some of these.

4 Register, Genre, Style and Stylometry

In this section I try to distinguish the terms of Register, Genre, Style and Stylometry with respect to the definitions and explanations I have researched during this study.

4.1 Register & Genre

According to Crystal, the term Register *"refers to a variety of language defined according to its use in social situations, e.g. a register of scientific, religious, formal English"* [4], whereas Genre can be understood as *"the generalization of a term well established in artistic and literary criticism for an identifiable category of literary composition (e.g. poetry, detective story)"* [4]. Crystal continues that: *"A genre imposes several identifiable characteristics on a use of language, notably in relation to subject-matter, purpose (e.g. narrative, allegory, satire), textual structure, form of argumentation, and level of formality"*, [4].

From the point of view of IPD, Register and Genre have strictly to be ignored during the analysis step (or more precisely the outlier detection step). The reason for this is, that none of them really reflects an individual writing style and hence, detection methods are doomed to fail, if they are taken into account. This however, raises the crucial question: What is style exactly?

4.2 Style & Stylometry

According to my literature research, there is no absolute criteria for "style", therefore I refer to the term of "stylistics", which Crystal defines as follows: *"A branch of linguistics which studies the features of situationally distinctive uses (varieties) of language, and tries to establish principles capable of accounting for the particular choices made by individual and social groups in their use of language"*, [4].

In order to approximate style reasonably, such that IPD (and other disciplines) can be implemented, a field of study named "Stylometry" has been established that analyzes so-called stylometric features. Simon et al. defines Stylometry as "*a discipline that determines authorship of literary works through the use of statistical analysis and machine learning*" [7].

To date, many variants of stylometric features have been investigated by researchers across the fields of linguistics, forensics, machine learning, etc. Style, as well as Stylometry are extremely important in the context of IPD, since the core aussumption of IPD is, that each individual has its own specific writing style [24] and hence, it is the only possibility to disinguish authors from eachother.

4.3 IPD & Stylometry

Stein et al. [25] explain that the most appropriate stylometric features for the IPD discipline fall in one of the following five categories:

1. *"**Text statistics:** which operate at the character level"*, [25].
2. *"**Syntactic features:** which measure writing style at the sentence-level"*, [25].
3. *"**POS features:** to quantify the use of word classes"*, [25].
4. *"**Closed-class word sets:** to count special words"*, [25].
5. *"**Structural features:** which reflect text organization"*, [25].

According to these findings, each category refers to one specific text layer. In order to give the whole idea a more formalized meaning, one can use a very basic mathematical concept in the following manner. Assume the above categories are represented as sets such, that each set contains a finite number of distinctable features. Then, style can be roughly defined as:

$$
\begin{aligned}
\mathcal{STYLE} := \ & \text{TEXT STATISTICS} \cup \\
& \text{SYNTACTIC FEATURES} \cup \\
& \text{POS FEATURES} \cup \\
& \text{CLOSED-CLASS WORD SETS} \cup \\
& \text{STRUCTURAL FEATURES} \cup \\
& \text{OTHER}
\end{aligned}
$$

where OTHER denotes an underspecified set, which forms a union of all possible (unknown) feature sets. Finding a way to define clearly the OTHER set is absolutely impossible. If one assumes that a precise definition exists then, this would entail that any discipline, which involves discriminant analysis in terms of style (including IPD), would not be an open scientific problem anymore.

5 Features

Features reflect the core concept of IPD. Strictly speaking, if features would not exist, IPD would not be possible to implement.

In order to use features in the IPD task, they first must be extracted of a given document. The literature [29], [27], [32] denotes this process as "feature extraction". Feature extraction can be applied on both, the entire document (globally) or just parts of it (locally). Since "part" is an unclear term for the IPD task, I use the better known term "passages" instead, which already has been termed by the authors in [25].

Passages consists of a number of sentences. Unfortunately, it is very difficult to say how many of them, a single passage should include. The reason for this is that judging, if a sentence is supposed to be a "real" sentence, is a science in its own. As a result, all passages should be chosen of approximately equal size, when segmenting the text. Once the document is segmented into its passages, feature extraction can be then applied on each single passage. This however, raises the question which features can be considered in order to gain the best discrimination power. Stein et al. [34] for instance have compiled the following collection of some well-known and suitable features for the IPD task:

