ΕΘΝΙΚΟ ΚΑΙ
ΚΑΠΟΔΙΣΤΡΙΑΚΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΦΙΛΟΣΟΦΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΦΙΛΟΛΟΓΙΑΣ
ΤΟΜΕΑΣ ΓΛΩΣΣΟΛΟΓΙΑΣ

ΙΝΣΤΙΤΟΥΤΟ
ΕΠΕΞΕΡΓΑΣΙΑΣ ΤΟΥ
ΛΟΓΟΥ ΕΡΕΥΝΗΤΙΚΟ
ΚΕΝΤΡΟ «ΑΘΗΝΑ»

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ
ΠΟΛΥΤΕΧΝΕΙΟ ΣΧΟΛΗ
ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΣΗΜΑΤΩΝ ΕΛΕΓΧΟΥ
ΚΑΙ ΡΟΜΠΟΤΙΚΗΣ

Technoglossia VIII

Interdisciplinary Interuniversity Postgraduate Course in Language Technologies

Master Thesis

Authorship Attribution Forensics: Feature selection methods in authorship identification using a small e-mail dataset.

Korasidi Andriana Maria

First Supervisor: G.K.MIKROS,Ph.D. Professor of Computational and Quantitative Linguistics

Co-supervisors: George Markopoulos- Dionysis Goutsos , National and Kapodistrian University of Athens,

Athens, 2016

AUTHORSHIP ATTRIBUTION FORENSICS: FEATURE SELECTION METHODS IN AUTHORSHIP IDENTIFICATION USING A SMALL E-MAIL DATASET

"Anytime a linguist leaves the group the recognition rate goes up."

Fred Jelinek (1988)

Abstract

In the present study we confront the problem of identifying an author of anonymous e-mails, a type of communication that conceals inherent security risks and malicious tendencies. A digital forensic investigator needs to determine the authorship of the e-mail by using processes and methods that have not been standardized in the e-mail forensic field. Additionally, through this examination, it is necessary to recover all the legally admissible evidence to be presented in a court of law.

The problem of identifying the most appropriate author from group of potential suspects, who are considered as classes, is a typical authorship identification and classification problem. The challenging part is that e-mails do not have a considerable amount of content and therefore, identification is much harder and complicated than other documents.

Lexical and character features have been proved efficient for small data sets, whereas syntactic features deal better with large data sets. Thus, in the present study we make an effort to examine and compare two of the applied methods used in digital forensics for authorship attribution and partially try to expand the research model by extracting the most important features determining a person's writing style. Therefore we create an n-gram baseline and at first apply a Support Vector Machine to the problem. Secondly we use a well-known method of data mining, the frequent pattern mining technique, in order to extract the specific writing footprint of an author .

Acknowledgments

First of all, I would like to express my gratitude to my first supervisor professor George Mikros whose expertise, understanding, patience and motivation enriched considerably my graduate experience. Without his guidance this research project would never have materialized.

I would also like to acknowledge my professor Yianis Maistros for giving me precious advice and trusting my capabilities from the beginning. His insightful comments and constructive criticism were always thought-provoking and useful at the different stages of my academic attendance.

I ought to extend my gratitude to the National and Kapodistrian University of Athens and the National Technical University of Athens for the support they have provided through the duration of my academic career . My fellow students also deserve special mention for all these exchanges of knowledge and skills , but mainly for venting of frustration during my graduate program.

Moreover, I would like to express my thankfulness to my family for their unconditional support and fully trust that makes it possible for me to strive for my own dreams. Finally, a very special thanks goes to my beloved friend and partner John Bitros, who provided me with encouragement, direction and technical support. I doubt that I will ever be able to convey my appreciation fully, but I owe him my eternal gratitude.

Contents

Chapter 1. Introduction

1.1. Authorship analysis

Authorship attribution has been crucial in identifying authors of texts and has been very successful for literary and conventional writings. Specially, stylometric features have been extensively used for long time. This line of research is called stylometry and it consists of the analysis of linguistic styles and writing characteristics of the authors for identification, characterization, or verification purposes. It is based on the fact that the writing style is an unconscious habit and furthermore it varies from one author to another in the way he/she uses words and grammar to express an idea. Stylometry is therefore the study of the unique linguistic styles and writing behaviors of individuals in order to determine authorship. A person's writing pattern contains many features that reveal an individual uniqueness and identity.

The investigation of authorship attribution has existed for centuries and studies into the authorship of famous literature works and have been conducted with the assumption that the identity of the author can be determined based upon his/her unique style features (Holmes, 1998). The origins of stylometry date back in the 1700's ,when Edmond Malone questioned whether or not Shakespeare really wrote some of the plays bearing his name(Malone, 1700). In 1851 the English logician Augustus de Morgan suggested in a

letter to a friend that questions of authorship might be settled by determining if one text "does not deal in longer words" than another (de Morgan,1882). After that, in year 1887, Thomas Mendenhall proposed that an author has a "characteristic curve of composition" determined by how an author uses words of different lengths frequently (Mendenhall, 1887). A year later, William Benjamin Smith who was a mathematician, published two papers describing a "curve of style" to distinguish authorial styles based on average sentence lengths (C.Mascol, 1888). In 1893 a professor of English, Lucius Sherman, discovered that writing style over time changes with average sentence length (L.A. Sherman, 1893).

These attempts were followed by statistical studies in the first half of the 20th century by British statistician, Yule (Yule, 1938) who was interested in the statistical characteristics of prose style with particular reference to questions of disputed authorship. His earlier work concerned sentence length, but he later turned to noun frequency (Yule,1942). A few years earlier American linguist, Zipf had popularized the notion of "regularity in the distribution of sizes" by first studying word frequencies (Zipf,1932). Later, one of the most influential work in authorship attribution was accomplished, when Mollester and Wallace (Mollester & Wallace, 1964) studied elaborately the authorship of "The Federalist Papers" (a collection of 85 articles and essays written by Alexander Hamilton, James Madison, and John Jay, promoting the ratification of the United States Constitution, among which 12 were claimed by both Hamilton and Madison). Their method was based on Bayesian statistical analysis of the frequencies of a small set of common words. Since then, studies on authorship attribution were dominated by attempts to define features for quantifying writing style, a type of research which we previously called stylometry.

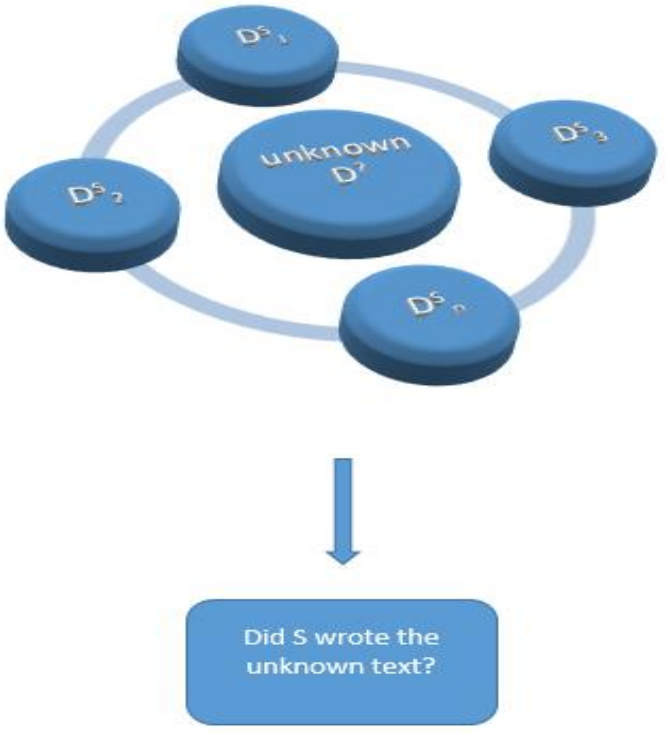
As we've noticed, authorship analysis for resolving disputes over literature has a long history in academic research (Burrows, 1897). However, the fact that each author has a unique stylistic tendency has been also proved helpful for forensic investigation.

1.1.1 Authorship identification

Authorship identification can be described as identifying the author of a text from anonymous writing , based on the author's past writing records. Authorsip identification has been previously researched as a text classification problem in which the authors are each considered classes. In every authorship identification problem, there is a set of candidate authors, a set of text samples of known authorship covering all the candidate authors (training corpus), and a set of text samples of unknown authorship (test corpus), each one of them should be attributed to a candidate author (Efstathios Stamatatos,2009). The rationale of authorship identification is shown below:

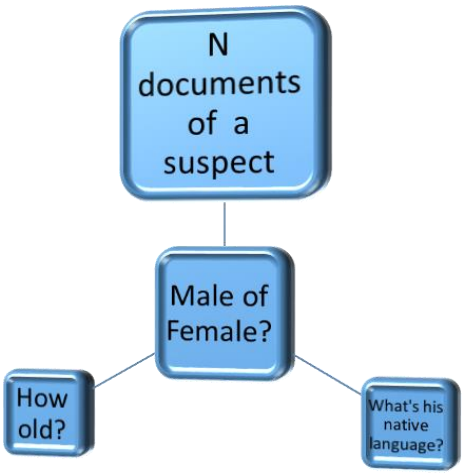
1.1.2. Authorship Verification

A more challenging problem is author verification where given a set of documents written by a single author and the questioned documents, the task is to assess whether the text in dispute was written by the same author of the known texts. The underlying rationale of authorship verification is that the same author has some writing styles that are believed to be difficult to camouflage in a short period of time, and therefore based on these writing styles, a document claimed from the same author can be verified. According to Koppel et al. (Koppel et al, 2004) "verification is significantly more difficult than basic attribution and virtually no work has been done on it, outside the framework of plagiarism detection". Previous works on authorship verification focus on general text documents and even though it is proved to be extremely helpful in various criminal cases concerning online documents , it can also be very difficult to achieve because of their relatively short lengths and poor structure or writing.



1.1.3. Author profiling

Authorship profiling or characterization consists of determining the characteristics of the author of the anonymous document and is used to collect demographic and psychological clues that author's written documents can reveal unconsiously . As in Argamon et al. (2008), one can consider the following profile dimensions: author's *gender* (Koppel et al. 2002; Argamon et al. 2003), *age* (Burger and Henderson 2006; Schler et al. 2006), *native language* (Koppel et al. 2005) and *neuroticism level* (Pennebaker & King 1999; Pennebaker, Mehl, & Niederhoffer, 2003). Other authors have also considered dimensions such as the education level (Corney et al. 2002). Profiling is employed in situations where no training set of the potential suspects is available for analysis. In this case we can exploit the sociolinguistic observation that different groups of people speaking or writing in a particular genre and in a particular language use that language differently (cf. Chambers et al. 2004).



Chapter 2: Digital Forensics

2.1. Computer forensics

Computers forensics is recognized as the discipline that combines elements of law and computer science to collect and analyze data from computer systems, networks, wireless communications and storage devices in a way that is admissible as evidence in a court of law. It undertakes the reconstruction of the sequence of events arising from an intrusion carried out by an external agent or as a result of illegal activities performed by an unauthorised user. As part of the larger field of forensics, computer forensics is lumped together with interesting fields of study, such as forensic anthropology, DNA analysis or forensic linguistics, which covers all areas where law and language intersect. It covers a wide set of applications, uses a variety of evidence and is usually supported by a number of different techniques.

Forensic authorship analysis is considered to be a part of forensic linguistics and it consists of inferring the authorship of a document by extracting and analyzing the writing features from the document content. During the last two decades , authorship analysis of computer-mediated communication (CMC) or online documents for prosecuting terrorists, pedophiles and scammers in the court of law, has received great attention in several studies (O.de Vel , Abbasi & Chen, Koppel) and has been growing in prominence.

