AUTHORSHIP RECOGNITION IN ONLINE SOCIAL PLATFORMS

by

Rıdvan Salih Kuzu

B.S., Electrical and Electronics Engineering, Boğaziçi University, 2010

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in System and Control Engineering
Boğaziçi University
2017

ACKNOWLEDGEMENTS

This thesis is supported by the Scientific and Technological Research Council of Turkey (TUBITAK) with project number 114E481 and by the Turkish Ministry of Development under the TAM Project number DPT2007K120610.

Apart from this, I would like to express my deepest gratitude to my thesis supervisor Albert Ali Salah for his insight, endless patience and stimulation that I needed to complete this work. Without his supervision and efforts, this work may not have seen the light of day.

I thank my friends and colleagues who do not lose faith in me about completing my master degree sooner or later.

The greatest thanks is to my family for their encouragement, endless support and trust. I am specifically grateful to my lovely wife, Zeynep Oba, who believed in me with boundless patience throughout this thesis.

ABSTRACT

AUTHORSHIP RECOGNITION IN ONLINE SOCIAL PLATFORMS

Biometrics is the identification of a person by personal properties and traits, and can be divided into physiological based and behavioural based methods. In this thesis we investigate the identification of users of a social platform from their verbal behaviour, which is an example of behaviour based biometrics. Online social platforms implement moderation mechanisms to filter out unwanted content and to take action against possible cases of verbal aggression and abuse, sexual harassment, and such. Since they can have large numbers of users, it is desirable to automatize parts of this process. What we call chat biometrics aims to re-identify a user from chat messages. The typical application scenario is the re-identification of banned users, returning under different identities, and aggressors operating through multiple fake accounts. We propose a processing pipeline, and contrast the problem with the authorship identification problem, which is well-studied in the literature. We evaluate our proposed approach on a large corpus of multiparty chat records in Turkish (namely, the COPA database), which was collected from a multiplayer game environment. We also introduce a new corpus in this study, collected from a well-known Turkish social platform called Ekşisözlük, in order to test the robustness of the system across domain changes, as well as on Portuguese and English news datasets, to show performance across languages. We evaluate both profile-based and instance-based approaches, and provide detailed analyses with regards to the required amount of text to identify a person reliably.

ÖZET

ÇEVRİMİÇİ SOSYAL PLATFORMLARDA YAZAR TANIMA

Biyometri bir kişinin tutum ve özelliklerine bağlı olarak kimliğini tespit etme işlemidir ve fizyolojik ve davranış temelli olmak üzere ikiye ayrılmaktadır. Bu tezde, davranış temelli bir biyometrinin örneği olarak, kişilerin kimliğini sosyal platformlardaki yazım alışkanlıklarından tespit etmeye çalışmaktayız. Çevrimiçi sosyal platformlar, istenmeyen içeriği filtrelemek için denetleme mekanizmalarını uygular ve sözlü saldırı, istismar, cinsel taciz gibi durumlara karşı harekete geçmeye çalışır. Burada biyometri olarak adlandırdığımız şey, bir sosyal platformda engellenen kullanıcıların farklı kimlikle geri dönmesi durumunda kimliğinin tespit edilmesi ya da sahte hesapların ardındaki kişilerin ortaya çıkarılmasıdır. Bu amaçla bir kimlik tanıma sistemi ortaya koyarak literatürde yaygınlıkla işlenenen diğer biyometri yöntemleri ile karşılaştırmaktayız. Ortaya koyduğumuz biyometrik kimlik tanıma yaklaşımı, COPA olarak adlandırılan ve çevrimiçi bir oyun platformundan toplanmış olan ikiden fazla kişinin çevrimiçi grup sohbetlerini içeren bir Türkçe veritabanında ölçümlenmektedir. Önerdiğimiz kimlik tanıma yönteminin farklı sosyal mecralarda da dayanıklılığını ölçümlemek için Ekşisözlük adlı Türkiye'de yaygın bilinirliği olan sosyal bir platformdan da veri toplamış bulunuyoruz. Ayrıca, önerilen yöntemin farklı dillerdeki kimlik tanıma başarımını ölçümlemek amacıyla İngilizce ve Portekizce haber kayıtlarını da kullanmaktayız. Bu içerikler üzerinde, hem genel profil bilgisini hem de yazı örneklerini ayrı ayrı ele alarak modellediğimiz kimlik tanıma sisteminde bir kişiyi güvenilir şekilde tespit etmek için en az ne kadar yazı içeriğine ihtiyacımız olduğunu da araştırmaktayız.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$\boldsymbol{A}$	Author corpus
$\boldsymbol{a_i}$	Author i in a given author corpus
$\boldsymbol{a_{test}}$	Unknown author to be predicted
$\boldsymbol{d}$	Document
$\boldsymbol{d_{i,j}}$	j^{th} document of author i
$\boldsymbol{d}_j^{\phi}$	j^{th} weighted document in term-document matrix
$f_{i,j}$	Frequency of term i in document j
h_i	Local histogram extracted from document i
K	Kernel smoothing function
$\boldsymbol{q}$	Term weighted document query from an unknown author
$P(x)$	Profile of an author x
t	Term
t_i	Term i in term dictionary
$\boldsymbol{V}$	Vocabulary (Term dictionary)
W_i	Terms of document i in order of appearance
α	Mixing coefficient in ELM
$\boldsymbol{\Gamma}$	Local weighting scheme
$\boldsymbol{\Theta}$	Global weighting scheme
λ	Number of hidden layers in ELM
$\Phi_{i,j}$	Weighted frequency of term i in document j
$\boldsymbol{\Phi}$	Weighted term-document frequency matrix
ω	Width coefficient in ELM

LIST OF ACRONYMS/ABBREVIATIONS

AR	Authorship Recognition
AD	Authorship Discrimination
ADF	Augmented Doppelgänger Finder
AP	Authorship Profiling
AV	Authorship Verification
BD	Bhattacharya Distance
BMR	Bayesian Multinomial Regression
CE	Cross Entropy
ChF	Character Features
CMC	Cumulative Match Score
CNG	Common N-Grams
CsF	Content Specific Features
DA	Discriminant Analysis
DADT-P	Probabilistic Attribution with Author-Document Topic Model
DT	Decision Trees
ED	Euclidian Distance
ELM	Extreme Learning Machine
EM	Expectation Maximization
FN	False Negative
FP	False Positive
GFR	Global Frequent Ranking
GI	General Impostor
IAF	Inverse Author Frequency
IBA	Instance Based Approach
ICA	Independent Component Analysis
ID	Authorship Intent Determination
IDF	Inverse Document Frequency
KLD	Kullback-Leiber Distance