Stylometric feature		Reference
Lexical features (character-based)	Character frequency	[41]
	Character n-gram frequency/ratio	[17], [28], [15], [19]
	Frequency of special characters (’(’, ’&’, ’/’, etc.)	[41]
	Compression rate	[32]
Lexical features (word-based)	Average word length	[12], [41]
	Average sentence length	[12], [41]
	Average number of syllables per word	[12]
	Word frequency	[26], [12], [19]
	Word n-grams frequency/ratio	[28]
	Number of hapax legomena	[37], [41]
	Number of hapax dislegomena	[37], [41]
	Dale-Chall index	[5], [3]
	Flesch Kincaid grade level	[6], [16]
	Gunning Fog index	[9]
	Honore’s R measure	[13], [37], [41]
	Sichel’s S measure	[37], [41]
	Yule’s K measure	[40], [12], [37], [41]
	Type-token ratio	[40], [12], [41]
	Average word frequency class	[42]
Syntactic features	Part-of-speech	[32], [19]
	Part-of-speech n-gram frequency/ratio	[18], [19]
	Frequency of function words	[1], [18], [26], [12], [41], [19]
	Frequency of punctuations	[41]
Structural features	Average paragraph length	[41]
	Indentation	[41]
	Use of greetings and farewells	[32], [41]
	Use of signatures	[32], [41]

Table 2: Compilation of important and well-known features, [34] (**Note:** References have been adjusted)

According to Stein et al. [34] the following five features (out of 30) have found out to be the most discriminative in terms of style:

Stylometric feature	F-Measure
Flesch Reading Ease Score	0.208
Average number of syllables per word	0.205
Frequency of term: of	0.192
Noun-Verb-Noun tri-gram	0.189
Noun-Noun-Verb tri-gram	0.182

Table 3: Feature rankings (adopted from [34])

6 Similarity metrics

Although its excessive usage in IPD, the term "similarity metric" is not always clearly defined in the literature. Therefore I use my own definition within the scope of this study.

- **Informal definition**: A similarity metric is a function which measures how similar two objects (represented as vectors) are.
- **Formal definition**: Let $X, Y \in \mathbb{R}^n$ denote two real-valued vectors. A similarity function can be than defined as follows:

  $sim : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$, and more precisely:

  $sim(X, Y) \mapsto \{q \mid q \in \mathbb{R} \wedge 0 \leq q \leq 1\}$.

The formal definition implies that the resulting value of $sim(X, Y)$ falls in the intervall of $[0; 1]$, where 1 can be interpreted as highly similar and 0 as the opposite.

Similarity metrics are used in almost any conceivable domain e.g. biology, chemistry, computer science, mathematics, physics, psychology, statistics, etc. In the context of IPD, similarity metrics are important (or even essential) in order to detect style discrepancies between text fragments (e.g. sentences or phrases) and the entire document.

According to Lin [21] a similarity measure should fulfill the following intuitions:

1. *"The similarity between A and B is related to their commonality. The more commonality they share, the more similar they are"* [21].
2. *"The similarity between A and B is related to the differences between them. The more differences they have, the less similar they are"* [21].
3. *"The maximum similarity between A and B is reached when A and B are identical, no matter how much commonality they share"* [21].

Note: A refers to X and B refers to Y in the context of my formal definition.

To date, alot of similarity metrics have been proposed. Some of the most popular are listed below:

Cosine Similarity, [2]:

$$\frac{\sum_{i=1}^{n}(x_i y_i)}{\left(\sqrt{\sum_{i=1}^{n} x_i^2}\right)\left(\sqrt{\sum_{i=1}^{n} y_i^2}\right)}$$

Dice Coefficient, [2]:

$$\frac{2\sum_{i=1}^{n}(x_i y_i)}{\left(\sum_{i=1}^{n} x_i^2\right) + \left(\sum_{i=1}^{n} y_i^2\right)}$$

Jaccard Coefficient, [2]:

$$\frac{2\sum_{i=1}^{n}(x_i y_i)}{\left(\sum_{i=1}^{n} x_i^2\right) + \left(\sum_{i=1}^{n} y_i^2\right) - \left(\sum_{i=1}^{n}(x_i y_i)\right)}$$

Czekanowski Index, [2]:

$$\frac{\sum_{i=1}^{n} min(x_i,\ y_i)}{min\left(\sum_{i=1}^{n} x_i\ ,\ \sum_{i=1}^{n} y_i\right)}$$

In the literature some authors use the term "distance functions" as a synonym for similarity metrics. This however, is not correct due to the fact that: *"From the scientific and mathematical point of view, distance is defined as a quantitative degree of how far apart two objects are"* [2], whereas a similarity metric measures how similar two objects are. Nevertheless, distance functions can be transferred into similarity metrics with the following comprehensible formula, [11]:

$$\frac{1}{1 + dist(X, Y)}$$

where $dist(X, Y)$ represents any arbitrary distance function. The resulting value of the fraction always falls in the interval of $[0; 1]$. This can be concluded as follows, let d represent the denominator $1 + dist(X, Y)$ than ,the lower bound of the interval is given by the limit:

$$\lim_{d \to \infty} \frac{1}{d} = 0$$

In contrast, if X and Y are identical or more precisely $dist(X, Y) = 0$, than the fraction results in:

$$\frac{1}{1+0} = 1$$

which is the upper bound of the interval. Hence, similarity metrics and distance functions can be used both for the IPD task.