2.2. E-mail Forensics

The field of forensic linguistics has to increasingly deal with email as this is becoming an important form of communication for many computer users. Email is considered to be one of the most widely used forms of CMC. Due to its salient features, it is the preferred source of written communication for almost every population connected to the Internet as well as for many companies and government departments. It is quick, ansychronous and used for various purposes ranging from formal to informal communication. Formal emails include official correspondence, meeting or seminar calls, business communication etc., while informal emails have to do with personal messages, greetings or invitation between family and friends. It is also used in the exchange and broadcasting of messages or documents for conducting electronic commerce. All these examples of legal applications of email are overshadowed by a number of illegitimate purposes of email, which can be misused for the distribution of unsolicited and/or inappropriate messages and documents.

Information security is gaining good attention from experts in the community especially after the growing penetration of the e-based systems and e- information at large- scale worldwide (Casey, 2010). As trillions of business letters, financial transactions, government orders and friendly messages are exchanged through email each year, one can notice that the increase in email traffic comes also with an increase in cybercrime. According to Loader and Thomas (Cybercrime, 2000), any illegal act using CMC is termed as a cybercrime. Many unique features of email enable the facilitation of criminal activity, such as phishing, spamming, email bombing, threatening, cyber bullying, racial vilification, child pornography, sexual harassment etc. Emails are also abused for commiting infrastructure crimes by transmitting worms, viruses, Trojan horses, hoaxes and other malicious executables over the Internet. An interesting and prolific aspect of email's illegal use is sending 'spam' mails. This type of mails consists of solicitous messages that the receiver is not interested in. Most of them are sent for advertisement purposes, but there is also another type of spam, which is sent specifically to obtain

personal information. The messages may contain disguised links that appear to be for familiar websites but in fact lead to phishing web sites or sites that are hosting malware. Spam email may also include malware as scripts or other executable file attachments.

Terrorist groups and criminal gangs are also using email systems as a safe channel for their communication. The 9/11 Comission report (Comission report, 2002) reveals that a number of emails were sent by the terrorists before the event took place. The sender and receiver attempted to avoid allied collection of this operational messages by triggering presumed 'spam' filters (Mathematics and the NSA, 2015). The investigations of the Mumbai attacks in 2008 (Mumbai terror attacks, The Economic Times 2011) show that some emails were discovered revealing hard evidence of the attacks.

Like other criminals, the cyber criminals attempt to hide their true identity. One can easily do this while sending fraudulent emails and can spread fear among a large population, across a large geographic location. Presently, there is no adequate proactive mechanism to prevent email misuses. In this situation, e-mail authorship attribution can help email service provider to detect a hack and the recipient to detect a fraud (Rahman Khan, 2012).

2.3. Challenging e-mail characteristics.

Authorship analysis has been quite successful in resolving authorship attribution disputes over various types of writings (Mendenhall, 1887). However, e-mail authorship attribution poses special challenges due to its characteristics of size, vocabulary and composition when compared to literary works (de Vel et al., 2001a and de Vel et al., 2001b). Here are the main problems a linguist can face when researching digital corpora based on e-mail communication.

- E-mails are short in length. Literary documents are usually large in size, comprising of at least several paragraphs; they have a definite syntactic and semantic structure. In contrast, e-mails are short and usually do not follow well defined syntax or grammar rules.
- The composition style used in formulating an e-mail document is often different from normal text documents written by the same author, thus it could be described as a combination of formal writing and a speech transcript.
- The author's composition style used in e-mails can be very depending upon the intented recipient and evolve quite rapidly over time. The composition of formal e-mails differ from informal ones. Even in the context of informal e-mails there could be several composition styles (one style for personal relations and one for work relations).
- The vocabulary used by authors in e-mails is not stable, facilitating imitation.
- Similar vocabulary subsets (technology based words) may be used within author communication.
- E-mails have few sentences/paragraphs making contents profiling based on traditional text document analysis techniques (e.g "bag-of- words" representation) more difficult.
- CMC documents are often poorly structured (spelling and grammatical errors) as they are often written in para language.
- E-mail writers are more likely to eliminate many function words thereby making it difficult to calculate the word summary feature.
- Some e-mail authors may use IM (Instant Message) type abbreviations in their writings such as "lol" or "btw", which can represent different phrases depending upon the context of the message or sentence.
- Especially when approaching authorship identification methods, the identification of an author is usually attempted from a small set of known candidates and the text body of the e-mail is not the only source of identifying an author. Evidence in the form of the e-mail headers, e-mail trace route, e-mail attachments, file-time stamps should be used in conjunction with the analysis of the e-mail text body.

2.4. Introduction to the Problem

The e-mail has brought many advantages to the work place and to individuals. However it is closely related to an alarming increase of cybercrimes. E-mail systems are inherently vulnerable to misuse for three reasons (Bogawar, 2012).

a. An e-mail can be spoofed and the meta data contained in its header about the sender and the path along which the message has travelled can be forged or anonymised. An e-mail can be routed through anonymous e-mail servers to hide the information about its origin.
b. E-mail systems are capable of transporting executables, hyperlinks, Trojan horses, and scripts.
c. The Internet including e-mail services is accessible through public places, such as net cafes and libraries, which further deteriorates the anonymity issues.

In order for digital forensic investigators to determine if a genuine user has been compromised or not, many methods and techniques have been used. In the context of cyberspace, a digital document found can be used as an evidence to prove that a suspect is the author of the document in dispute or not. Digital forensic investigators are obliged to provide credible proof in order to prosecute a suspect in a court of law. If the suspect authors are unknown, they have to deal with what is commonly known as an authorship identification problem.

The methods used to confront this authorship analysis problem include processes that have not been standardized in the e-mail forensic field. Even though many successful techniques have been invented for authorship attribution of literaly works, these techniques are unable to perform well in e-mail authorship attribution due to informality and short size of e-mails. This fact makes it difficult to provide legally admissible evidence in the court of law.

2.5. Research Questions

The research can be divided in two phases:

- Authorship identification by:

  - applying n-gram based author profiles and then classifying them by using a Support Vector Machine learning algorithm

  - applying the concept of frequent patterns (a.k.a. frequent itemset) (Agrawal et al., 1993) from data mining on the n-gram baseline in order to extract frequent n-gram patterns

  - and then use the same machine learning algorithm (SVM) comparing the performance of both methods.

Our main research questions are:

- Evaluation of both methods concerning identification problems.
- Compare the performance of these methods when used in a small e-mail dataset.

Chapter 3: Background of the Study

Most previous studies on authorship analysis focus on general text documents. Studies on CMC or online documents are limited. Similarly, features of online documents , such as structural features of e-mails , are different than the traditional textual works. However, the trend of using stylometric features is also found in e-mail authorship attribution studies not only for authorship identification, but also for authorship verification and authorship characterization.

3.1. Methods commonly used in short text authorship attribution.

3.1.1. Stylometry and feature selection.

Stylometry can be used for the author identification for text documents as the non–repudiation and integrity of the message are the major concerns. Stylometry is not only identifying a writing pattern but also the gender of the author. Stylometric study is used to identify and authenticate the authorship of e-mail text messages (Calix et al, 2008). The interest has been growing in applying stylometry to the content generation where the content is checked whether it is original or copied from others style. Shane Bergsma, Matt Post, David Yarowsky are evaluating stylometric techniques in the novel domain of scientific writing (2012).

Features

Stylometric features have been extensively used for long time .More than 1000 stylometric features comprising of lexical, syntactic, structural, content-specific, and idiosyncratic characteristics have been used in many studies . In these studies, specific author features such as unusual diction, frequency of certain words, choice of rhymes, and habits of hyphenation have been used as tests for author attribution. These authorial features are examples of stylistic evidence which is thought to be useful in establishing the authorship of a text document.

Stylometric features used in early authorship attribution studies were character or word based, such as vocabulary richness metrics (e.g., Zipf's word frequency distribution and its variants), word length etc. Some earlier works that have surveyed or compared various types of feature sets include Forsyth & Holmes (1996), Holmes (1998), McEnery & Oakes (2000), Love (2002), Zheng et al. (2006), Abbasi & Chen (2008) and Juola (2008). One of the advantages of modern machine learning methods is that they permit us to consider a wide variety of potentially relevant features without suffering great degradation in accuracy if most of these features prove to be irrelevant.

This trend of using stylometric features is also found in e-mail authorship attribution studies. For example, F. Iqbal et al. used 292 stylometric features and analyzed these features using different classification and regression algorithms. Chen and Hao's (2011) also extracted 150 stylistic features from e-mail messages for authorship verification. B. Allison et al. generated the grammar rules used in the e-mail and used these as features. Canales et al.(2011) extracted keystroke dynamics and stylistic features from sample exam documents for the purpose of authenticating online test takers .The extracted features consisting of keystroke timing features and 82 stylistic features were analyzed using a K-Nearest Neighbor (KNN) classifier.

Feature Selection

The stylistic features can be categorized as lexical, syntactic, semantic, and application specific.

- **Character features** are commonly used and refer to letter frequency, capital letter frequency, total number of characters per token and character count per sentence e.t.c. It is believed that they are the most powerful character features (Iqbal, Fung, Khan, & Debbabi, 2010). These features can imply the author's preference of using some special characters (Iqbal, Fung, Khan, & Debbabi, 2010). Moreover, character *n*-grams, which are consecutive sequences of *n* characters, have been proved to be effective to solve the topical similarity problems (Damashek, 1995).

- **Lexical features** are related to the words or vocabulary of a language and also known as word-based features/token-based features. Recent studies have used more than 1000 frequently used words to represent the style of an author. Lexical features encompass not only the frequency of characters or words found in a text but also vocabulary richness, sentence/line length, word length distribution, n-grams and lexical errors. They are language- independent and can be applied to almost all the languages with the assistance of a tokenizer. Moreover, some researchers (Escalante, 2011; Mikros & Perifanos, 2011; Tanguy et al., 2011) have used word *n*-grams to solve the authorship attribution problems. N-grams are tokens formed by a contiguous sequence of n items. The most frequent n-grams constitute the most important feature for stylistic purposes.

  Vocabulary richness measures the diversity of vocabulary in a text by quantifying the total number of unique vocabulary, the number of hapax legomenon (i.e., a word which occurs only once in a text) and the number of hapax dis legomenon (e.g., dis legomenon or tris legomenon, referring to double or triple occurrences). That might be, however, ineffective because the difference between texts written

by the same author can be as different as the texts written by different authors with regard to the vocabulary richness (Hoover, 2003).

What are N-grams?

N-GRAMS are a promising alternative text representation technique for stylistic purposes based on the old theoretical notion of "double articulation" reffering to the two levels into which language can be divided. Meaningful units of sound, called morphemes, make up the first level, while the second level consists of phonemes, or sounds without meaning by themselves. This notion indicates that stylistic information is constructed in blocks of segments of increasing semantic load from character to word n-grams (collocations and lexical strings). The concept of N-grams was first introduced in Shannon's seminal paper on Information Theory. A token is generated by moving a sliding window across a corpus of text where the size of the window depends on the size of the token (N) and its displacement is done in stages, each stage corresponds to either a word or a character. Based on the different types of displacements, N-grams can be classified into two categories: 1) character based and 2) word based .

N-grams are able to capture complicated stylistic information on the lexical, syntactic or structural preferences of an author or even indicate grammatical and orthographic tendencies without the need for linguistic background knowledge (making application to different languages trivial).