KNN	K-Nearest Neighbour
LDA	Linear Discriminant Analysis
LDR	Local Distinctive Ranking
LFR	Local Frequent Ranking
LOWBOW	Locally Weighted Bag of Words
LSA	Latent Semantic Analysis
MC	Markow Chains
MLP	Multilayer Perceptron
NB	Naive Bayesian
NLP	Natural Language Processing
NMF	Non-negative Matrix Factorisation
NN	Neural Network
PBA	Profile Based Approach
PCA	Principal Component Analysis
PD	Profile Dissimilarity
POS	Part of Speech
PPCA	Probabilistic Principal Component Analysis
PRIM	Patient Rule Induction Method
RBF	Radial Base Function
RF	Random Forest
RKHS	Reproducing Kernel Hilbert Spaces
RLP	Recentered Local Profile
RM	Regression Model
SCAP	Source Code Author Profiling
SeF	Semantic Features
SM	Similarity Measure
StF	Structural Features
sTF	Sublinear Term Frequency
SVD	Singular Value Decomposition
SVM	Support Vector Machine
SyF	Syntactic Features

TF	Term Frequency
TN	True Negative
TP	True Positive
VSM	Vector Space Model
WAN	Word Adjacency Networks
WoF	Lexical Word Features

1. INTRODUCTION

1.1. What is Biometrics?

Biometrics refers to the automatic recognition of people, or determination of their particular features extracted from physiological and/or behavioural characteristics to distinguish them to some extent [1]. In this thesis we investigate the identification of users of a social platform from their verbal behaviour, which is an example of behaviour-based biometrics.

Biometric systems operate in two modes: either confirming (verification) or determining (recognition) the identity of an individual. While verification means comparison of a template acquired from biometric information of an unknown user with the claimed identity, recognition means comparison of all the templates with all users in the database. For that reason, verification and recognition are two separate problems, which should be handled individually.

According to Delac and Grgic [2], a biometric system contains four fundamental elements: (i) data acquisition module to collect biometric information; (ii) feature extraction module in which feature vectors are extracted by means of processing the acquired information; (iii) matching module in which comparison of feature vectors against templates is conducted; (iv) decision making module where a claimed identity is verified or the user's identity is set.

Any behavioural or physiological attitude can represent biometric characteristics if it meets the following requirements: (i) everyone should have it - universality; (ii) two feature from different individuals should not be the same - distinctiveness; (iii) it should be invariant over a given period of time - permanence; (iv) acquisition of it should be easy - collectability.

In case of real life applications, additional three factors should be also taken into account: (i) it should be potent enough against fraudulent actions; (ii) it should not harm users; and (iii) it should be inline with speed, accuracy, infrastructure limitations.

1.2. Authorship Problem in Online Data

Computer-mediated communication with text messages has become very prevalent with the rise of the Internet. Instant messaging applications on mobile devices such as WhatsApp, Line, Viber, Skype, SnapChat have received widespread attention, thousands of multiplayer online games and dedicated platforms provide chat facilities. These Internet based services constantly generate large amount of text data, which can be processed by applications of sentiment analysis and user analytics. The informal nature of these texts, their unordered structure, and the large amount of spelling mistakes bring additional challenges to the typical natural language processing based analysis.

In this thesis, we try to identify users of a social platform by their text contributions, and in particular by their chat records. The typical application scenario in this thesis is the re-identification of banned users of a social platform, returning under different identities, as well as aggressors operating through multiple fake accounts.

While content and style of text messages depend on many factors, it may be possible to match an unsuspected person by using a pre-collected corpus. It may also be possible to deduce gender, age, and ethnicity, based on the specific words and forms used during chat, and based on particular mistakes. Writing style is unique for everybody, and some identity-related cues remain even if the individual consciously attempts to change the writing style [3]. This issue was investigated in the context of authorship recognition, which seeks to identify the author of a piece of text from among a set of candidate authors, whose texts are available for supervised classifier training. The electronic chat domain is significantly different from the literary text domain. These differences are particularly prominent in word and character frequencies, use of punctuation marks, intentional and unintentional misspellings, vocabulary usage, sentence length, and the particular ordering of words. The increased freedom in the usage of language, coupled with (typically) much more limited vocabulary makes chat biometrics an interesting challenge.

In this thesis, we use data acquired from the chat interface of a multiplayer online game, together with meta-data concerning more than a thousand complaints filed by players [4,5]. In such games, some users who are blocked by administrators for various reasons (such as cheating, foul language, hate speech, abusive behaviours) may return to the game using an impostor account. Finding these matching accounts is a very hard problem to tackle manually. Game communities spend resources to preserve a user friendly gaming environment, which includes offending players. Reducing the number of suspects might be very useful, even if finding the real offender is difficult.

We investigate the rate of success in identifying these malicious users in multi-participant chat environments by means of extracting relevant features and supervised classification techniques. In our approach, we apply and compare several methods to match users to a gallery by their chat records. In the literature, methods developed for matching personal text content have been mostly evaluated with Indo-European languages. We test our approach with documents that have Turkish chat content, which bring additional challenges due to the agglutinative nature of the language (i.e. many postfixes can be applied on word roots). Text analysis in agglutinative languages includes a normalisation step to isolate the roots of the words, which we additionally assess in the context of chat biometrics.

1.3. Motivation and Contribution

In biometrics, sufficiently high accuracy rates have been achieved during controlled experiments. Nevertheless, robustness, dependability and convenience are still major issues for real applications outside controlled settings. In a similar way, authorship identification is a developing research area, moving from initial linguistic studies to concurrent progresses in text information retrieval.

The work introduced here is closely related to “chat mining” [6], where comments and discussions from online social platforms are mined for specific purposes, such as identifying the unknown author of a post among suspects. We are motivated in this work by moderation mechanisms implemented for online social platforms, for which it is essential to filter out unwanted content and to take action against possible cases of verbal aggression and abuse, sexual harassment, and such [4]. Automatic evaluation of aggression cases is typically performed via user profiling, and the textual content of interaction is not processed [4, 5]. In this work, we propose a text-based system for monitoring the platform for repeated offenders.

In particular, we study the following questions:

- What is the effectiveness of existing author identification methods for attributing authorship of a set of chat texts to one person among a closed set of suspects?
- What are the influential and effectual features of chat messages for the purposes of chat biometrics?
- How much text is required to extract an accurate author profile?
- How can we improve the author recognition performance?

We extend the literature in a number of ways: (i) measuring effects of dictionary size on performance, (ii) comparing weighting schemes to be used in authorship analysis, (iii) suggesting a novel recognition pipeline which gives better results than state-of-the-art baselines, (iv) comprehensive literature comparison with concurrent studies in chat biometrics, and (v) measuring the effects of domain changes in regards of intra-language and inter-language variations. We have presented the results of this thesis in several venues [7, 8].