7 General problems with IPD

Like any other scientific discipline, IPD also suffers from several types of problems. This section elaborates on the most important problems in IPD.

7.1 Decide if plagiarized elements truely exist

An important prerequisite that one has to take into account, before applying IPD methods on a given document, is to decide (in some way) if it contains any plagiarized text elements at all. Stamatatos [30] believes that two options can be considered, in order to to decide whether a document contains plagiarized text elements or not:

- *"**By pre-processing:** A criterion must be defined to indicate a plagiarism-free document. If this is the case, there is no further detection of plagiarized sections"* [30].

- *"**By post-processing:** The algorithm detects any likely plagiarized sections and then a decision is taken based on these results"* [30].

For the former option Stamatatos conclude to use a criterion that is based on the variance of the so-called "style change function" (details about this can be looked up in the section "Existing Approaches"). This function behaves stable if the document is written by only one author. To the contrary, if a document is potentially written by more than one author (which means it is partly plagiarized), then, *"the style change function will be characterized by peaks that significantly deviate from the average value"* [30]. The presence of such peaks are indicated by the standard deviation. Stamatatos judges that a document is considered to be plagiarism-free, if the standard deviation is lower than a predefined threshold. One possible value for that threshold was determined empirically at 0.02 [30].

7.2 Document & passages length

Another very important and a still opened question, which has to be answered, is the fact how long specific text elements (sliding blocks/windows, sentences, paragraphs or the document itself) should be, in order to achieve practical results within the IPD task. During the examination of the literature, no definitive requirements have been spotted so far.

Nevertheless, I have noted several settings and suggestions as for instance by Granitzer et al. [22]. The authors gained quite well results with respect to a sentence window size of 2000 characters (evaluated on the PAN'09 corpus). In terms of paragraph lengths, Suarez et al. [36] used 200 characters for their system that was also trained on the same corpus. Stein et al. reported a size of 40 - 200 words of one passage, *"which is ambitious from the analysis standpoint—but which corresponds to realistic situations"* [25]. With regard to the documents length Stein et al. mentioned: *"Experience shows that a style analysis becomes statistically unreliable for text lengths below 250 words"* [34].

Conclusion: unsurprisingly, general rules cannot be defined for lengths of text elements, but one fact always remains the same, the longer the length of a document is, the more stylometric features can be captured and hence, a better stylistic model can be constructed. The better such a modell is, the better stylistic outliers can be detected.

8 Existing Approaches

In this section I introduce some of the already existing approaches in the field of IPD.

8.1 Vector Space Models

In [22] Granitzer et al. propose their so-called Vector Space Models approach, which is composed of the following three stages:

1. *"'Vectorization of each sentence in the suspicious document"'*, [22].

2. *"'Determination of outlier sentences based on the document's mean vector"'*, [22].

3. *"'Post processing of the detected outlier sentences"'*, [22].

In 1.) the authors chose a window of k sentences around the suspicious sentence, which is surrounding of $\frac{k}{2}$ sentences. Next, they construct for each sentence S_i a vector for each stylometric feature space. The feature space is consisting of the following feature categories, which were presented in [8] and [25]:

- **Average word frequency class**
- **Punctuation**
- **Part of speech tags**
- **Pronouns**
- **Closed class words**

With regard to these five categories, the resulting five vectors of each sentence S_i are than concatenated into a single vector, which is denoted by $\vec{Q_i}$. After that, they build the mean vector $\vec{m}$ of all the $\vec{Q_i}$ vectors as follows, [22]:

$$\vec{m} = \frac{1}{n} \sum_{i=1}^{n} \vec{Q_i}$$

In 2.) the authors use an outlier detection scheme (based on the cosine similarity metric), which tries to find those sentences (represented through their feature vectors $\vec{Q_i}$) that deviate significantly from the mean vector $\vec{m}$.