- Word N-grams

  Word n-grams consist of groups of one, two, or more words and they have been widely used in the past as features for analysis. Many researchers (Escalante, 2011; Mikros & Perifanos, 2011; Tanguy et al., 2011) have used word *n*-grams to solve the authorship attribution problems mainly because word n-grams approach syntax organization including different lexical bundles, phrases, collocation structures among others. However, richness of vocabulary is claimed that it might

be ineffective because many word types from the texts are *hapax legomena*, which means they only appear once in the entire text.. This explains why word n-grams have been used as input features for automated techniques of distinguishing and identifying authors with both encouraging (Hoover, 2002, 2003; Coyotl-Morales et al., 2006; Juola, 2013) and poor (Grieve, 2007; Sanderson and Guenter, 2006) results.

- Character N-grams

  A promising alternative text representation technique for stylistic purposes makes use of character n-grams (consecutive sequences of *n* characters of fixed length). Character n-grams approach phonology and morphology capturing quantitative information regarding syllable structure, phonotactics, consonant clusters, prefix and suffix structure. Several authors have proposed that the frequencies of various character n-grams might be useful for capturing stylistic preferences. Additionally, the selection of parameter *n* of character *n*-gram features has significant impact on the result (Stamatatos, 2009). According to Stamatatos (2009), if the parameter *n* is small (e.g. 2, 3), then the character *n*-grams would be able to represent sub-word information such as syllable information, but it fails to capture the contextual information. If the *n* is large, it would be able to represent contextual information such as thematic information. Grieve (2007) has found that character bigrams work surprisingly well for attribution of newspaper opinion columns. Chaski (2005, 2007) found character n-grams to work well for attribution in a forensic context. Character n-grams have also been shown useful for related stylistic classification tasks such as document similarity (Damashek 1995) or determining the native language of the writer (Zigdon 2005), though Graham et al. (2005) found that character n-grams did not work as well as syntax-based features for stylistic text segmentation. Zhang and Lee (2006) find clusters of character n-grams that prove useful for a variety of text categorization problems. The caveats regarding content words apply also to the use of character n-grams, as many will be closely associated to particular content words and roots.

- **Syntactic features** can be divided into average of punctuation and part-of-speech (POS). Baayen, van Halteren and Tweedie first discovered the effectiveness of syntactic elements (e.g. punctuation marks and function words) in identifying an author (Baayen, van Halteren, & Tweedie, 1996). Syntactic pattern is an unconscious characteristic and it is considered to be more reliable than lexical information. However, even though punctuation is highly important for defining boundaries and identifying the meaning of paragraphs that are then split to setences and tokens, it is not always sufficient to analyze the punctuation before formatting the text.

The part-of- speech tagging (POS tag or POST) is to categorize the tokens according to their function in the context. Basic POS tags include the functional words that express a grammatical relationship. Moreover, it is believed that the function words are used unconsciously and it is consistent by the same author regardless of the topics and has a low possibility of being deceived (Koppel, Schler, & Argamon, 2009). Recently, Patchala, Bhatnagar and Gopalakrishnan (2015) demonstrated a very effective use of a syntactic feature of an author's writing – text's parse tree characteristics – for authorship analysis of email messages. They defined author templates consisting of context free grammar (CFG) production frequencies occurring in an author's training set of email message and then used similar frequencies extracted from a new email message to match against various authors' templates to identify the best match.

- **Semantic features** are called as rich stylometric features (Tanguy et al., 2011) and are related to the meaning of language. They involve factors such as the meaning of words, grammatical construction, semantic relationships, and content-specific features. According to the result of Tanguy et al. (2011), simply

depending on the rich stylometric features did not reach desirable results as the more detailed the text analysis required for extracting stylometric features is, the less accurate (and the more noisy) the produced measures. However, the combination of the rich features with poor features has improved the results obtained using them separately.

- **Content-specific features** are derived by measuring the use of certain vocabulary in the text and are dependent on the topics of the documents, which are a collection of the keywords in the specific topic domain (Iqbal, Fung, Khan, & Debbabi, 2010). These features can be useful when they identify the gender, age, or a specific group the author may be part of. For example, within the same group, authors tend to use identical taxonomy in their communication and each generation has its own unique vocabulary. Some approaches measure the use of words indicative of the individual's race, nationality, and even tendency towards certain types of violence [8], as well as the number of gender-specific words [18], and psycho-linguistic cues. The biggest disadvantage of the content-specific features is that they may vary substantially in different topics with the same author. Consequently, the high performance of one model using content-specific features may perform badly if the topic has changed (Koppel, Schler, & Argamon, 2009).Therefore, the selection of the content-specific features is tailor-made to a specific context and should be dealt carefully.

- **Structural features** are related to the organization and format of a text and are usually more flexible in online documents such as e-mail. The unit of analysis of structural features is the entire text document, and the structural features evaluate the overall appearance of the document's writing style (Iqbal, Fung, Khan, & Debbabi, 2010). These features can be categorized at the message-level, paragraph-level or according to the technical structure of the document [4]. As a matter of fact, analyzing a large number of features does not necessarily provide the best results, as some features provide very little or no predictive information.

However, in the authorship verification problem of computer–mediated online messages such as blogs and emails the structural features may be very promising (Koppel, Schler, & Argamon, 2009).

- **Idiosyncratic features** refer to the presence of mistakes, e.g. spelling mistakes,formatting and syntactic mistakes in the document (Iqbal, Fung, Khan, & Debbabi, 2010).For instance, the frequency of sentence fragments, run-on sentences, unbroken sequences of multiple question marks or other punctuation, words shouted in CAPS ,various categories of common spelling errors and so forth. Thus, it is difficult to make a collection of all the idiosyncratic features, but it is possible to make a list for each person based on analysis on the spelling errors and syntactic errors from the existing written documents of the author.

3.1.2. Machine Learning Approach

On essence machine learning is a subfield of computer science that evolved from the study of pattern recognition and computational learning theory in artificial intelligence. Machine learning is closely related to and often overlaps with computational statistics; a discipline which also focuses in prediction-making through the use of computers.The application of machine learning methods is straightforward: training texts are represented as labeled numerical vectors and learning methods are used to find boundaries between classes (authors) that minimize some classification loss function. The authorship analysis techniques include univariate, multivariate statistics, and machine learning techniques such as Support Vector Machine, Decision Trees, Neural Nets (Iqbal, Fung, Khan, & Debbabi, 2010).

- **Support Vector Machines** (SVMs) are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. Recently they gained popularity in the learning community [Vap98]. In its simplest linear form, a SVM is a hyperplane

that separates a set of positive examples from a set of negative examples with maximum interclass distance, the margin.

The formula for the output of a linear SVM is

$u = w * x + b$

where w is the normal vector to the hyperplane, and x is the input vector.

Of course, not all problems are linearly separable. Cortes and Vapnik [CV95] proposed a modification to the optimization formulation that allows, but penalizes, examples that fall on the wrong side of the decision boundary.

Support vector machines are based on the structural risk minimization principle [Vap98] from computational learning theory. The idea is to find a model for which we can guarantee the lowest true error. This limits the probability that the model will make an error on an unseen and randomly selected test example. The use of a structural risk minimization performance measure is in contrast with the empirical risk minimization approach used by conventional classifiers. Conventional classifiers attempt to minimize the training set error which does not necessarily achieve a minimum generalization error. Therefore, SVMs have theoretically a greater ability to generalize. An SVM finds a model which minimizes (approximately) a bound on the true error by controlling the model complexity (VC-Dimension). This avoids over-fitting, which is the main problem for other semi-parametric models. Unlike many other learning algorithms, the number of free parameters used in the SVM depends on the margin that separates the data and does not depend on the number of input features. Thus the SVM does not require a reduction in the number of features .This is clearly an advantage in the context of high-dimensional applications, such as text document analysis and authorship categorization, as long as the data vectors are separable with a large margin. SVMs require the implementation of optimization algorithms for the minimization procedure which can be computationally expensive. Many , scholars have applied SVMs to the problem of text document analysis and categorization

using approximately thousands of features in some cases, concluding that, in most of the cases, SVMs out-performs conventional classifier. SVMs also can be used for classifying e-mail text and documents as spam or non-spam and compared it to boosting decision trees. Comparative studies on machine learning methods for topic-based text categorization problems (Dumais et al. 1998; Yang 1999) have shown that in general, support vector machine (SVM) learning is at least as good for text categorization as any other learning method and the same has been found for authorship attribution (Abbasi & Chen 2005; Zheng et al. 2006).

Teng et al. (2004) and De Vel (2000) applied Support Vector Machine classification model over a set of stylistic and structural features for e-mail authorship attribution. More specifically De Vel et al. (2001b) and Corney et al (2002) performed extensive experiments and found that the classification

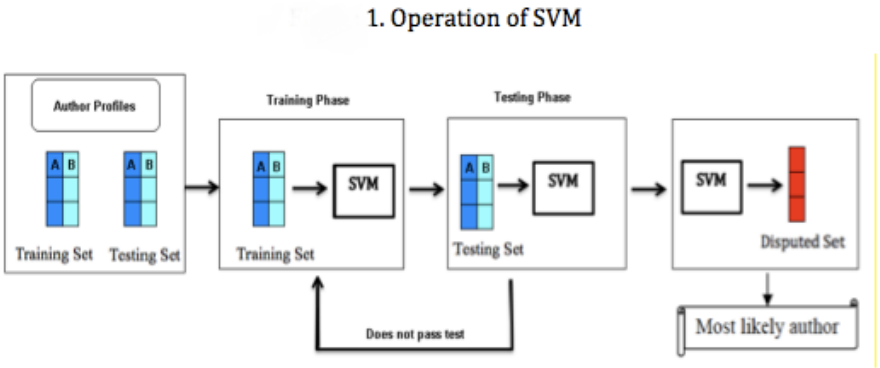


Figure 1- SVM Operation

accuracy decreases when the training set decreases, the number of authors increases, or the length of documents decreases. Some of the previous studies on authorship identification used SVM in order to identify patterns of terrorist communications (Abbasi, Chen 2005), the author of a particular e-mail for

forensic purposes (Iqbal,2010), as well as ways to collect digital evidence for investigations (Chaski,2005).

- **Distance measures** continue to be used in recent studies examining the efficacy of different metrics and feature sets. These compression-based similarity methods are motivated from the Kolmogorov complexity theory. The compression-based approaches are practical implementations of the information distances expressed in the non-computable Kolmogorov complexity. Having a distance metric to find similar objects, a straightforward approach to recognize unseen objects is to attribute them to the author of the most similar object in the training database. In other words, the nearest neighbour classification method can be applied. This simple notion is quite powerful, so such distance measures continue to be used in recent studies examining the efficacy of different metrics and feature sets. Distance measures can be applied between texts written by the same author (inter-compression distances), as well as between texts written by different authors (intra-compression distances).

  There are, however, some downsides to the use of the nearest neighbour rule. When using this approach, to classify an unseen object x, the distances between x and all training objects need to be calculated. This is a computationally expensive procedure. Another problem is that the nearest neighbour approach runs the risk of overfitting.