1.4. Outline

This thesis is structured as follows. In Chapter 2 we provide an extensive survey for this relatively new domain. Chapter 3 describes baseline approaches and the proposed methodology for identifying people from their online writings. Chapter 4 explains some authorship recognition approaches re-implemented for comparing the proposed methodologies. We introduce one novel and three existing databases used in experimental evaluations of this study in Chapter 5. Chapter 6 provides the experimental results, and a discussion thereof. Finally, Chapter 7 concludes the study by highlighting the main findings.

2. RELATED STUDIES

In this part, studies related with authorship analysis are reviewed. Based on features types, analytical techniques, language issue, and other related parameters, common authorship analysis methodologies are summarised.

2.1. Authorship Analysis

Authorship analysis aims to identify individuals by the statistical properties and characteristics of their language use. To distinguish text written by different authors, textual features, as well as machine learning techniques can be employed. Gray *et al.* identified several approaches of authorship analysis that can be applied to software forensics [9]. Based on their definitions and other related studies, authorship analysis can be grouped into five major categories, as summarised in Table 2.1.

Table 2.1. Types of authorship analysis in summary

Category	**Description**	**Label**
Authorship Recognition	Uses a training set of different authors' writings to determine the likelihood of authorship on a new piece of writing.	AR
Authorship Verification	Uses a set of documents by an author, determines whether a new document is also by that author or not.	AV
Authorship Profiling	Determines an author profile by summarising features obtained from the works of the author.	AP
Authorship Discrimination	Given a document or a corpus, decides whether it is written by a single author or by multiple authors without actually identifying who they are.	AD
Author Intent Determination	Seeks certain intentionally produced properties of a given document or corpus, including style.	ID

Authorship attribution simply aims at determining the author of a document. To achieve this purpose, one compares a query text with a model of the candidate author and determines the likelihood of the model for the query. One of the early attempts on authorship analysis was performed by Mendenhall [10]. Mendenhall looked into word lengths, comparing Dickens, Thackeray, and Mill, and decided that a hundred thousand words were enough to determine a signature for an author. The most curious series of works in this area were realised by Mosteller and Wallace in the 60s [11]. In their study of the authorship attribution on 146 political essays (known as the Federalist Papers), a Bayesian approach on small word frequencies was applied, and promising results were obtained. In general, their outcome was accepted by historical scholars and became a milestone in this research area [12, 13].

Authorship attribution can be considered in two different categories: (i) authorship recognition, and (ii) authorship verification:

In the recognition mode, a script from an unknown author is compared with all the authors' textual records for a match. It is a one-to-many comparison to detect identity of an author without a pre-claim an identity, and the identification attempt should fail unless the author is enrolled in the database before [1].

In the verification mode, the system validates an author's claimed identity by comparing the captured textual data with his/her previously stored scripts. Hence, implementing an author verification system typically necessitates: (i) building a response function based on the features extracted from the query text for a given author, (ii) setting a threshold value to determine if the query text belonged to the author in question [14].

Authorship profiling or characterisation aims to appoint the writings of an author into a set of categories by regarding the author's social and linguistic properties. Some of the properties previously examined in the literature are gender, educational and cultural background, and language familiarity.

Authorship discrimination is aiming to determine if a document or a corpus is written by a single individual, or by multiple authors. Inter-subject and intra-subject variability of a text are computed in such problems to determine the validity of the claim that two texts belong to different authors without regarding their identities. Many studies in this field are also related with plagiarism detection. Plagiarism encloses partial or holistic replication of a piece of work and plagiarism detection is used for investigating suspicious documents against potential original documents [15].

Author intent determination, as initially defined in the code domain aimed to detect intentionally malicious code [9]. In a biometric setting, it can refer to the detection of any stylistic or content-related property of a document produced by the author.

These problems can be adopted to a chat setting, where the system typically has access to some additional profile information about the user, but has no guarantee of the correctness of this information. While such features can be beneficial in authorship analysis, we do not tackle them in this work. The next section describes the most commonly used features in the literature.

2.2. Feature Types

The state-of-art approaches in authorship analysis depend on stylometric features, which can be divided into six major categories: (i) lexical word, (ii) character, (iii) syntactic, (iv) structural, (v) content specific, and (vi) semantic features, respectively. A brief description and the relative discrimination capability of each type of feature are given next.

Lexical word features are used to learn about the preferred use of words of an individual. The pioneer efforts on attributing authorship were based on trivial measures like counts of sentence length and word length [12]. The use of such features points out the tendency of an individual to use particular words or phrases. One of the most important benefits of lexical word features is that they can be easily performed on any language and any textual database without the need for additional tools apart from a tokenizer which divides texts into tokens (e.g., words, characters). Nevertheless, it can be hard in logographic writing systems like Chinese.

Character features get extensive attention in the literature to represent textual data as shown in Table 2.3. Wide range of character-level features exist including frequency of individual symbols in the alphabet, total number of upper/lower case letters, distribution of capital letters used in the beginning of sentences, average number of characters per word, and average number of characters per sentence, etc. [13]. In this domain, extracting frequency distribution of character n-grams (i.e. strings of length n) is more comprehensive and also computationally simple procedure. Additionally, this method is capable of catching style nuances along with lexical information, (e.g., $|\_in\_|$, $|text|$), contextual marks, (e.g., $|in\_t|$), as well as punctuation and capitalisation preferences, etc. Another important aspect is that, representation of the text in the n-gram domain is more robust to noise, compared to lexical word representations. This is an important point especially for the chat domain that we tackle in this thesis, as chat messages are very noisy, and a single word can have many different alternatives.

Syntactic features, such as part-of-speech (POS) tags, chunks, sentence and phrase structure, can capture writing style of an author at the sentence level. The discriminating power of syntactic features is extracted from people's different attitudes while organising sentences, which can be a cue to detect authorship. While function words do not contribute much to semantics, they describe relationships between content words, and their distribution and specific usage can be informative [16].

Structural features depend on distinctive habits of people while organising a document, such as paragraph length, use of indentation, and use of signature. These features are also prominent in online documents, which have less content information, but more flexible structure or rich stylistic information. Structural features were first suggested by de Vel *et al.* for e-mail authorship attribution and led to high identification performance [17].

Content-specific features are preferred to represent textual data via limited key words or terms if they commonly exist in some specific activities, discussion and/or social groups. To illustrate, messages aiming cybercrime activities like fraudulent sale offers, spamming and phishing frequently have phrases containing slang or street words [13]. Generally in this case, extracted features for a specific domain cannot be directly applied into other domains.

Semantic features are used for determining semantic resemblances among words and phrases with the assistance of linguistic analysis. Synonyms, antonyms, hyponyms and hypernyms of the words are prevalently preferred to exploit semantic information. Some low level attempts like partial parsing, sentence splitting, part-of-speech tagging etc. can be undertaken via NLP tools for the purpose of extracting semantic features. On the other hand, sophisticated processes like pragmatic and semantic analysis, syntactic parsing etc. cannot be probed in detail yet. In other words, extracting complex semantic features have been rarely attempted until now [12].