The authors state that a j'th sentence is marked as an outlier, if the following inequality holds, [22]:

$$cos(\vec{v_j}, \vec{m}) < mean - \epsilon * stddev$$

where $\vec{v_j}$ is a (plagiarized) sentence vector and ϵ is a small constant ≥ 1. Both $mean$ and $stddev$ are formalized as follows, [22]:

$$mean = \frac{1}{n} \sum_{i=1}^{n} cos(\vec{Q_i}, \vec{m})$$

$$stddev = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (cos(\vec{Q_i}, \vec{m}) - mean)^2}$$

In 3.) the authors derive the final blocks of plagiarized passages, which are based on those sentences that deviate to much from the mean.

Granitzer et al. evaluated their system on both, the development corpus and the competition corpus of the PAN'09 plagiarism detection competition, where they report the following results, with regard to the F_1-Measure:

- **Development Corpus:** 0.4603 ,[22]
- **Competition Corpus:** 0.2286 ,[22]

Their system have taken the 3rd out of 4 places in the IPD task within the PAN'09 competition. For future work the authors conclude to improve their outlier detection method and also to investigate more stylometric features for the task.

8.2 Character n-gram Profiles

In [30] Stamatatos proposed a method that was based on character n-gram profiles (CNP). CNP represents the set of different character n-grams, encountered in document and their normalized frequencies. The main idea of his approach *"is to define a sliding window over the text length and compare the text in the window with the whole document"*, [30]. The following illustration underlines the idea:

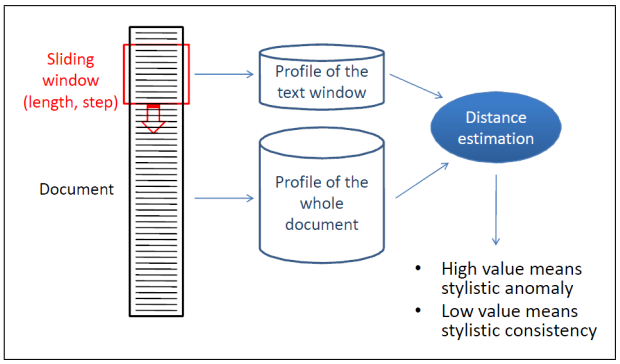

Figure 1: Stamatatos' CNP approach, [31]

Instead of using a similarity metric, Stamatatos applied the following dissimilarity meassure with the intention to detect stylistic irregularities within the document, [30]:

$$nd_1(A, B) = \frac{\sum_{g \in P(A)} \left(\frac{2\left(f_A(g) - f_B(g)\right)}{f_A(g) + f_B(g)} \right)^2}{4|P(A)|}$$

The elements of the formula are described as follows, A, B are two texts, where $P(A)$ denotes the profile of A. The more precisely, $P(A)$ is actually the vector of normalized frequencies of all the character n-grams that appear at least once in

the text. $|P(A)|$ represents the size of the profile (number of all the character n-grams). Furthermore, $f_A(g)$ and $f_B(g)$ denote the frequency of occurrence (normalized over the text length) of the n-gram g in A and B. Another interesting observation is, that the resulting value of the above function always falls between 0 and 1 (although it is not a similarity metric), this is ensured by the denominator $4|P(A)|$.

The style change function, denoted by $sc(i, D)$ parameterizes the above dissimilarity function as follows: $nd_1(w_i, D)$ for $i = 1, \ldots, |w|$, where w stands for the sliding window of the length l and $|w|$ represents the total amount of all the windows in the document D. Let x be the number of all characters in D and s the step of the sliding window (in characters), than the amount of all the documment windows can be computed as, [30]:

$$|w| = \lfloor 1 + \frac{x - l}{s} \rfloor$$

As well as Granitzer et al. [22], Stamatatos has also participated in the PAN'09 competition, where he held the first place in the IPD task. Stamatatos' system achieved the following results (measured against the F_1-Measure):

- **Development Corpus:** 0.2876 ,[30]
- **Competition Corpus:** 0.3086 ,[30]

8.3 Kolmogorov Complexity Measures

In their work [29] Seaward et al. follow an interesting idea. They use the Kolmogorov Complexity measures in order to extract structural information from a given document. The authors are not treating the document as a bag-of-words (which is usually sufficient for the IPD task), due to the fact that this model ignores the information about the text structure, as the following example illustrates:

"This is a nice sentence"