  One such method is Burrows's (2002a) Delta, which has been extended and used for a variety of attribution problems (Burrows 2002b; Hoover 2004a, 2004b).A number of other similarity functions computed as distance measures for authorship attribution have been applied to different feature sets as well (Craig 1999; Chaski 2001; Stamatatos et al. 2001; Keselj et al. 2003; van Halteren et al. 2005; Burrows 2007). Recently, Grieve (2007) has run an exhaustive battery of tests using this type of method. A related class of techniques was developed

earlier by Burrows (1987; 1989), who applied principal components analysis (PCA) on word frequencies to analyze authorship. The idea is to visualize the differences between texts written by different authors by projecting high-dimensional word-frequency vectors computed for those text onto the 2-dimensional subspace spanned by the two principal components; if good separation is seen between documents known to be written by different authors, then new texts may be attributed by seeing which authors' comparison documents are closest to them in this space. This method was elaborated on by Binongo and Smith (1999), and has been used to resolve several outstanding authorship problems (Burrows 1992; Binongo 2003; Holmes 2003). Furthermore, Jaccard's coefficient, (or 'Jaccard Index', 'Jaccard', or 'intersection distance') is a statistic used for comparing the similarity and diversity of sample sets. Jaccard is widely used as a similarity metric across a range of scientific disciplines such as ecology (Jaccard, 1912; Izsak and Price, 2001; Pottier et al., 2013; Tang et al., 2013) forensic psychology and crime linkage (Bennell and Jones, 2005; Woodhams et al., 2008; Markson et al., 2010) and document comparison (Rajaraman and Ullman, 2011; Deng et al., 2012; Manasse, 2012). Drawing on these various different uses of the coefficient, Jaccard has been introduced into forensic authorship analysis as a way of measuring the similarity or distance between questioned and known documents based on a range of different linguistic features (Grant, 2010, 2013; Wright, 2012; Larner, 2014; Juola, 2013).

- **Decision trees** are a well-known method of classification (Apte and Weiss, 1997) and have been proposed by (Alfonseca & Manandhar, 2002a) for extending WordNet with new concepts. Decision rules and decision tree based approaches to learning from text are particularly appealing, since rules and trees provide explanatory insight to end-users and text application developers Decision rules and decision tree based approaches to learning from text are particularly appealing, since rules and trees provide explanatory insight to end-users and text application developers. They introduce so called distributional (topic) signatures

to describe concepts. These descriptions are then used to compute similarity between concepts. New concepts are inserted into an existing taxonomy tree by traversing it top-down, at each step descending via the most similar child of the current node, stopping when that node is more similar to the new concept than any of its children. The size of the resulting tree is limited by cross-validation. The predictive performance of trees is sometimes not nearly as strong on unseen data as that obtained on the training data. This phenomenon is often described as overfitting, where the tree is too specialized to the training data. This may arise due to the existence of high variance in the data. Researchers have observed that the variance can be greatly reduced by inducing multiple decision trees from the same data. The classification of an unseen case is then determined by a weighted combination of the classifications assigned by the multiple trees. Decision trees have been used mainly for the problem of automatically filtering unwanted electronic mail messages (Carreras & Marquez 2001), but have also been applied to automatic text categorization by Apte, Damerau and Weiss (1998), who begun exploring methods for maximizing the predictive accuracy of the models constructed from the mining process. This an important requirement, particularly in real world applications, where noisy and limited samples are a pervasive problem. Additionally, Abbasi and Chen (2005) analyzed the individual characteristics of participants in an extremist group web forum using decision tree and SVM classifiers producing really efficient results.

- **Neural Networks** have recently been a matter of extensive research and popularity. Their application has increased considerably in areas in which we are presented with a large amount of data and we have to identify an underlying pattern**.** In machine learning and cognitive science, **neural networks** (**ANNs**) are used to estimate or approximate functions that can depend on a large number of inputs and are generally unknown. Some types of neutral networks used in authorship analysis, such as radial basis function, feed-forward neural networks, cascade correlation and Markov Chains, have great performance in e-mail

forensics. Among the earliest methods to be applied in authorship attribution were various types of neural networks, typically using small sets of function word as features (Matthews & Merriam 1993; Merriam & Matthews 1994; Kjell 1994a; Lowe & Matthews 1995; Tweedie et al. 1996; Hoorn 1999; Waugh et al. 2000). Recently, Graham et al. (2005) and Zheng et al. (2006) used neural networks on a wide variety of features.

- **Radial basis function (RBF) networks** are suitable for producing approximations to an unknown function f from a set of input data abscissa. The approximation is produced by passing an input point through a set of basis functions, each of which contains one of the RBF centers. They start with a number of prototype feature vectors for each class and assume that the feature vector of a new exemplar is 'close' to some prototype of its class. The distance to the prototype is measured by the common Euclidean distance or some generalized version weighting specific features. RBF-networks model the style of an author directly as a mixture of different styles, which may depend on topic and genre. Therefore they are especially suited to stylometric analysis. Also, prior knowledge can be used to initialize weight vectors. This is important for relatively small data sets, a situation that can easily arise in authorship attribution RBF-networks were used by D. Lowe and R. Matthews for stylometric analysis (1995). They used the frequency of five function words as features, normalized to zero mean and unit variance. They were used to discriminate plays of Shakespeare and Fletcher with a total of 50 samples for each author.

- **K-Nearest Neighbor (K-NN) Classifiers** are quite similar to RBF networks as they use the distance to prototypes as a criterion .The idea of the k-nearest neighbor (K-NN) classifiers (Silverman and Jones, 1989) is as follows. For an unlabeled email, the classifier searches for the k nearest training emails according to a certain distance function. Then, the unlabeled email is given the same label of the class, to which most of the k nearest training emails belong. Lam and Yeung

(2007) proposed a K-NN classification based approach for spam detection. The sender features, such as the numbers of emails received and sent by the email user, respectively, and the number of interactive neighbors a user has, are extracted from a social network built from the email logs. The mean K-NN similarity score used to label an unlabeled email is the mean of the distances between the email sender and her k-nearest neighbors. The positive/negative sign of the score can be used to classify whether the email is spam or not. Halvani, Steinebach and Zimmermann proposed their k-Nearest Neighbor (k-NN) based Authorship Verification method for the Author Identification (AI) task of the PAN 2013 challenge. Their method follows an ensemble classification technique based on the combination of suitable feature categories. For each chosen feature category they applied a k-NN classifier to calculate a style deviation score between the training documents of the true author A and the document from an author, who claimed to be A. Kucukyilmaz et al. (2008) used k-NN classifier to identify the gender, age, and educational environment of a user.

- **Adaboost**

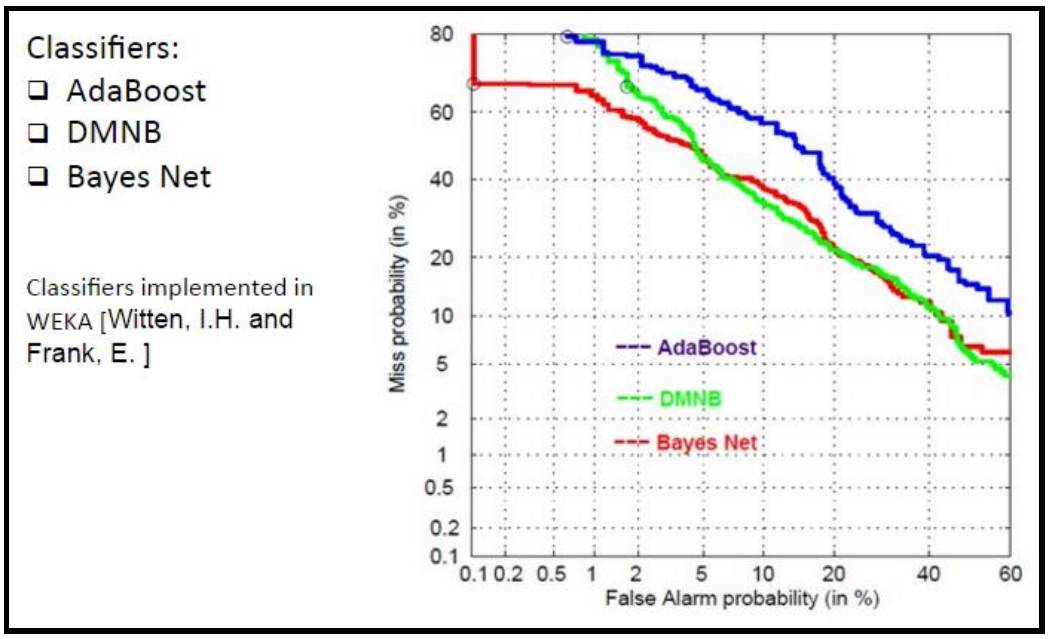
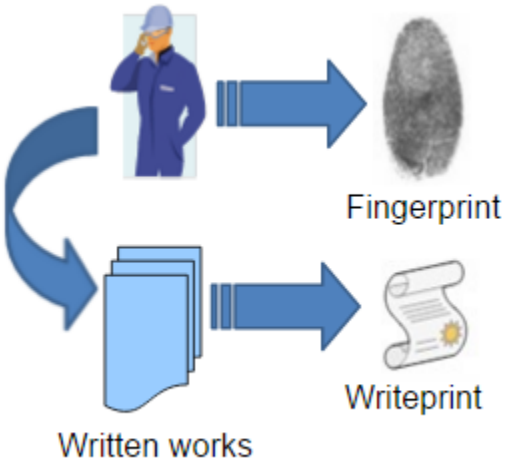


Figure 2-Experimental Evaluation of classifiers

is a machine learning meta-algorithm formulated by Yoav Freund and Robert Schapire who won the Gödel Prize in 2003 for their work. It can be used in conjunction with many other types of learning algorithms to improve their performance. Unlike neural networks and SVMs, the AdaBoost training process selects only those features known to improve the predictive power of the model, reducing dimensionality and potentially improving execution time as irrelevant features do not need to be computed. Cheng et al. (2011) investigated the author gender identification from text by using Adaboost and SVM classifiers to analyze 29 lexical character-based features, 101 lexical wordbased features, 10 syntactic, 13 structural, and 392 functional words. Abdallah et al. also used Adaboost combined with Decision trees, SVM, Random forest, Functional tree, Logistic, Naive Bayes and his results indicated that the classification accuracy obtained using the Simple Logistic and AdaBoost was comparatively better than the recognition accuracy achieved using the other machine learning classifiers.

- **Naive Bayesian Network classifiers** are directed acyclic graphs that allow efficient and effective representation of the joint probability distribution over a set of random variables. In Literature it is found that Bayes theorem plays a critical role in probabilistic learning and classification. It uses prior probability of each category given no information about an item. De Vel (1999) studied the comparative performance of text document categorisation algorithms using the Naive Bayes, Support Vector Machines, multi-layer Perceptron and k–NN classifiers. Some researchers studied e-mail text classification in the context of automated e-mail document filtering and filing. Sahami et al (1998) focused on the more specific problem of filtering junk e-mail using a Naive Bayesian classifier and incorporating domain knowledge using manually constructed domain-specific attributes such as phrasal features and various non-textual features. Khan also proposed a simple but powerful and robust method based on ensemble method and Naïve Bayes classifier (2012).

- **Frequent patterns .** Iqbal et al. (2008) proposed another approach named AuthorMiner , which consists of an algorithm that captures frequent lexical, syntactical, structural and content-specific patterns. Their main idea is to model the **writeprint** of a person, in other words to capture the writing style of a person from his/her written text by employing the concept of *frequent patterns.* The experimental evaluation used a subset of the Enron dataset, varying from 6 to 10 authors, with 10 to 20 text samples per author. The authorship identification accuracy decreased from 80.5% to 77% when the authors' population size increased from 6 to 10. Their approach ensured that insignificant stylometric features can't



have a great impact on extracting the writeprint of an author, simply because they are not frequent and therefore irrelevant. Thus, an investigator can simply add all available stylometric features, leaving behind the burgain of worrying about the appropriate feature selection, without causing the degradation of quality. They also suggested that the identification of sub-writeprints could improve the accuracy of authorship identification by revealing the fine-grained writing styles of an individual. This provides valuable information for investigators or authorship analysis experts who can present the writeprint and explain the finding in a court of law, something that cannot be always achieved by traditional authorship attribution methods, such as SVMs or neural networks. However, in the pre-processing phase, the spaces, punctuations, special characters and blank lines, which are important information that can be used to mine author's writing-styles are removed.