Based on the previously described feature categories, some of the most commonly used feature types to represent a text for authorship analysis are listed in Table 2.2.

If we consider simplicity and language independence as primary factors, lexical features are expected to perform better than other features. Especially, the character n-gram representation has been used as one of the most effective measures of authorship attribution [16,18]. If authors tend to use similar patterns in their writings, this would imply that syntactic and semantic features may lead to superior results. On the other hand, language-specific NLP tools like part-of-speech taggers, stemmers, spell checkers etc. are needed to exploit these features.

2.3. Analysis Techniques

A typical authorship recognition problem contains a set of text samples for candidate authors, and query text samples from unknown authors. Each sample should be attributed to a candidate author. Identification approaches can be distinguished as profile-based and instance-based, according to whether the set of text samples for each author is treated individually, or cumulatively [12].

Table 2.2. Commonly used features for authorship analysis

Lexical Word Features **(WoF)**	**Character Features** **(ChF)**	**Structural Features** **(StF)**
-total # of words -total # of unique words -ratio of short words -mean word length -ratio of distinct words -# of hapax legomena -# of hapax dislegomena -word n-grams -skip-grams -word frequencies -# of words of each length -vocabulary richness	-total # of characters -ratio of alphabetic chars. -ratio of upper case letters -ratio of digit characters -ratio of white space chars. -ratio of tab space chars. -ratio of special chars. -ratio of emoticons -ratio of char. repetition -character n-grams -vowel combination -compression methods	-# of sentences -# of paragraphs -# of quoted content -# of lines -# of characters per paragraph -# of words per paragraph -# of sentences per paragraph -farewells -greetings -indentations -signature
Syntactic Features **(SyF)**	**Content Specific Features** **(CsF)**	**Semantic Features** **(SeF)**
-freq. of function words -freq. of punctuation marks -part of speech (POS) tags -total # of line -total # of sentences -ratio of spelling errors	-# of stop words -# of abbreviations -# of keywords -gender/age based words -slang words -writing speed -turn duration (for chat)	-synonyms of words -hypernyms of words -hyponyms of words -semantic dependency graphs -latent semantic analysis -systemic functional grammar

Concatenating training texts per author in one single text file is known as the profile-based approach (PBA). This large single file is used to extract properties of the author's style. A text sample from an unknown author is compared with each author profile, and a distance measure is used to find the most likely author. In this approach, features related with the variety of texts in the training corpus are not taken into consideration.

Instance-based approach (IBA), on the other hand, considers each text sample independently, hence the differences in the training texts by the same author are not neglected. Both approaches have their own advantages, but if text documents are very concise and limited, concatenation of the text (as in profile-based approaches) may help to create a sufficiently long document for capturing the author's style [19]. Instance-based approaches are believed to be more effective when sufficient amount of text per author is available. However, Potha and Stamatatos [20] reported the best results on the PAN-2013 Authorship Analysis Competition with a profile-based approach.

Performance in this domain also depends on pre-processing techniques, document set sizes, weighting schemes, language characteristics, and feature sets. In terms of used features, character n-grams, word tokens, distribution-based similarity features are typically preferred. Some common identification and attribution approaches in terms of feature extraction and matching methods are summarised in Table 2.3.

2.4. Chat Biometry Case

2.4.1. Non-agglutinative Languages

The bulk of authorship analysis approaches in the literature focus on the English language, and there are a few important studies related to chat biometrics on texts in English. Inches *et al.* [40] used two different internet relay chat (IRC) datasets containing homogeneous and heterogeneous topics separately. Traditional chi-squared distance and KLD were used to determine the similarity between the author profiles. The study achieved up to 61% accuracy on heterogeneous chat records. Layton *et al.* [51] used IRC records of 50 users (50 chat messages for each). By applying an ensemble classification approach, in which distance between the closest and second closest authors is used to weight each classification, 55% identification accuracy was achieved via the recentered local profile (RLP).

Roffo *et al.* [42] proposed to adopt features inspired by conversation analysis (specifically for turn-taking), as well as to derive the features from separate turns instead of entire conversations. The corpus used for their study contains 312 dyadic Italian chat conversations, collected via Skype over a time span of 5 months. They report a recognition rate of 76.9% on a total of 78 subjects. Their approach does not take the actual content into consideration, but focuses on different features for analysis, such as character writing speed, total chat time, or other features of temporal nature. These features are typically not stored by software that handle chat records. Subsequently, chat based biometrics can be extended to behavioral biometrics, only if special software requirements are met. In this thesis, we focus on the much more common case where such information is discarded.

2.4.2. Agglutinative Languages

Text analysis can be significantly different on agglutinative languages such as Hungarian, Finnish and Turkish. Although research on highly inflected languages like Greek and Sanskrit, or fusional languages like German may be useful in order to understand language dependent approaches on chat biometrics to some extent [19,21,46], agglutinative languages differ from these by a complex word structure, which is formed by stringing together morphemes without changing them in spelling or phonetics. In Turkish, for example, the word "ev-ler-iniz-den" would translate to "house-plural-your-from", i.e. "from your houses". This causes difficulties in stemming and in syntactic analysis, particularly for noisy text obtained from social media. Moreover, transposed sentence structure is very common and word order in a sentence is so flexible without changing the meaning. For instance, the sentence, "ben eve geldim" ("I came home"), has only 3 words and can be expressed with 6 different word orders in Turkish. Hence, even if the same words are used, ordering habits can give some hints about its author.

There are some prior studies on authorship attribution in Turkish. Tufan *et al.* used style marker features on a gallery of 20 authors [29]. Amasyali *et al.* categorised texts in terms of author (18-class), genre (3-class) and gender (2-class) by using stylistic and n-gram features [24]. Hence, Tufan *et al.* and Amasyali *et al.* obtained success rates of 80% and 83.3% respectively in author recognition by using default Naive Bayes classifier of WEKA. Both studies used newspaper articles, but neither the gallery size, nor the domain characteristics are representative for chat biometrics. Moreover, contrary to our current work, they consist of some language dependent features.

In another study, a chat mining framework was tested on a Turkish dataset containing peer-to-peer text messages [6]. This work is one of the most exhaustive efforts on chat biometrics in Turkish, and while it does not cover multiparty chat, it established that context plays a significant role on vocabulary use and writing style in peer-to-peer communications. The authors reported that term-based features achieved better results compared to style-based features on a 100-author problem.