$\Downarrow$

{ is , sentence , a , This , nice }

Like the n-gram approach of Stamatatos [30], the Kolmogorov complexity features preserve the structure of style features within the text. These features can be than used by other IPD methods in order to increase the detection quality of plagiarized text elements. But before such features can be gained for further analysis, some preliminary work has to be done beforehand. Seaward and Matwin therefore analyze each text segment with regard to its word class distribution. The following sample sentence (taken from [29]) illustrates the basic idea behind the distribution of word classes, with respect to its corresponding binary string representation:

Sentence:	"Billy	walked	the	dog	yesterday"
Noun distrib.	1	0	0	1	0
Verb distrib.	0	1	0	0	0
Stopwords distrib.	0	0	1	0	0

Once a sentence has been decomposed into its word class distribution, one can quantify the structure, in order to measure its randomness/complexity. One possible method of doing this is the so-called Kolmogorov complexity measures. But what are these measures exactly?

The authors state that *"Kolmogorov complexity measures the informativeness of a given string (...) as the length of the algorithm required to describe/generate that string"* [15]. In other words, the Kolmogorov complexity meassures the randomness of the string (according to its binary representation). *"Unfortunately, Kolmogorov complexity is formally uncomputable, in a strict technical sense related to the Halting Problem"* [15]. However, it is still possible to approximate it with lossless compression algorithms, as for example the Lempel-Ziv compression algorithm (Zlib), which was used during the experiments in [20]. The authors define the approximate Kolmogorov complexity of a string x, using C as a compression algorithm, denoted by $K_c(x)$ as follows, [29]:

$$K_c(x) = \frac{Length(C(x))}{Length(x)} + q$$

where q represents the length in bits of the program that implements the compression algorithm C and $C(x)$ is the result of compressing x using C.

Note: *"In practice, q is usually ignored as it is not useful in comparing complexity approximations and it varies according to which programming language implements C"* [29]. In order to

answer the question how the resulting value of the function above can be interpretated, the authors explain that if C was not able to compress the string x very much, than $K_c(x)$ is high and thus x has a high complexity (and vice versa).

The experiment results of Seaward et al. revealed that complexity features are able to outperform normalized count features (e.g. word-, sentence-, POS-counts, vocabulary richness, etc.) as one can infer from the following table:

Rank	Feature
1	Adjective complexity
2	Adjective count
3	Topic word complexity
4	Verb word complexity
5	Passive word complexity
6	Active word complexity
7	Preposition count
8	Stop word count
9	Avg. word length per sentence
10	Topic word complexity

Table 4: Top 10 features, calculated by a χ^2 feature evaluator (adopted from[29])

The authors used a Support Vector Machine (SVM) and a Neural Network as classifiers for their IPD experiment, where the latter one showed the most improvement with complexity measures (F_1-Measure of 0.603 against 0.587 via the SVM classifier). However, the researchers conclude that: *"More research needs to be done in using compression models which have prior knowledge of the language to be analyzed and/or the prior probabilities of word classes. This would result in more meaningful complexity features which would likely aid in the difficult task of intrinsic plagiarism detection"* [29].

9 Corpora for IPD

In former times, where only a little research has been conducted on IPD, researchers were forced to build individual test corpora, in order to run their evaluations. Fortunately, this has changend thanks to the "1st International Competition on Plagiarism Detection", which was held in cooperation of the researchers from Bauhaus University Weimar and Universidad Politécnica de Valencia.

The researchers compiled a large-scale corpus for the evaluation of automatic plagiarism detection algorithm, in both disciplines, IPD and EPD. In this section I summarize the relevant facts about the corpus and its parameters.

9.1 PAN'09

The PAN'09 corpus was compiled in 2009 by Potthast, Eiselt, Stein, Barrón-Cedeño, and Rosso [39] and made publicly available for researchers in the field of IPD and especially EPD.

The corpus consists of German, Spanish and English documents, where the latter one reflects the majority. Within these documents *"all types of plagiarism cases can be found, namely monolingual plagiarism with varying degrees of obfuscation, and translation plagiarism from Spanish or German source documents"*, [38].

9.2 Plagiarism Obfuscation Strategies

As mentioned above, the PAN'09 corpus contains plagiarism cases with varying degrees of obfuscation. In this subsection I briefly point out the ofuscation strategies that have been used in the corpus, in order to make the plagiarized sections more difficult to detect.

Aaccording to [39]: *"The random plagiarist employs random combinations of the following strategies, and each strategy with varying strength"*:

- **Paraphrasing:** *"Given a sequence of tokens from a passage of text, each token is replaced by one of its synonyms, antonyms, hyponyms, or hypernyms, chosen at random. If neither are available for a given token the token is retained"*, [39].