Chapter 4: Methodology

4.1. Statement of the Problem

A digital document can be used as an evidence to prove the guilt of a suspect involved in cybercrime. If the suspect authors are unknown ,then we deal with an authorship identification problem. In some cases the identification of the author is not needed . It is enough just to know whether the document in dispute was written by a certain author of the documents given. This problem is commonly known as an authorship verification problem. In both cases, the founds must be powerful enough to stand in a court of law. Many methods concerning both problems have been proposed through out time and have been faced by many forensic linguistic experts. Our main research problem is to compare and combine two of the most successful methods applied and check their effectiveness on a small dataset for authorship identification purposes.

Derived from the problem description, the main research question is described as follows:

Given a set of suspicious and source documents provided by Pan'11 lab, the task is to select a limited number of documents and suspects, build a n-gram author profile of an author's writing for identification purposes and then try to improve the performance of the method using frequent patterns to extract the unique writing style of an author.

4.2. Purpose of the Study

The objective of this research is to creatively compare two identification models and find out which methods can reach a relatively high performance concerning digital forensic documents in order to encourage further research.

4.3. Pre-processing

4.3.1. Corpus

For our research we use a corpus (also available for the **PAN Author** Identification task) based on the Enron email corpus, to account for several different common attribution and verification scenarios. More specifically we used the "Large" training set that initially contains 9337 documents by 72 different authors. Among them we've chosen 30 texts of 20 authors and extracted the 2000(4*500) most frequent n-grams from each group in order to make the whole process more manageable and less time consuming. . The PAN 2011 data set has an average e-mail length of about sixty words. Author set size has received only limited attention so far, but nevertheless has a significant impact on classification performance as well as on the features in the attribution model.

Personal names and email addresses in the corpus have been automatically redacted, and replaced (on a token-by-token basis) by ¡NAME/¿ and ¡EMAIL/¿ tags, respectively. This redaction is admittedly imperfect, but random spot- checking was applied to reduce the likelihood of missing occurrences. Other than this redaction, each text is typographically identical to the original electronic text, so systems could, in principle, rely on line length, punctuation, and the like. Finally, authorship was determined based on From: email headers; this necessitated determining, in some cases, that multiple email addresses corresponded to the same individual.

4.3.2. Why N-grams?

N-grams grams are able to capture complicated stylistic information on the lexical, syntactic or structural preferences of an author and they have **great performance in language independent processing,** One of their many advantages is the fact that they **resist in a robust manner to the presence of different kinds of textual errors** without affecting the quality of research, mainly because errors affect only a limited number of n grams (Cavnar and Trenkle, 1994), proving their tolerance to noise. Furthermore, they automatically **detect words that share the same root form** and specifically character n-grams **unavoidably capture thematic information** in addition to the stylistic information (Stamatatos, 2009). This capacity can be viewed either as an advantage, when they indicate the preference of the authors on specific thematic-related choices, or as a disadvantage, when the available texts are not on the same thematic area. Aditionally, it is quite **simple to measure** them.

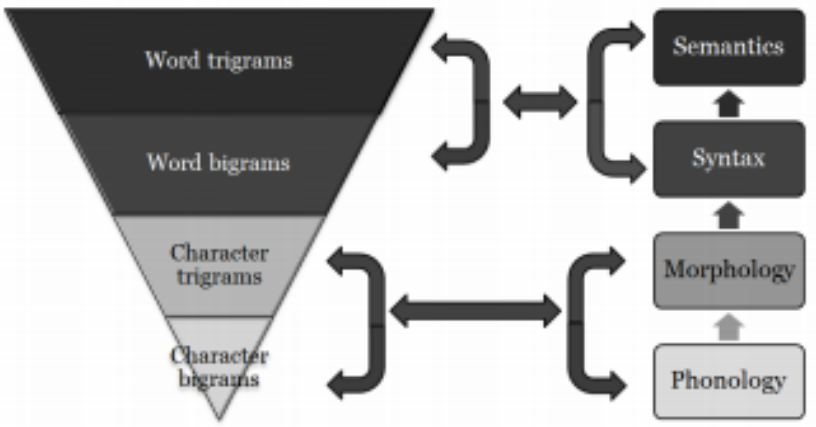


Figure 3-A hierarchical representation of n-gram features and related linguistic levels. (Mikros & Perifanos 2013, p.3).-

4.3.3. N-gram feature selection

The proposed method for variable-length n-gram feature selection is based on an existing approach for extracting multiword terms from texts. For our research feature we combined single groups of :

- **Character Bigrams (cbg):** Character Bigrams are a special case of N-grams and are used in one of the most successful language models for speech recognition .Bigram frequency is one approach to statistical language identification and many researches have proved their effectiveness in authorship attribution studies.
- **Character Trigrams (ctg):** Character trigrams are long enough to capture morphology without mapping too obviously to specific words. They also have the additional merit that they BTW, etc.also represent common email acronyms like FYI, FAQ,
- **Word Bigrams (wbg)**: Word bigrams are gappy bigrams with an explicit dependency relationship and have been used successfully in authorship attribution.
- **Word Trigrams (wtg):**Word trigrams have great performance at classifying data in NLP and they give plenty of contextual and structural information about an author's style.

The most frequent n-grams were detected using the Ngram Statistics Package (NSP) [2], a PERL module designed word and character n-gram identification created by George Mikros and Kostas Perifanos for the PAN'11 lab contest. Tokenization in n-gram identification followed the following rules:

– Token was identified any sequence of alphanumeric characters using the following regular expression: \w+.

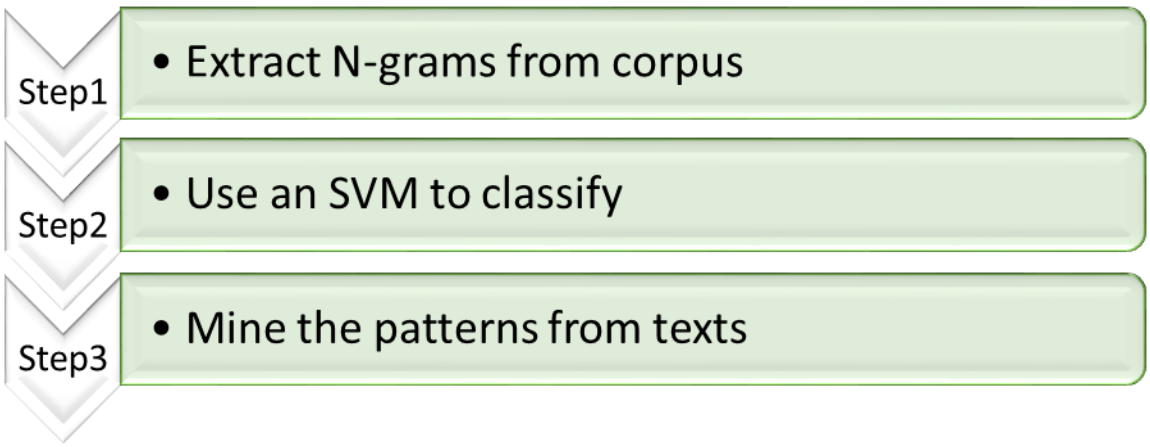
In order to define words we add the plus sign (+) next to w. A regular expression followed by a plus sign (+) matches one or more occurrences of the one-character regular expression.

– As tokens were identified also the punctuation marks defined in the following regular expression: [\.,;:\?!]. Punctuation usage often reflects author-related stylistic habits [8] and n-grams with punctuation can capture better possible these stylistic idiosyncrasies.

– All tokens were converted to lowercase.

4.3.4. Processing

Our methodology is organized as follows:



Output files from NSP were converted to vectors using custom PERL script which aggregated n-gram counts from each text file and normalized their frequency to the text length. After calculating n-gram vectors from the corpus we ended up with a tab delimited text file presenting the structure of our n-gram list.

```
1  254483
2  t<>h<>e<>3973 23430 11825 30264 6542 5249 5280
3  a<>n<>d<>1561 19835 17284 8476 3497 1949 2352
4  i<>n<>g<>1535 18456 17284 4517 4538 1621 1902
5  i<>o<>n<>1340 18456 18333 17284 1548 1896 3335
6  e<>n<>t<>1251 30264 17284 23430 3064 3935 2983
7  t<>i<>o<>1100 23430 18456 18333 3294 2536 1548
8  t<>h<>a<>1001 23430 11825 19835 6542 2248 2093
9  a<>t<>i<>936 19835 23430 18456 3314 2226 3294
10 h<>e<>r<>833 11825 30264 15058 5280 1224 4173
11 h<>a<>t<>824 11825 19835 23430 2093 1162 3314
12 t<>e<>r<>765 23430 30264 15058 2699 1911 4173
```

Figure 4-Structure of N-gram list

First line depicts the total frequency of occurrence of character trigrams in the corpus.

- 254,483 character trigrams (tokens not types)

First number depicts the frequency of the specific trigram

- t<>h<>e<>: 3973 occurrences in the corpus.

The second, third and fourth numbers denote the number of trigrams in which the tokens “t", “h" and “e" appear in the first, second and third positions respectively. Thus, “t" occurs as the token in the first position in 23430 trigrams. Similarly, the tokens “h" and “e" appear as the second and third tokens respectively of 11825 and 30264 trigrams.

The fifth number denotes the number of bigrams in which “t" occurs as the first token and “h" occurs as the second token. The sixth number denotes the number of bigrams in which “t" occurs as the token in the first place and “e" occurs as the token in the third place. The seventh number denotes the number of bigrams in which “h" occurs as the token in the second place and “e" occurs as the token in the third place.

For the purposes of the authorship attribution tasks we used the **Caret package** (Kuhn et al., 2012) of R(short for classification and regression training) that contains functions to streamline the model training process for complex regression and classification problems. We then created a trainControl object for controlling the validation procedure of the algorithm and applied a 10 fold-cross validation in order to estimate how accurately the predictive model will perform in practice.

When extracting n-grams we end up with many zero n-grams that increase the size of the dataset without adding information gain. “In some situations, the data generating mechanism can create predictors that only have a single unique value (i.e. a "zero-variance predictor"). For many models (excluding tree-based models), this may cause the model to crash or the fit to be unstable. Similarly, predictors might have only a handful of unique values that occur with very low frequencies, i.e. this is a very common situation with n-gram profiles where many low frequency n-grams appear only in a handful of texts and have a 0 in all the other. The concern here that these predictors may become zero-variance predictors when the data are split into cross-validation/bootstrap sub-samples or that a few samples may have an undue influence on the model. These "near-zero-variance" predictors may need to be identified and eliminated prior to modeling. To

identify these types of predictors, the following two metrics can be calculated" (Mikros, Profile Lab presentation):

◦ the frequency of the most prevalent value over the second most frequent value, which would be near one for well-behaved predictors and very large for highly-unbalanced data (freqCut).

◦ the "percent of unique values" is the number of unique values divided by the total number of samples (times 100) that approaches zero as the granularity of the data increases.

If the frequency ratio is less than a pre-specified threshold and the unique value percentage is less than a threshold, we might consider a predictor to be near zero-variance. Another method recommended during the pre-processing phase is to apply Regression Analysis that determines if an explanatory variable is statistically significant or not**.**

We trained the Random Forests algorithm in order to calculate additionally the importance of the variables in the classification and examined the model performance, Confusion Matrix and the Variable Importance of the 5 most important variables of the model. Furthermore we trained the SVM algorithm normalizing the data and then tried 3 different values of the SVM parameters.