Table 2.3. Summary of studies in authorship analysis

Previous	Category				Features						Techniques				# of
Studies	AR	AP	AV	AD	WoF	ChF	SyF	StF	CsF	SeF	PBA	IBA	Detail	Language	Subjects
Stamatatos *et al.*'00 [14]	✓	✓	✓		✓		✓					✓	RM,DA	Greek	10
De Vel *et al.*'01 [17]	✓			✓		✓	✓	✓				✓	SVM	English	63
Kešelj *et al.*'03 [21]	✓					✓					✓		CNG,PD	Multiple	10
Clough'03 [22]				✓		✓						✓	SM	English	N/A
Amasyalı & Diri'03-06 [23, 24]	✓	✓				✓						✓	NB,SVM,MLP-RBF	Turkish	18
Zhao *et al.*'06 [25]	✓						✓					✓	KLD,SVM,NB	English	7
Zheng *et al.*'06 [13]	✓				✓	✓	✓	✓	✓			✓	DT,NN,SVM	Multiple	20
Juola'06 [16]	✓					✓					✓		CE	Multiple	13
Sanderson & Guenter'06 [26]			✓		✓	✓						✓	MC	English	50
McCarty et. al.'06 [27]	✓									✓		✓	Coh-Metrix	English	3
Frantzeskou *et al.*'07 [28]	✓					✓					✓		SCAP	C++ / Java	8
Tufan & Görür'07 [29]	✓				✓		✓					✓	NB	Turkish	20
Meyer zu Eissen *et al.*'07 [15]				✓	✓		✓					✓	DA	English	4
Estival *et al.*'07 [30]		✓				✓	✓	✓				✓	SVM,RF,DT	English	1,033
Küçükyılmaz *et al.*'08 [6]	✓	✓			✓	✓	✓					✓	KNN,NB,SVM,PRIM	Turkish	100
Argamon *et al.*'09 [31]		✓			✓		✓					✓	BMR	Multiple	19,320
Koppel *et al.*'11 [32]	✓					✓					✓	✓	NB	Multiple	10,000
Solorio *et al.*'11 [33]	✓				✓	✓	✓				✓		SM	English	100
Escalante *et al.*'11 [34]	✓					✓				✓		✓	LOWBOW,SVM	English	10
Oliveira *et al.*'12 [35]	✓					✓					✓		NCD	Portuguese	100
Layton *et al.*'12 [36]	✓				✓	✓	✓	✓			✓		RLP,CNG,SCAP,PD	Multiple	13
Savoy'12 [37]	✓				✓						✓		Z-Score	Multiple	20
Cristani et al.'12 [38]	✓				✓	✓	✓		✓		✓		BD, ED	Italian	77
Seidman'13 [39]			✓								✓		GI	Multiple	20
Inches *et al.*'13 [40]	✓				✓		✓				✓		KLD, Chi-Square	English	1,502
Monaco *et al.*'13 [41]			✓		✓	✓			✓			✓	KNN	English	30
Roffo *et al.*'13 [42]	✓		✓		✓	✓	✓		✓		✓		RKHS	Italian	78
Brocardo *et al.*'13 [43]		✓				✓					✓		SM	English	87
Iqbal *et al.*'13 [44]	✓	✓			✓	✓	✓		✓			✓	EM, K-Means	English	150
Rappoport *et al.*'13 [45]	✓				✓	✓						✓	SVM	English	1,000
Mikros *et al.*'13 [46]	✓				✓	✓					✓	✓	SVM	Greek	10
Portha & Stamatatos'14 [20]			✓		✓						✓		CNG	Multiple	20
Qian *et al.*'14 [47]	✓				✓	✓	✓					✓	CNG,SVM,RM	English	62
Seroussi *et al.*'14 [48]	✓	✓			✓					✓		✓	DADT-P	English	72
Segarra *et al.*'15 [49]	✓	✓	✓	✓			✓				✓		WAN	English	21
Overdorf & Greenstadt'16 [50]	✓				✓	✓	✓					✓	RM,ADF	English	100

Abbreviation List of Techniques Used in the Literature

BD	Bhattacharya Distance	**KNN**	K-Nearest Neighbour	**RF**	Random Forest
BMR	Bayesian Multinomial Regression	**MC**	Markow Chains	**RKHS**	Reproducing Kernel Hilbert Spaces
CNG	Common N-grams	**MLP**	Multilayer Perceptron	**RLP**	Recentered Local Profile
DA	Discriminant Analysis	**NB**	Naive Bayesian	**RM**	Regression Model
DT	Decision Trees	**NN**	Neural Network	**SCAP**	Source code author profiling
ED	Euclidian Distance	**PD**	Profile Dissimilarity	**SM**	Similarity Measure
EM	Expectation Maximisation	**PRIM**	Patient Rule Induction Method	**SVM**	Support Vector Machine
GI	General Impostor	**RBF**	Radial Base Function	**WAN**	Word Adjacency Networks
KLD	Kullback-Leiber Distance	**DADT-P**	Probabilistic Attribution with Author-Document Topic Model	**CE**	Cross Entropy
ADF	Augmented Doppelgänger Finder			**LOWBOW**	Locally-weighted bag of words

3. METHODOLOGY

We propose two separate approaches for instance-based and profile-based author attribution.

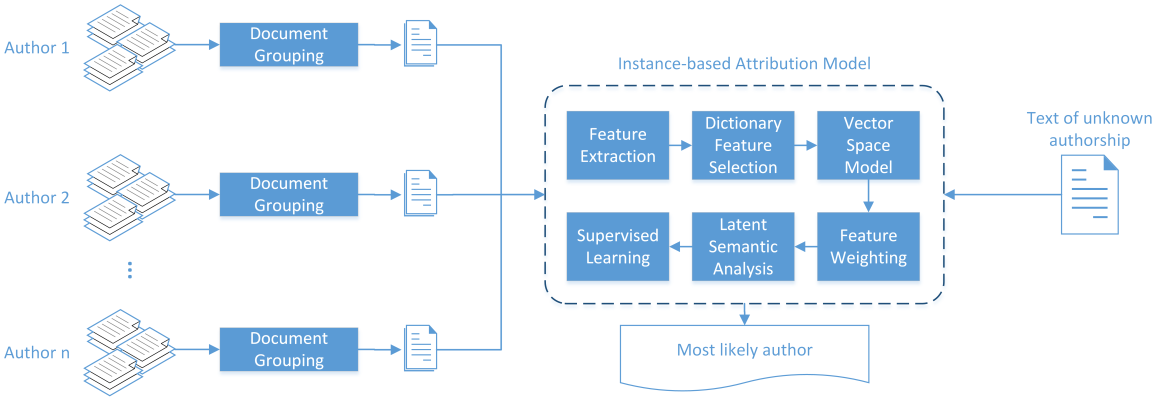


Figure 3.1. Pipeline of the proposed instance-based approach.

The pseudo-code of our instance-based model are given in Algorithm A.2, and it has following steps, as illustrated on Figure 3.1:

(i) Documents of each known author are randomly grouped and concatenated (e.g. if an author has 1,000 documents and group size is set to 20, then the author will have 50 enriched documents after concatenation.)