- **Parts-of-Speech Reordering:** *"Given a sequence of tokens from a passage of text, its sequence of parts of speech is determined. Then the tokens from the text are reordered at random while their original sequence of parts of speech is maintained"*, [39].

- **Random Text Operations:** *"Given a sequence of tokens from a passage of text, words or short phrases are shuffled, removed, inserted, or replaced at random until a halting criterion is reached. Insertions and replacements may come from the new context in which the obfuscated passage will be inserted, or from other sources"*, [39].

9.3 Corpus Statistics

In this subsection some important statistics in regard to the PAN'09 corpus are given.

- *"**Corpus size:** 20 611 suspicious documents, 20 612 source documents"*, [39]
- *"**Document lengths:** small (up to paper size), medium, large (up to book size)"*, [39]
- *"**Plagiarism contamination per document:** 0%-100% (higher fractions with lower probabilities)"*, [39]
- *"**Plagiarized passage length:** short (few sentences), medium, long (many pages)"*, [39]
- *"**Plagiarism types:** monolingual (obfuscation degrees none, low, and high), and multilingual (automatic translation)"*, [39]

9.4 Performance Measures

The plagiarism detection systems have been measured according to their: precision, recall, and granularity on detecting the plagiarized passages in the corpus. The formalization of these terms are given as follows: s denotes a plagiarized passage from the set of all plagiarized passages S. r denotes a detection from the set R of all detections. S_R represents a subset of S for which detections exist in R. $|s|$, $|r|$ denote the char lengths of s, r and $|S|$, $|R|$, $|S_R|$ are the sizes of the respective sets, [38]. With these notation the required formulas for the measuring task can be defined as:

Recall, [38]:

$$\frac{1}{|S|}\sum_{i=1}^{|S|}\left(\frac{\#(detected\ chars\ of\ s_i)}{|s_i|}\right)$$

Precision, [38]:

$$\frac{1}{|R|}\sum_{i=1}^{|R|}\left(\frac{\#(plagiarized\ chars\ of\ r_i)}{|r_i|}\right)$$

F_1-Measure, [38]:

$$\frac{2PrecisionRecall}{Precision + Recall}$$

Granularity, [38]:

$$\frac{1}{|S_R|}\sum_{i=1}^{|S_R|}\left(\#(detections\ of\ s_i\ in\ R)\right)$$

Overall, [38]:

$$\frac{F_1 - Measure}{log_2(1 + Granularity)}$$

The following table shows the results for the PAN'09 competition:

Rank	Overall score	F_1-Measure	Precision	Recall	Granularity	Participant
1	0.2462	0.3086	0.2321	0.4607	1.3839	E. Stamatatos, University of the Aegean, Greece
2	0.1955	0.1956	0.1091	0.9437	1.0007	B. Hagbi and M. Koppel, Bar Ilan University, Israel
3	0.1766	0.2286	0.1968	0.2724	1.4524	M. Granitzer, M. Muhr, M. Zechner, and R. Kern, Know-Center Graz, Austria
4	0.1219	0.1750	0.1036	0.5630	1.7049	L. M. Seaward and S. Matwin, University of Ottawa, Canada

Table 5: Performance results for the IPD task (adopted with slight formatting modifications, [39])

10 Discussion

In this study I have tried to bring light into the darkness of the relatively young IPD discipline (namely "Intrinsic Plagiarism Detection").

In comparision to EPD (namely "Extrinsic Plagiarism Detection") IPD is comparatively a poorly investigated area. Despite of the fact that several promising approaches exist, there is a serious need for further research to be carried out in this field, in order to build reliable systems for real-world scenarios. However, IPD is a very fascinating discipline, which attracts considerable attention in current research.

Beyond the scope of plagiarism, IPD algorithms can be used in other related research fields. For instance, it might be conceivable to use IPD in a preliminary stage of authorship attribution, how exactly? In order to perform authorship attribution, one needs a training set from already attributed documents. Each one of these documents must be written by only one author, otherwise the learned attribution model is useless. In practice however, it cannot be always guaranteed that these documents are clearly produced by only one author (e.g. when a the training set consists of forum posts or something similar). Hence, IPD can be used here to decide, if the writing style within the documents is consistent enough before proceeding with the authorship attribution process.

The PAN'11 competition, which will held at the end of September 2011 in in Amsterdam, Netherlands, will hopefully show promising improvements of the presented approaches or even new ideas and concepts how to deal with IPD.

References

[1] Shlomo Argamon and Marin Saric. Style mining of electronic messages for multiple authorship discrimination: First results, 2003.
Cited on: 5.