These are the statistical information of our sample:

Authors (20)	E-mails per author	Words	Max. words per mail	Min. words per mail

aa	30	1446	145	5
ab	30	327	60	1
ac	30	1547	189	9
ad	30	740	154	1
....				
as	30	706	116	5
at	30	991	97	4

Sampes	Predictos	Classes
600	2000	20

The results show a low accuracy when using a small corpus with n-gram feature selection. This is absolutely expected as not only we have a limited dataset, but the average number of words found in each e-mail are also limited. This fact corresponds to real life situations, as in the initial stage of investigation the investigators usually have very little information of the case and the true authors of suspicious e-mail collection.

Sample size	***Character bigrams***	***Character trigrams***	***Word bigrams***	***Word trigrams***
540	**0,214**	**0,173**	**0,063**	**0,066**
540	0,201	0,159	0,042	0,044
540	0,196	0,154	0,044	0,046
540	0,198	0,156	0,044	0,046

Accuracy was used to select the optimal model using the largest value and it rises up to 21,4% as shown in the above table.

When experimenting with less suspects (for instance 3) the accuracy increases to a large extent using the same feature amount (the most often n-grams of texts) and it reaches 42% as shown below.

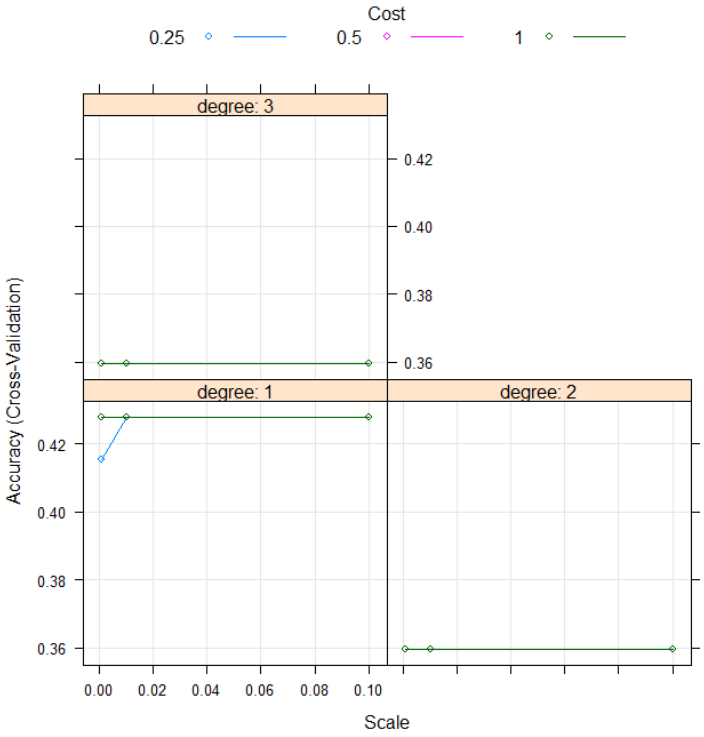


Figure 5-Prediction accuracy for 3 authors

However, SVM classification proves to be inefficient for small e-mail datasets with limited number of words, especially when the number of suspects increases. The SVM classification is appropriate for large training corpora with a great amount of extracted characteristics, but it is time consuming and needs a great deal of computational power.

Chapter 5. Expanding the model: Frequent N-gram Patterns

5.1. Pattern Mining

Pattern mining consists of using/developing data **mining algorithms** to discover interesting, unexpected and useful patterns in databases. By the term interesting some researchers indicate the patterns that appear *frequently* in a database. Other researchers tend to discover *rare patterns*, patterns with a high *confidence*, the top patterns, etc.

The idea of using frequent stylometric patterns for authorship identification in e-mail forensics was firstly applied by Iqbal et al (2008). Their proposed method, AuthorMiner, consisted of a novel approach of authorship attribution and formulated a new notion of write-print based on the concept of frequent patterns. Unlike the write-prints in previous literature that are a set of predefined features, their notion of write-print was dynamically extracted from the data as combinations of features that occur frequently in a suspect's e-mails, but not frequently in other suspect's e-mails. They also noticed that a person may have different writing styles when addressing to different recipients (e.g formal- informal level) and therefore they proposed an improved version of AuthorMiner based on this hypothesis. Their AuthorMiner 2 groups the training sample messages by the types of message recipients and identifies stylistic variations by capturing the sub-writeprints in order to improve the accuracy of authorship attribution.

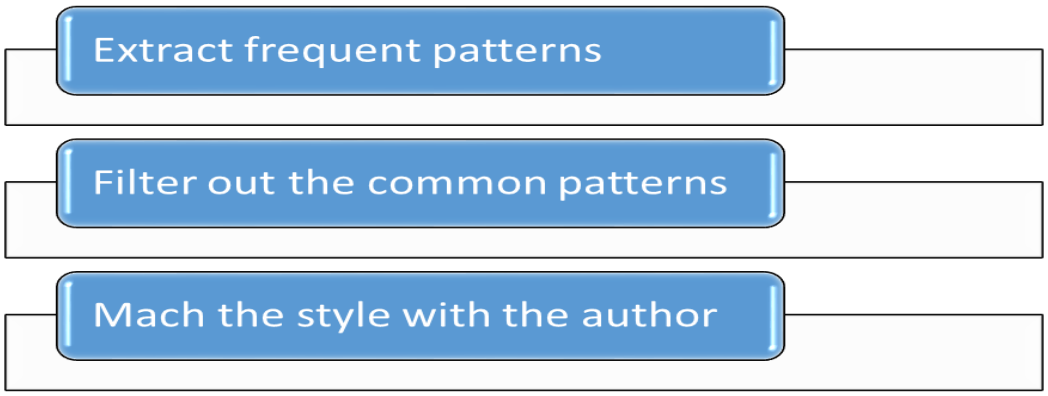
The present study is based on the frequent pattern mining approach of Iqbal (2008). However, the frequent mining approach will be applied on the n-gram findings of the previous step rather than on typical stylometric features. Our goal is to extract the frequent n-gram patterns of the texts and then match a suspicious mail with the most plausible author. This idea is based on the hypothesis that n-grams contain a huge amount of stylometric information and could hypothetically lead to a more precise writing fingerprint of an author. Additionally, the combination of n-grams and data mining could solve theoretically flaws from both sides. For instance, n-grams fail to account by themselves non-contiguous patterns, while pattern mining methods can do so quite naturally. On the other hand, in the pre-processing phase of the frequent pattern mining, the spaces, punctuations, special characters and blank lines are removed, even though they offer important information that can be used to mine an author's style. All these are captured when extracting n-grams from texts.

5.2. Steps of the frequent n-gram pattern model.

Several researchers use n-grams as features for authorship attribution and authorship verification tasks. A distinction is made here between word n-grams and character n-grams. The former is a pattern of n words, whereas the latter is a pattern of n characters.

We choose to study the performance of character n-grams, as it's hard even impossible to extract safe word- patterns from a limited number of texts. An important characteristic of the character-level n-grams is that they avoid (at least to a great extent) the problem of sparse data that arises when using word-level n-grams. That is, there is much less character combinations than word combinations, therefore, less n-grams will have zero frequency (Kanaris, 2006).

Our method will be elaborated in three main steps, as shown below:



5.2.1 Step 1

5.2.1.1 First phase of model design

The problem of authorship attribution in e-mail forensics can be refined into two subproblems:

- to identify the n-gram fingerprint FP(Ei) from each set of e-mails Ei ¸ {E1,., Em}
- to determine the author of the malicious e-mail m by matching m with each of {FP(E1),., FP(Em)}

Before moving to the first step we had to adjust the Perl script we used before, so that it will provide the raw frequency of n-grams.

Filename	h<>e<>	i<>n<>	e<>r<>	a<>n<>	r<>e<>	o<>n<>
100228_a1146.xml.txt.ng	5	2	3	4	5	4
100228_a1165.xml.txt.ng	4	3	3	9	2	4
100228_a1376.xml.txt.ng	0	0	0	0	0	0
100228_a146.xml.txt.ng	2	2	0	3	1	3
100228_a1496.xml.txt.ng	1	4	1	1	4	0
100228_a1498.xml.txt.ng	10	1	6	4	1	10
100228_a1530.xml.txt.ng	3	3	3	5	1	2
100228_a1623.xml.txt.ng	10	12	7	9	7	1
100228_a1646.xml.txt.ng	0	1	0	0	0	0
100228_a1852.xml.txt.ng	6	5	8	1	1	2
100228_a1987.xml.txt.ng	2	2	1	1	0	1

Figure 6-N-grams raw frequency

Problem of limited texts and suspects

If the number of texts is small (=30) , it is usually considered insufficient and therefore inefficient to extract frequent patterns directly without grouping first the texts of each author according to the similarity of their writing styles. But this applies when one does not know the authors of the anonymous texts. Forensic experts not only want to identify the author given a small set of suspects, they also want to make sure the author is not someone else not under investigation. They often deal with short emails or letters and have only limited data available. Clustering then is considered necessary. In this case, we know a priori the suspects from the information provided by Pan Corpus.

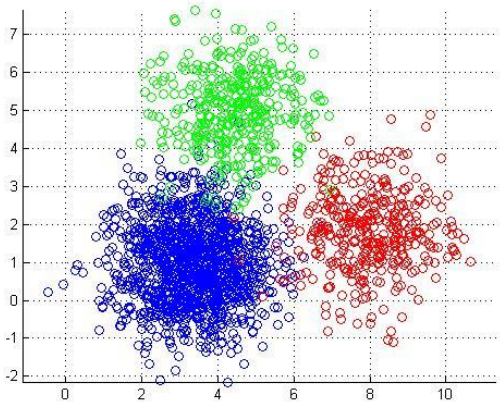


Figure 7-Implementation of the original k-means clustering algorithm

Even though clustering is used for identifying the topic of a discussion, it is proved to also be effective for identifying e-mails written by the same author (Iqbal et al., 2010). Thus, it creates groups of stylistics based on the hypothesis that every author has a unique style determined by his writing preferences. K-means clustering is a method of vector quantization, originally from signal processing, that is popular for cluster analysis in data mining. It separates the observations of the *n*-by-*p* data matrix X into k clusters, and returns an *n*-by-1 vector containing cluster indices of each observation. Rows of X correspond to points and columns correspond to variables. By default, k-means uses the squared Euclidean distance measure and the *k*-means++ algorithm for cluster center initialization. In our case, this step is not needed.

The task is to now extract the frequent stylometric patterns from each message group with same stylometric features Mi of suspect Si. For this we use Apriori algorithm in the context of mining frequent stylometric patterns. The Apriori Algorithm is an influential algorithm for mining frequent itemsets for boolean association rules. Some key concepts for Apriori algorithm are:

• Frequent Itemsets: The sets of item which has minimum support (denoted by Li for ith-Itemset).

• Apriori Property: Any subset of frequent itemset must be frequent.

• Join Operation: To find Lk , a set of candidate k itemsets is generated by joining Lk-1 with itself.

Apriori is a level-wise iterative search algorithm for mining frequent itemsets for Boolean association rules. It is designed to be applied on a transaction database to discover patterns in transactions made by customers in stores. But it can also be applied in several other applications. A transaction is defined a set of distinct items (symbols). Apriori takes as input (1) a minsup threshold set by the user and (2) a transaction database containing a set of transactions. Apriori outputs all frequent itemsets, i.e. groups of items shared by no less than minsup transactions in the input database.