(ii) N-gram features are extracted for all enriched author documents. In this work, character n-grams are preferred due to its superiority over word n-grams.

(iii) A subset of the dictionary is extracted after ranking by using one of the mentioned feature selection methods from Section 3.2.

(iv) All the enriched documents of the authors are represented with a vector space model where columns are documents, rows are terms of the dictionary subset, as explained in Section 3.3.

(v) Global and local feature weighting schemes are applied on the vector space model after L_2 normalisation of document vectors.

(vi) The weighted vector space model of author documents is transformed into a subspace by using latent semantic analysis (LSA).

(vii) Multi-class supervised learning model is trained with the transformed vector space model in which each author represent a class with his/her documents. (In this study, extreme learning machine excels other supervised learning methods.)

(viii) When a text or group of texts from an unknown author are given, the all steps in the attribution model are applied for also them. Thus, the preferred supervised learning method predicts the most likely author of the query texts.

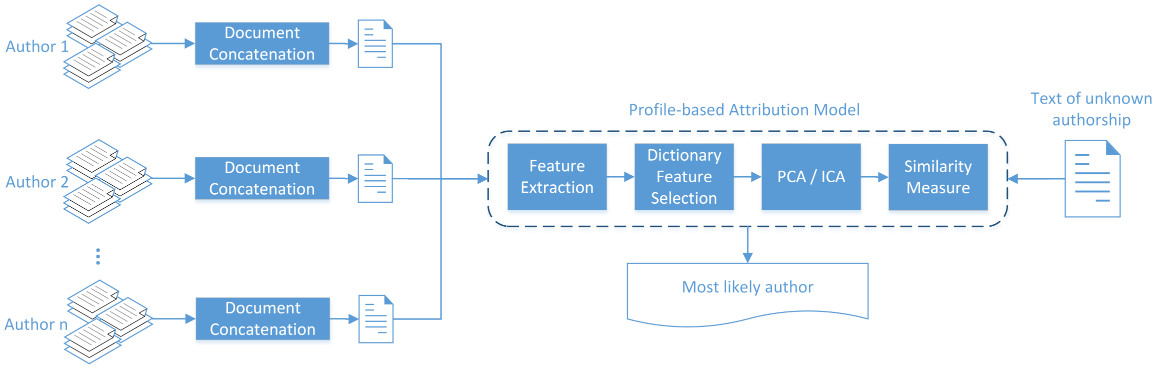


Figure 3.2. Pipeline of the proposed profile-based approach.

As illustrated on Figure 3.2, for profile-based author attribution, all the documents of each known author are concatenated to create author profiles. After that, feature extraction and dictionary feature selection steps are followed as similar to instance-based approach. The main difference here is that each author has only one enriched documents while more documents can exist for each author in instance-based approach. In order to suppress noise in textual data and to represent author profiles in a compact manner, all profile are transformed into a subspace with independent component analysis (ICA) or principal component analysis (PCA). Finally, the profile which give the minimum dissimilarity score with the document of unknown author is attributed as the most likely author. In this study, cosine dissimilarity is preferred to compare projected profiles of authors and unknown document query. Pseudo-codes of the profile based approach are also given in Algorithm A.1.

3.1. Feature Extraction

In this study, we used 4 different kinds of features: (i) n-gram characters, (ii) n-gram words, (iii) stylometric features, (iv) locally weighted bag of words (LOWBOW).

For sequences of words and characters, n-gram features are extracted as seen in Table 3.1. In case of character n-gram, n value is changed from 2 to 6. Nevertheless, for words, only 1-gram features are extracted due to computational limitations that will be explained in Section 3.2.

Table 3.1. N-gram character and word feature samples

Feature Type	**Utterance**	**Extracted Features**
3-gram character	“Anayurt Oteli”	“Ana”, “nay”, “ayu”, “yur”, “urt”, “rt_”, “t_O”, “_Ot”, “Ote”, “tel”, “eli”
3-gram word	“Tired with all these, for restful death I cry”	“# Tired with”, “Tired with all”, “with all these”, “all these for”, “these for restful”, “for restful death”, “restful death I”, “death I cry”, “I cry #”

Both word and character n-gram features consider only term frequency, and semantic term relations are mostly ignored. For higher order n values, limited semantic information (i.e. contextual marks, style nuances) can be caught.

Combining position of the words or characters with their frequency weightings is one of the ways to store more semantic relations [34]. For that reason, LOWBOW framework has been also compared with raw n-gram features (How LOWBOW is extracted will be explained in more detail in Section 4.5).

Moreover, due to widespread usage of stylometric features in the literature, we have also applied them as described in Section 4.4 in order to analyse accuracies for author recognition in a comparison with the other features.

3.2. Dictionary Feature Selection

Features extracted from character or lexical word frequencies have generally high dimensionality. Especially, representation of a text in n-grams with $n > 2$ requires thousands of features. On the other hand, n-grams with higher n degrees do not only provide lexical information, but also provide clues about syntactic behaviours of an author. For that reason, there should be a trade-off between dictionary size and expressivity.

However, in natural languages, word frequencies follow a known distribution [52], such that most of the tokens in textual data are actually composed of few words with very high frequency (e.g. “a”, “the”, “I”, etc.), and many words with low frequency (e.g. “accordion”, “catamaran”, “ravioli”). What is remarkable is that the distribution is mathematically simple, roughly obeying a power law known as Zipf’s Law: the r^{th} most frequent word has a frequency $f(r)$ that scales according to

$$f(r) \propto \frac{1}{r^{\alpha}} \tag{3.1}$$

for $\alpha \approx 1$. In this equation, r is called the “frequency rank” of a word, and $f(r)$ is its frequency in a natural corpus. n-grams for both character and word features also obey the same principle with different values of α. Since the actual observed frequency will depend on the size of the corpus examined, this law states frequencies proportionally: the most frequent feature ($r = 1$) has a frequency proportional to 1, the second most frequent feature ($r = 2$) has a frequency proportional to $\frac{1}{2^{\alpha}}$, the third most frequent feature has a frequency proportional to $\frac{1}{3^{\alpha}}$, and so on.

Our premise is that focusing on a part of the dictionary by leveraging the Zipf’s Law distribution will not only reduce the dimensionality of the document vectors, but if well-selected, can even improve accuracy. Furthermore, what part of the dictionary should be deemed relevant could be a domain-specific question. For that reason, whether high-frequency words or more discriminative words should be prioritised, is an important question.