[2] Sung-Hyuk Cha. Comprehensive survey on distance/similarity measures between probability density functions, 2007.
Cited on: 6.

[3] Jeanne Chall and Edgar Dale. *Readability Revisited: The New Dale-Chall Readability Formula*. Brookline Books, Brookline, Massachusetts, 1995.
Cited on: 5.

[4] David Crystal. *A dictionary of linguistics and phonetics / David Crystal*. B. Blackwell, Oxford, UK ; Cambridge, Mass., USA :, 3rd ed. edition, 1991.
Cited on: 3 and 4.

[5] E. Dale and J.S. Chall. *A formula for predicting readability*. Bureau of Educational Research, Ohio State University, 1948.
Cited on: 5.

[6] Rudolph Flesch. A new readability yardstick. *Journal of Applied Psychology*, 32(3):221–233, 1948.
Cited on: 5.

[7] Ramesh Iyer Jayendra Tailor Dr. Sandra Westcott Gregory Shalhoub, Robin Simon. Stylometry system – use cases and feasibility study. In *Proceedings of Student-Faculty Research Day, CSIS, Pace University, May 7th, 2010*. Seidenberg School of CSIS, Pace University, White Plains, NY 10606, USA, 2010.
Cited on: 4.

[8] J Grieve. Quantitative authorship attribution: An evaluation of techniques. *Literary and Linguistic Computing*, 22(3):251–270, 2007.
Cited on: 8.

[9] R. Gunning. *The Technique of Clear Writing*. McGraw-Hill, New York, 1952.
Cited on: 5.

[10] Yaakov HaCohen-Kerner, Aharon Tayeb, and Natan Ben-Dror. Detection of simple plagiarism in computer science papers. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 421–429, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.
Cited on: 2.

[11] Gerhard Heyer, Uwe Quasthoff, and Thomas Wittig. *Text Mining: Wissensrohstoff Text – Konzepte, Algorithmen, Ergebnisse*. W3L-Verlag, 2008.
Cited on: 6.

[12] David I. Holmes. The evolution of stylometry in humanities scholarship. *Literary and Linguistic Computing*, 13(3):111–117, September 1998.
Cited on: 5.

[13] A Honoré. Some simple measures of richness of vocabulary. *Association for Literary and Linguistc Computing Bulletin*, 7(2):172–177, 1979.
Cited on: 5.

[14] Infoniac. The amount of digital information. http://www.infoniac.com/hi-tech/amount-digital-information-reached-281-exabytes.html, 2008.
Cited on: 1.

[15] Patrick Juola. Authorship attribution. *Foundations and Trends ® in Information Retrieval*, 1:233–334, December 2006.
Cited on: 5 and 9.

[16] J. Peter Kincaid, Robert P. Fishburne, Richard L. Rogers, and Brad S. Chissom. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Technical report, February 1975.
Cited on: 5.

[17] Bradley Kjell, W. Addison Woods, and Ophir Frieder. Discrimination of authorship using visualization. *Inf. Process. Manage.*, 30(1):141–150, 1994.
Cited on: 5.

[18] Moshe Koppel and Jonathan Schler. Exploiting stylistic idiosyncrasies for authorship attribution. In *In IJCAI'03 Workshop on Computational Approaches to Style Analysis and Synthesis*, pages 69–72, 2003.
Cited on: 5.

[19] Moshe Koppel, Jonathan Schler, and Shlomo Argamon. Computational methods in authorship attribution. *J. Am. Soc. Inf. Sci. Technol.*, 60:9–26, January 2009.
Cited on: 5.

[20] Ming Li and Paul M.B. Vitnyi. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer Publishing Company, Incorporated, 3 edition, 2008.
Cited on: 9.

[21] Dekang Lin. An information-theoretic definition of similarity. In *Proc. 15th International Conf. on Machine Learning*, pages 296–304. Morgan Kaufmann, San Francisco, CA, 1998.
Cited on: 6.

[22] Roman Kern Michael Granitzer Markus Muhr, Mario Zechner. External and Intrinsic Plagiarism Detection using Vector Space Models. In Efstathios Stamatatos Moshe Koppel Eneko Agirre Benno Stein, Paolo Rosso, editor, *3rd Pan workshop, uncovering plagiarism, authorship and social software misuse. 25th annual conference of the spanish society for Natural Language Processing, Sepln 2009*, Lecture Notes in Computer Science, pages 47–55. Springer, 2009.
Cited on: 2, 7, 8, and 9.

[23] Dr. Heather McDermid. Genet 418/518 - human genetics. winter 2010.
Cited on: 2.