In stylometry Apriori uses frequent stylometric κ-patterns to explore the frequent stylometric (κ+1)-patterns. The set of frequent stylometric patterns are found by scanning the messages of each group G that corresponds to an author, accumulating the support count of each stylometric pattern, and collecting the stylometric pattern F that has $support(F|G) \geq min\ sup$. The features in the resulting frequent 1-patterns are then used to find frequent stylometric 2-patterns, which are used to find frequent stylometric 3-patterns, and so on, until no more frequent stylometric (κ + 1)-patterns can be found.

A frequent pattern is a pattern such that its support is higher or equal to minsup**.** The support of a pattern (also called "frequency") is the number of transactions that contains the pattern divided by the total number of transactions in the database. A key problem for algorithms like Apriori is how to choose a minsup value to find interesting patterns. There is no really easy way to determine the best minsup threshold. Usually, it is done by trial and error, but it highly depends on the datasize. In the present study we suppose that the minimum support equals to 0.2 as we have a limited text selection and the percentage has to be adjusted so that we can have results, in other cases this number differs respectively to the size of the corpus and the aims of the study. The items having support ≥ 0.2 are then considered frequent stylometric 1-patterns, denoted by $L_1 = \{\{X_2\}, \{Y_1\}, \{Z_1\}, \{Z_2\}\}$. Then, we join L_1 with itself, i.e., L_1 on L_1, to generate the candidate list $L_2 = \{\{X_2, Y_1\}, \{X_2,Z_1\}, \{X_2,Z_2\}, \{Y_1,Z_1\}, \{Y_1,Z_2\}, \{Z_1,Z_2\}\}$ and scan the table once to identify the patterns in ℓ_2 that have support ≥ 0.2, called frequent stylometric 2-patterns $L_2 = \{\{X_2, Y_1\}, \{X_2,Z_1\}, \{Y_1,Z_1\}, \{Y_1,Z_2\}\}$. Similarly, we perform L_2 on L_2 to generate L_3 and scan the table once to identify $L_3 = \{X_2, Y_1,Z_1\}$. The finding of each set of frequent κ-patterns requires one full scan of the table.

This property appears to deal with a great problem when facing n-grams. As Apriori algorithm works by identifying the frequent individual items in the database and extending them to larger and larger item sets, the output we get after running the algorithm for the first time is zero. This is a frequency that does not offer information

gain for our research goal, so we have to scan the table twice, after treating zero as a missing variable. The expected outcome after zero-ignorance is most of the times 1, meaning the unique appearance of this n-gram in a text for n times. This corresponds to the true fact of the unique appearance of n-gram features in texts, which might indicate a stylometric preference or a random presence of a specific n-gram . However we keep this piece of information supposing it could solve a difficult discrimination case between authors. Therefore we proceed to a third full scan of the table that indicates some of the most important stylometric features we need for our research, after treating 1 as nonexistent. In some cases we can't move to a further scan after the first one because of the lack of other patterns.

```
Best rules found:

 1. cbg18=2 3 ==> cbg11=2 3    <conf:(1)> lift:(4.75) lev:(0.12) [2] conv:(2.37)
 2. cbg10=3 2 ==> cbg1=2 2    <conf:(1)> lift:(4.75) lev:(0.08) [1] conv:(1.58)
 3. cbg8=3 2 ==> cbg2=2 2    <conf:(1)> lift:(4.75) lev:(0.08) [1] conv:(1.58)
 4. cbg2=3 2 ==> cbg3=3 2    <conf:(1)> lift:(6.33) lev:(0.09) [1] conv:(1.68)
 5. cbg2=3 2 ==> cbg19=3 2    <conf:(1)> lift:(6.33) lev:(0.09) [1] conv:(1.68)
 6. cbg6=4 2 ==> cbg3=3 2    <conf:(1)> lift:(6.33) lev:(0.09) [1] conv:(1.68)
 7. cbg15=2 2 ==> cbg3=3 2    <conf:(1)> lift:(6.33) lev:(0.09) [1] conv:(1.68)
 8. cbg4=4 2 ==> cbg14=2 2    <conf:(1)> lift:(4.75) lev:(0.08) [1] conv:(1.58)
 9. cbg4=4 2 ==> cbg19=2 2    <conf:(1)> lift:(6.33) lev:(0.09) [1] conv:(1.68)
10. cbg4=9 2 ==> cbg9=4 2    <conf:(1)> lift:(4.75) lev:(0.08) [1] conv:(1.58)
```

Figure 8- Frequent n-gram patterns

5.2.1.2 Second phase of model design

After extracting all the frequent patterns it is necessary to filter out the common stylometric frequent n-gram patterns between any two authors "aa" and "ab" where aa ≠ ab .The general idea is to compare every frequent pattern of an author "a_x" with every frequent pattern in all other authors and to remove them from and their frequent patterns respectively if are the same. The remaining stylometic frequent patterns indicate the writing identity of an author.

N-grams don't share a big amount of frequent patterns especially when applying this method on limited dataset. However, we are able to eliminate the characteristics used for identifying the author.

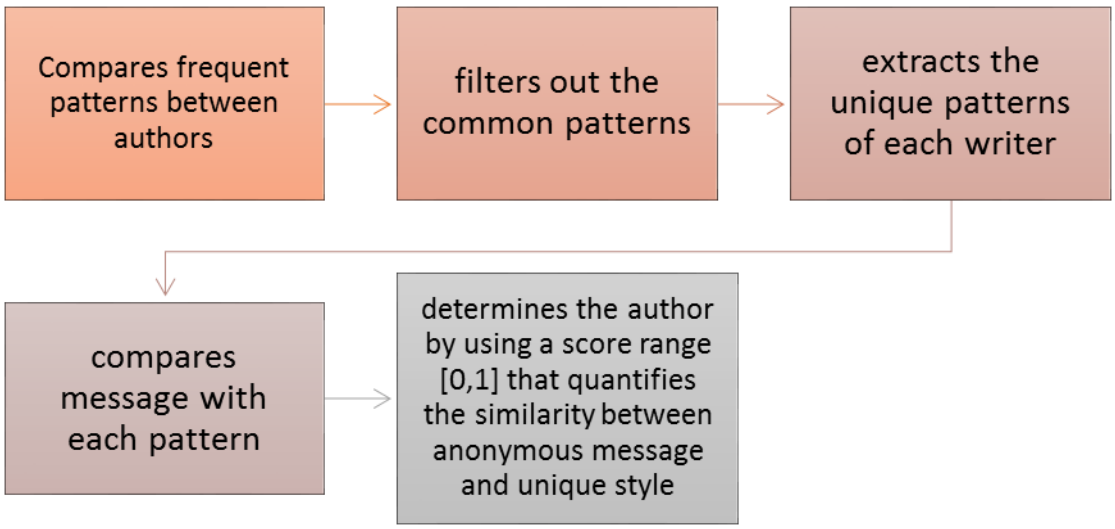
5.2.2. Step 2

At this point we have to face the second subproblem: the identification of the author by comparing the malicious e-mail **m** with each frequent n-gram profile and identify the most similar that maches **m.**

Thus, we have to create an algorithm that is able to:

- filter out all the common stylometric frequent patterns between any two writers who are clearly disparate.
- compare each message with each writing style of every suspect and identify the most similar style to the message in order to determine the author.

Itinerary of the algorithm



5.3. Results of our model

We run the script twice.

Number of authors	Number of documents	Total sum (accuracy)
3	90	70%
20	600	38,9%

We first run the script trying to find the most plausible author between 3 suspects and accuracy reaches almost 70%. When augmenting the number of suspects to 20 the accuracy decreases from 70% to 38, 9%. Still the performance of frequent n-gram patterns seems better than SVM classification. Definitely, the extraction of stylometric features or even a bigger sample could give us more accurate results, but according to the law enforcement unit, having 70%-80% of identification accuracy is acceptable, especially in the early phase of an investigation when a crime investigator often has little clue to begin with.

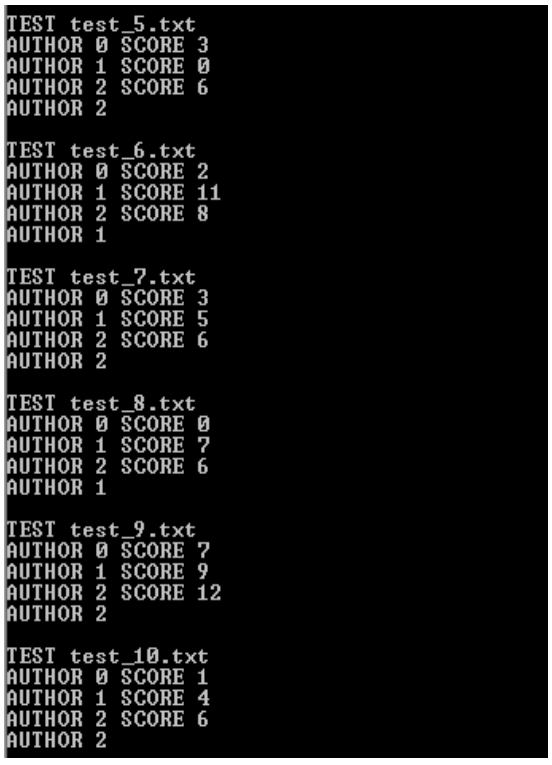


Figure 9- Algorithm results presentation

Chapter 6: Methodological improvements

The main issue of this project was to compare two methods concerning a limited data set and focus on authorship attribution on smaller (3-4) or larger (20) sets of authors. The features extracted were eliminated, which made the procedure more difficult and demanding.

It is already known that the distinctive advantage of the SVM for text categorization is its ability to process many thousand different inputs. This opens the opportunity to use all features in a text directly, which would improve the performance of SVM classification method. This increases dramatically the performance of such a method. Selecting features might be less time consuming, but it rejects a great amount of useful information. Simultaneously, for extracting frequent patterns one needs as more features as possible. The n-gram frequent patterns even though they provide accumulated information, might be proved to be poorer compared to a traditional selection of lexical and syntactic features. Additionally e-mails give the opportunity to measure interesting features such as:

- punctuation after greeting or farewell
- gabs between greeting and main text
- the last-punctuation mark used in every e-mail
- the tendency of an author to use capitalization at the start of a message
- the frequency of upper- or lowercase
- time format (2:00 or 2 o’ clock)

- date format (11/15, 11/2015, 11-2015…etc.)
- use of repetitive words

The grouping of e-mails to formal and informal depending on the person they address to would also be considered as a methodological improvement as the writing style of an author may be different depending on the target recipient (Iqbal et al, 2013). People use different writing styles in their everyday life for various types of communication. This should be taken into consideration, as it could lead to safer conclusions about someone's stylistic preferences of every level.

Furthermore, it must be said that our algorithm can be adjusted according to different purposes, so that we can experiment with every common feature between authors on small or large data and author sets.

Chapter 7: Conclusion and future works

In the present study we tried to apply two methods on a limited dataset after extracting n-grams from our corpus. Both methods have been used in the past, providing both rich and poor results. The complexity level of identification problem is determined by the various parameters like the number of authors and size of training set. This both the parameters play vital role to determine prediction accuracy.

When having a few text sources and a small number of suspects, the data mining method has been proved efficient at least for the start of an investigation. N-gram frequent patterns haven't been used before for e-mail forensics, even though data mining has been heavily applied for computer forensics.

The information extracted from e-mails depends on various unstable factors. We've noticed that digital forensics even need e-mails containing one to four words, which makes unsafe every hypothesis about a precise stylistic preference. Thus, it is still necessary to improve the email authorship identification task by introducing new features or by improving the model so that it may be scalable to an unlimited number of email authors. At present the proposed model is capable for authorship identification, but in future it is expected to extend the model to authorship verification in case if an email belongs to anonymous author, to identify the email as unknown.