Hence, determining a cut-off threshold for a dictionary size is not only inter-domain issue in case of data changes but also intra-domain issue regarding preferred feature types. That’s why, dictionary size should change in accordance with feature and data to be tested unless computational complexity is out of question for the system. Extraction of sub-dictionary (or determining cut-off frequency) can be handled with various approaches as described below:

(i) Global Frequent Ranking (GFR): All the features extracted from the dataset are ordered with their frequencies in a descending way, and the top k unique features are selected to represent the sub-dictionary.

(ii) Local Frequent Ranking (LFR): Features of each author are ranked in descending order separately. After that, the sub-dictionary of each author is determined by their own top k unique features. An example of such ranking is given in the SCAP method proposed by Frantzeskou *et al.*'07 [28]

(iii) Local Distinctive Ranking (LDR): Similar to local frequent ranking, each author is ranked with their own distinctive features in a descending order. Then, top k of the features are used to represent each author separately. Re-centering local profiles of each author according to global dictionary features as mentioned in the study of Layton *et al.* is an example of such ranking [36].

3.3. Vector Space Representation

The vector space model (VSM) has been a state-of-art approach in natural language processing applications [53]. In the VSM, a textual data can be symbolised as a vector of terms (bytes, characters, words, etc.). Based on this, assume there is n unique authors in the corpus $\boldsymbol{A} = [\boldsymbol{a_1}, \boldsymbol{a_2}, ...\boldsymbol{a_i}..., \boldsymbol{a_n}]$ where each author $\boldsymbol{a_i} \in \boldsymbol{A}$ has a varying number of documents, totalling N. Thus, the profile of an author can be represented as $\boldsymbol{a_i} = [\boldsymbol{d_{i,1}}, \boldsymbol{d_{i,2}}, ...\boldsymbol{d_{i,j}}..., \boldsymbol{d_{i,n_i}}]$ where each $\boldsymbol{d_{i,j}}$ is the document of author $\boldsymbol{a_i}$ and n_i is the number of documents belonging to the author $\boldsymbol{a_i}$. In the corpus, each document can be represented as a fixed-size vector of frequencies in the term space, i.e. $\boldsymbol{d} = [t_1, ..., t_M]$, $t_i \in \boldsymbol{V}$ where M is the term (feature) set size and $\boldsymbol{V}$ is term dictionary (or language profile). Then, VSM is an $M \times N$ matrix composed of vector representations of all the documents in the author corpus:

$$\boldsymbol{A_{MxN}} = \begin{bmatrix} d_1(t_1) & \cdots & d_N(t_1) \\ \vdots & \ddots & \vdots \\ d_1(t_M) & \cdots & d_N(t_M) \end{bmatrix} \tag{3.2}$$

This VSM representation is frequency based, and disregards the order of terms in the document.

In our study on the COPA Database, chat sessions are assumed as documents, but they are sometimes too short (e.g. 1-2 words or emoticons). For that reason, in order to pave the way for more representative profiles, chat entries of each author are randomly grouped, and documents in each grouped are concatenated. Here, each group is generally comprised of the same number of documents. Subsequently, for COPA experiments, each column of VSM is not a single instance of chat instance, but a number of chat instances, grouped together.

3.4. Feature Weighting Schemes

Different terms (words, phrases, character combinations, or any other indexing units to identify the contents of a text) may have different importance for chat biometrics. Term weighting approaches can highlight distinctive features by assigning appropriate weights to the terms. Weighting schemes are based on two fundamental principles according to how they are used on VSM:

(i) Local Weighting Scheme: If a term is used more frequently than others in a text or by an author, the term should have more importance than others.

(ii) Global Weighting Scheme: If a term is used commonly in different texts or by various authors, it is less distinctive than infrequently used terms. Hence, its weight or importance should be reduced.

Based on these basic principles, the weight of each term $t_i \in \boldsymbol{V}$ for the corpus $\boldsymbol{A}$ can be found in VSM:

$$\Phi_{i,j} = \Theta_i . \Gamma_{i,j} \tag{3.3}$$

where $\boldsymbol{\Theta_{(m \times 1)}}$ is a global weighting scheme for $t_i \in \boldsymbol{V}$ over all $\boldsymbol{d_j}$'s, and $\boldsymbol{\Gamma_{(m \times n)}}$ is a local weighting scheme for each t_i in $\boldsymbol{d_j}$.

Some common global and local weighting schemes are summarised in Table 3.2. For instance in the binary schemes, the presence of a term results in a value of 1, and its absence a value of 0, regardless of the frequency of the term.

In text categorisation [54] and authorship identification [3, 51], term frequency - inverse document frequency was found superior to other local-global weighting combinations. Nevertheless, we have compared some of them in our study in order to clarify how they perform on different datasets.

Table 3.2. Common weighting schemes used in the literature, where $f_{i,j}$ is the frequency of term t_i in document d_j and P is the probability of $f_{i,j}$.

Global Weighting Schemes		**Local Weighting Schemes**	
Scheme Formula	**Denotation**	**Scheme Formula**	**Denotation**
$\Theta_i \in \{1, 0\}$	binary	$\Gamma_{i,j} \in \{1, 0\}$	binary
$\Theta_i = 1/\sqrt{\sum_j f_{i,j}^2}$	normal	$\Gamma_{i,j} = f_{i,j}$	term frequency
$\Theta_i = n/\lvert d \in D : t \in d\rvert$	inverse document frequency	$\Gamma_{i,j} = \log(1 + f_{i,j})$	logarithmic term frequency
$\Theta_i = 1 + \sum_j P(f_{i,j}) \log P(f_{i,j}) / \log n$	entropy	$\Gamma_{i,j} = f_{i,j}/max_i(f_{i,j})$	augnorm
$\Theta_i = \sum_j f_{i,j}/\lvert d \in D : t \in d\rvert$	global frequency - IDF	$\Gamma_{i,j} \in \{1 + \log f_{i,j}, 0\}$	sub-linear term frequency

3.5. Subspace Projection and Dimensionality Reduction

LSA and PCA are two approaches depending upon eigenvalues which are applied for projecting high-dimensional datasets into subdimensions without losing important information. Therefore, these approaches have strong similarities, but this does not mean they are in exact relation. LSA is computed on the term-document matrix, while PCA is calculated on the covariance matrix, which means LSA tries to find the best linear subspace to describe the data set, while PCA tries to find the best parallel linear subspace. In other words, with LSA, the context is provided in the numbers through a term-document matrix, while in the PCA, the context is provided in the numbers through providing a term covariance matrix [55].

If we compare PCA with ICA: while the PCA aims to find an orthogonal linear transformation in order to maximise the variance of the variables, the goal of ICA is to produce statistically independent, non-Gaussian and spatially localised vectors [56]. Contrary to PCA, in ICA, the basis vectors are not orthogonal and ranked in order. Mathematically, PCA is adequate if the data distribution is Gaussian, linear, and stationary [57]. If these assumptions do not hold, or the raw data appear to be very noisy, ICA can produce a better approximation by means of minimising the mutual information of the output [58].