[24] G.R. McMenamin and D. Choi. *Forensic Linguistics: Advances in Forensic Stylistics*. CRC Press, 2002.
Cited on: 4.

[25] Sven Meyer zu Eißen and Benno Stein. Intrinsic Plagiarism Detection. In Mounia Lalmas, Andy MacFarlane, Stefan Rüger, Anastasios Tombros, Theodora Tsikrika, and Alexei Yavlinsky, editors, *Advances in Information Retrieval: Proceedings of the 28th European Conference on IR Research (ECIR 06)*, volume 3936 LNCS of *Lecture Notes in Computer Science*, pages 565–569, Berlin Heidelberg New York, 2006. Springer.
Cited on: 2, 4, 7, and 8.

[26] F Mosteller and D L Wallace. *Inference and Disputed Authorship: The Federalist*. Addison-Wesley, 1964.
Cited on: 5.

[27] Naomie Salim Salha Alzahrani and Ajith Abraham. Understanding plagiarism linguistic patterns, textual features and detection methods. 2011.
Cited on: 4.

[28] Conrad Sanderson, , Conrad S, and Simon Guenter. On authorship attribution via markov chains and sequence kernels, 2006.
Cited on: 5.

[29] Leanne Seaward and Stan Matwin. *Intrin-*

sic Plagiarism Detection Using Complexity Analysis, pages 56–61. 2009.
Cited on: 2, 4, 9, and 10.

[30] Efstathios Stamatatos. *Intrinsic Plagiarism Detection Using Character n-gram Profiles*, volume 2, pages 38–46. 2009.
Cited on: 2, 7, 8, and 9.

[31] Efstathios Stamatatos. Intrinsic plagiarism detection using character n-gram profiles (talk). 2009.
Cited on: 8.

[32] Efstathios Stamatatos. A survey of modern authorship attribution methods. *JASIST*, 60(3):538–556, 2009.
Cited on: 4 and 5.

[33] Benno Stein, Nedim Lipka, and Sven Meyer zu Eissen. Meta analysis within authorship verification. In *Proceedings of the 2008 19th International Conference on Database and Expert Systems Application*, pages 34–39, Washington, DC, USA, 2008. IEEE Computer Society.
Cited on: 3.

[34] Benno Stein, Nedim Lipka, and Peter Prettenhofer. Intrinsic plagiarism analysis. *Language Resources and Evaluation*, 45(1):63–82, 2011.
Cited on: 3, 4, 5, and 7.

[35] Benno Stein, Benno Stein, and Marion Kulig. Plagiarism detection without reference collections. *Society*, pages 359–366, 2007.
Cited on: 2.

[36] P. Suárez, J. C. González-Cristóbal, and J. Villena-Román. A plagiarism detector for intrinsic plagiarism. In *Proceedings of the Workshop Uncovering Plagiarism, Authorship, and Social Software Misuse (PAN) at CLEF2010*, 2010.
Cited on: 7.

[37] Fiona J. Tweedie and R. Harald Baayen. How variable may a constant be? measures of lexical richness in perspective. *Computers and the Humanities*, 32(5):323–352, 1998.
Cited on: 5.

[38] Webis at Bauhaus-Universität Weimar and NLEL at Universidad Politécnica de Valencia. 1st International Competition on Plagiarism Detection. http://www.uni-weimar.de/medien/webis/research/events/pan-09/competition.html, 2009. Martin Potthast, Andreas Eiselt, Benno Stein, Alberto Barrón-Cedeño, and Paolo Rosso (editors).
Cited on: 10 and 11.

[39] Webis at Bauhaus-Universität Weimar and NLEL at Universidad Politécnica de Valencia. PAN Plagiarism Corpus 2009 (PAN-PC-09). http://www.webis.de/research/corpora, 2009. Martin Potthast, Andreas Eiselt, Benno Stein, Alberto Barrón-Cedeño, and Paolo Rosso (editors).
Cited on: 10 and 11.

[40] George Udny Yule. *The statistical study of literary vocabulary / by G. Udny Yule*. University Press, Cambridge :, 1944.
Cited on: 5.

[41] Rong Zheng, Jiexun Li, Hsinchun Chen, and Zan Huang. A framework for authorship identification of online messages: Writing-style features and classification techniques. *J. Am. Soc. Inf. Sci. Technol.*, 57:378–393, February 2006.
Cited on: 5.

[42] Sven Meyer zu Eissen and Benno Stein. Genre classification of web pages. In *KI*, pages 256–269, 2004.
Cited on: 5.