References

Abbasi, A., & Chen, H. (2005). Applying authorship analysis to extremist-group Web forum messages. *IEEE Intelligent Systems, 20*(5), 67–75.

Abdallah, E. E., Abdallah, A. E., Bsoul, M., Otoom, A. F., & Daoud, E. Al. (2013). Simplified features for email authorship identification. *International Journal of Security and Networks, 8*(2), 72.

Allocation, L. D., Barber??, P., Based, T. N., Categorization, T., Bash, E., Castro, A., … Lepore, J. (2015). Cross-collection topic models: automatically comparing and contrasting text. *American Printer, 1*(1), 1–5.

Allocation, L. D., Barber??, P., Based, T. N., Categorization, T., Bash, E., Castro, A., … Lepore, J. (2015). Cross-collection topic models: automatically comparing and contrasting text. *American Printer, 1*(1), 1–5.

Banday, M. T. (2011). Techniques and Tools for Forensic Investigation of E- Mail. *International Journal of Network Security & Its Applications (IJNSA), 3*(6), 227–241.

Berberich, K., & Bedathur, S. (2013). Computing n-gram statistics in MapReduce. *Proceedings of the 16th International Conference on Extending Database Technology - EDBT '13*, 101.

Bhaskari, N. S. D. L., & Avadhani, P. S. (2013). Inverted Pyramid Approach for E-Mail Forensics Using Heterogeneous Forensics Tools, (July), 21–23.

Brocardo, M. L., Traore, I., Saad, S., & Woungang, I. (2013). Authorship verification for short messages using stylometry. *2013 International Conference on Computer, Information and Telecommunication Systems, CITS 2013, 2013*(Cits).

Brocardo, M. L., Traore, I., Saad, S., & Woungang, I. (2013). Authorship verification for short messages using stylometry. *2013 International Conference on Computer, Information and Telecommunication Systems, CITS 2013, 2013*(Cits). doi:10.1109/CITS.2013.6705711

Calix, K., Connors, M., Levy, D., Manzar, H., McCabe, G., & Westcott, S. (2008). Stylometry for E-mail Author Identification and Authentication. *CSIS Research Day*, (May), 1–7.

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

Data, G. O. (2014). Description of the Research Used to Generate Our Data, 175–184.

de Vel, O., Anderson, A., Corney, M., & Mohay, G. (2001). Mining e-mail content for author identification forensics. *ACM SIGMOD Record*, *30*(4), 55.

Devendran, V. K., Shahriar, H., & Clincy, V. (2015). A Comparative Study of Email Forensic Tools. *Journal of Information Security*, *06*(02), 111–117. doi:10.4236/jis.2015.62012

Goodman, R., Hahn, M., Marella, M., Ojar, C., & Westcott, S. (2007). The Use of Stylometry for Email Author Identification : A Feasibility Study. *Pace Pacing And Clinical Electrophysiology*, 1–7.

Gupta, G., Gupta, G., Fellow, S. R., Fellow, S. R., Mazumdar, C., Mazumdar, C., … Rao, M. S. (2004). Digital Forensic Analysis of E-Mails: A Trusted E-Mail Protocol. *International Journal of Digital Evidence*, *2*(4), 1–11.

Halvani, O., Steinebach, M., & Zimmermann, R. (2013). Authorship verification via k-nearest neighbor estimation: Notebook for PAN at CLEF 2013. *CEUR Workshop Proceedings*, *1179*.

Halvani, O., Steinebach, M., & Zimmermann, R. (2013). Authorship verification via k-nearest neighbor estimation: Notebook for PAN at CLEF 2013. *CEUR Workshop Proceedings*, *1179*.

Houvardas, J., & Stamatatos, E. (2006). N-gram feature selection for authorship identification. *Artificial Intelligence Methodology Systems and Applications*, *4183*, 77–86.

Hutchison, D., & Franke, K. (2008). Computational Forensics. *New York*, *5158*(August 2009), 122–134–134.

Iqbal, F., Binsalleeh, H., Fung, B. C. M., & Debbabi, M. (2013). A unified data mining solution for authorship analysis in anonymous textual communications. *Information Sciences*, *231*, 98–112. doi:10.1016/j.ins.2011.03.006

Iqbal, F., Hadjidj, R., Fung, B. C. M., & Debbabi, M. (2008). A novel approach of mining write-prints for authorship attribution in e-mail forensics. *Digital Investigation*, *5*(SUPPL.), 42–51.

Iqbal, F., Khan, L. a, Fung, B. C. M., & Debbabi, M. (2010). E-mail authorship verification for forensic investigation. *Proceedings of the 2010 ACM Symposium on Applied Computing*, 1591–1598.

Johnson, A., & Wright, D. (2014). Identifying idiolect in forensic authorship attribution: An n-gram textbite approach. *Language and Law/Linguagem E Direito*, *1*(1), 37–69.

Johnson, A., & Wright, D. (2014). Identifying idiolect in forensic authorship attribution: An n-gram textbite approach. *Language and Law/Linguagem E Direito*, *1*(1), 37–69.

Khan, A. (2012). A simple but Powerful E-mail Authorship Attribution System. *Ipcsit.Net*, *25*, 151–155.

Koppel, M., Schler, J., & Argamon, S. (2008). Computational Methods in Authorship Attribution. *Journal of the American Society for Information Science and Technology*, *60*(1), 9–26.

Koppel, M., Schler, J., & Argamon, S. (2008). Computational Methods in Authorship Attribution. *Journal of the American Society for Information Science and Technology*, *60*(1), 9–26.

Lakshmi, & Pateriya, P. K. (2012). A Study on Author Identification through Stylometry. *International Journal of Computer Science & Communication Networks*, *2*(6), 653–657.

Lalla, H. (2010). E mail forensic authorship attribution. *Methods*, (December). Li, Z. (2013). An Exploratory Study on Authorship Verification Models for Forensic Purpose.

Luiz Brocardo, M., Traore, I., Saad, S., & Woungang, I. (2014). Verifying Online User Identity using Stylometric Analysis for Short Messages. *Journal of Networks*, *9*(12), 3347–3355.

Madigan, D., Genkin, A., Lewis, D. D., & Fradkin, D. (2005). Bayesian multinomial logistic regression for author identification. *AIP Conference Proceedings*, *803*(1), 509–516.

McKeown, K. (2009). N-Grams and Corpus Linguistics.

McKerlich, R., Ives, C., & McGreal, R. (2013). Measuring use and creation of open educational resources in higher education. *International Review of Research in Open and Distance Learning*, *14*(4), 90–103.

Mikros, G. K. (2013). Systematic stylometric differences in men and women authors: a corpus-based study. *Issues in Quantitative Linguistics 3. Dedicated to Karl-Heinz Best on the Occasion of His 70th Birthday*, 206–223.

Mikros, G. K. (2013). Authorship Attribution and Gender Identification in Greek Blogs. *Methods and Applications of Quantitative Linguistics*, 21–32.

Mikros, G. K., & Perifanos, K. (2011). Authorship identification in large email collections: Experiments using features that belong to different linguistic levels Notebook for PAN at CLEF 2011. *CEUR Workshop Proceedings*, *1177*.

Ranjan, N., & Prasad, R. . (2013). International Journal of Research in Advent Technology AUTHOR IDENTIFICATION IN TEXT MINING International Journal of Research in Advent Technology, *1*(5), 568–571.

Sharma, A., & Scholar, M. T. (2016). A Novel Method for Detecting Spam Email using KNN Classification with Spearman Correlation as Distance Measure, *136*(6), 28–35.

Stamatatos, E. (2013). on the Robustness of Authorship Attribution Based on Character N-Gram Features. *Journal of Law & Policy*, 421–439.

Vel, O. De, Corney, M., & Anderson, A. (2002). Language and gender author cohort analysis of e-mail for computer forensics. *Digital Forensics Research Workshop*, 1–16.

Weiss, S. M. (1998). Text Mining with Decision Trees and Decision Rules C. Apte, F. Damerau, and S.M. Weiss Conference on Automated Learning and Discovery Carnegie-Mellon University, June 1998. *Computer*, (June).

Witschel, H. (2005). Using decision trees and text mining techniques for extending taxonomies. *Learning and Extending Lexical Ontologies by Using Machine Learning Methods.*

Zhang, S., Yang, H., & Singh, L. (2014). Increased information leakage from text. *CEUR Workshop Proceedings*, *1225*(July 2003), 41–42.

Appendix

Source code

```
import re
import sys
listoflists = []
wpe = []

inc = 0
prev = ""
a_list = []
enter = 0
start = 0
if len(sys.argv) == 1:
    print "USAGE miner name_of_source test"
else:


   source = sys.argv[1]
   totaltests = len(sys.argv) - 2
print "source: %s NO_TESTS: %d" % (source, totaltests)



with open("%s" % source) as f:
   for line in f:
```

```
        if prev != ''.join(line[0] + line[1]):



            if start == 0:
                start = 1
            else:
                enter = 1

#               print ("NEW LIST")

        name = line.split()

        a_list.append(name[1])

        prev = ''.join(line[0] + line[1])
       # print("prev %s lin %s %d " % (prev, ''.join(line[0] + line[1]), inc))

        if enter == 1:
        # print ("APPEND LIST")
            a_list.pop()
        #    print a_list
            listoflists.append(a_list)
            inc = inc + 1
            a_list = []
            a_list.append(name[1])
            enter = 0
# print len(listoflists)
a_list.pop()
a_list.append(name[1])
```

```
# print a_list
listoflists.append(a_list)

# print listoflists[9]
# print inc
print ("TOTAL AUTHORS %s"%(inc+1))
we = []
for c in range(0, inc + 1):
    toberemoved = []
    for e in listoflists[c]:
        rem = 0

     #  print "OK "
     #   print c
     #   print listoflists[c]

        for d in range (c + 1, inc + 1):
            rem2 = 0
            for f in listoflists[d]:



                if e == f:

#                   print e
#                   print f
#                   print c
#                   print d
                  if e not in toberemoved:
                     toberemoved.append(e)
```

```
                listoflists[d].remove(f)




    for rrr in toberemoved:
      #  print rrr

        listoflists[c].remove(rrr)
   # print c
    we = listoflists[c][:]
    wpe.append(we)
   # print we

for ee in range (0, inc + 1):
    todel = []
    toins = []
    for ww in wpe[ee]:
        name = re.sub('=[0-9]*', '', ww)
#         print ww
#         print name
        todel.append(ww)
        # wpe[ee].remove(ww)
        toins.append(name)

    # print "first"
   # print ee
  #  print wpe[ee]
```

```
    for ins in todel:
        wpe[ee].remove(ins)
    for ins in toins:
        wpe[ee].append(ins)




   # print wpe[ee]

#for ii in range (2, len(sys.argv)):
print "\nTEST %s" % (sys.argv[2])
cf= []
with open("%s" % sys.argv[2]) as f:
    cc = 1
    for line in f:
        if cc == 1:
            ct = line.split()
            cc = 2
        else:
            cf= []
            cf = line.split()

#     print ct[3]
#     print cf[3]
            print cf


            hscore = -1
```

```
# print len(cf)
    if cc==2:
        for ca in range(0, inc + 1):
            em = 0
            score = 0
            for c in range(0, len(ct)):
                if cf[c] != "0":
                # print ct[c]
                # print wpe[ca]

                    if ct[c] in wpe[ca]:
                        score = score + 1
                else:
                    em = em + 1
                 #  print "LUS^O%S"


            #print "AUTHOR %d SCORE %d" % (ca, score)


            if score > hscore:
                hscore = score
                author = ca


        print "AUTHOR %d" % (author)
```