By depending on the difference mentioned above, PCA and ICA are preferred to transform author profiles, which do not have multiple documents due to concatenation in the profile-based method. On the other hand, LSA is used for creating correlation between terms and documents in instance-based model because the correlation can be ignored by VSM on which terms are orthogonal while representing documents.

3.5.1. Latent Semantic Analysis

LSA is an automated mathematical/statistical approach for extracting and deducing relationship among expected contextual usage of terms, words and phrases in passages of texts [59].

While different texts are being compared, texts with the same semantic content may not be obviously recognised as related with each other because expressions that give the same meaning with different terms can be preferred by the author. Furthermore, a term may have been used in different contexts to represent different meanings. For that reason, LSA, which is based on the principle that the semantic relationships between terms is present not explicitly, but only latently, aims to convert each document into a semantic structure for the purpose of probing into different layers of textual representation.

LSA is based on singular value decomposition (SVD) of VSM, such that:

$$\mathbf{\Phi} \approx \tilde{\mathbf{\Phi}} = \boldsymbol{U}\mathbf{\Sigma}\boldsymbol{V}^{\top}, \tag{3.4}$$

where $\boldsymbol{\Phi}$ is the weighted term-document matrix, $\tilde{\boldsymbol{\Phi}}$ is an approximation to $\boldsymbol{\Phi}$ composed by the truncated left and right singular matrices, $\boldsymbol{U}_{(MxR)}$ is the matrix of left singular vectors $\boldsymbol{u_i}$'s $(1 \leq i \leq M)$,$\boldsymbol{\Sigma}_{(RxR)}$ is the diagonal of singular values, and $\boldsymbol{V}_{(NxR)}$ is the matrix of right singular vectors $\boldsymbol{v_j}$'s $(1 \leq i \leq N)$. In other words, $\boldsymbol{u_i}$ and $\boldsymbol{v_j}$ are the projections of $t_i \in \boldsymbol{V}$, and $d_j \in \boldsymbol{A}$ respectively from the initial vector space model onto semantic representation domain [54].

The decomposition provides two different advantages: Firstly it eliminates sparsity by preserving significant elements of $\boldsymbol{\Phi}$, secondly it makes possible to truncate left and right singular vectors by depending on the size of R.

Let j^{th} weighted document in the term-document matrix $\boldsymbol{\Phi}$ be $\boldsymbol{d}_j^{\phi}$, by depending on the equation 3.4:

$$\boldsymbol{d}_j^{\phi} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{v}_j^{\top}, \tag{3.5}$$

$$\boldsymbol{v}_j = (\boldsymbol{d}_j^{\phi})^{\top}\boldsymbol{U}\boldsymbol{\Sigma}^{-1} \tag{3.6}$$

equations allow to expand the existing vector space with new documents. In such way, new query documents from an unknown author can be transformed into a pseudo-document of the semantic space:

$$query = \boldsymbol{q}^{\top}\boldsymbol{U}\boldsymbol{\Sigma}^{-1} \tag{3.7}$$

where $\boldsymbol{q}$ is the term-weighted query. After the query is transformed into the new space, similarity check of documents and/or terms can be possible.

3.5.2. Principal Component Analysis

PCA is an approach for handling with high-dimensional data, and using the dependencies between the variables to symbolise it in a more convenient, lower dimensional form, by preventing too much information loss. PCA is also known as one of the most straightforward and potent ways of reducing dimensionality. It is one of the oldest approach for such purpose, and has been rediscovered many times in many fields, so it is also known as the Hotelling transformation, the Karhunen-Loeve transformation, the method of empirical orthogonal functions, and SVD.

If the arrangement of points across many correlated variables are required, most significant directions in high dimensional data can be shown by using principal component analysis. Principal components are composed as follows:

- The first principal component is the linear combination of the standardised original variables that has the maximum variance.
- Each following principal component is the linear combination of the variables that has the maximum variance but does not have correlation with all previously specified components.

Let the vector space model for authorship analysis $\boldsymbol{A}$ be of $N \times M$ size, where N and M are the author number and variable number respectively in order to represent an author. Let us assume that $\boldsymbol{A}$ is centred, i.e. column means are zero. Then the covariance matrix $\boldsymbol{C}$ with a size of $N \times N$ is represented as:

$$\boldsymbol{C} = \frac{1}{N}\boldsymbol{A}^{\top}\boldsymbol{A} \tag{3.8}$$

The matrix is symmetric and can be diagonalised:

$$\boldsymbol{C} = \boldsymbol{V}^{\top}\boldsymbol{\Lambda}\boldsymbol{V} \tag{3.9}$$

where each column of $\boldsymbol{V}$ is an an eigenvector and $\boldsymbol{\Lambda}$ is a diagonal matrix of eigenvalues λ_i which are ordered on the diagonal in a descending way. The principal directions of the data is represented with these eigenvectors. Thus, principal components are actually projections of the data on the principal axes. In other words, the j^{th} column of $\boldsymbol{AV}$ gives j^{th} principal component, and the i^{th} row of $\boldsymbol{AV}$ gives the coordinates of the i^{th} data point in the transformed space.

If SVD of $\boldsymbol{A}$ is performed:

$$\boldsymbol{A} = \boldsymbol{U\Sigma V}^\top \tag{3.10}$$

where singular values σ_i composes $\boldsymbol{\Sigma}$ diagonal matrix. By depending on these, it can be seen that:

$$C = \frac{1}{N}\boldsymbol{V\Sigma U}^\top \boldsymbol{U\Sigma V}^\top = \boldsymbol{V}\frac{\boldsymbol{\Sigma}^2}{N}\boldsymbol{V}^\top \tag{3.11}$$

meaning that principal directions are right singular vectors V and that singular values are in correlation with the eigenvalues of covariance matrix:

$$\lambda_i = \frac{\sigma_i^2}{N}. \tag{3.12}$$

Then, principal components are given by:

$$\boldsymbol{AV} = \boldsymbol{U\Sigma V}^\top \boldsymbol{V} = \boldsymbol{U\Sigma} \tag{3.13}$$

While recognising the identity of an unknown test author $\boldsymbol{a}_{test}$, we calculate a projection $\tilde{\boldsymbol{a}}_{test}$, onto the principal components and compare the resulting vector of projection coefficients given by:

$$\tilde{\boldsymbol{a}}_{test} = \boldsymbol{a}_{test}\boldsymbol{U\Sigma} \tag{3.14}$$

Hence, the unknown author to be tested is identified as the author i if $\tilde{a}_{test}$ is closest to the projected feature vector $\tilde{a}_i$.

To reduce the author data dimension from N to $K < N$, K first columns of $\boldsymbol{U}$, and KxK upper-left part of $\boldsymbol{\Sigma}$ can be selected. Their product $\boldsymbol{U}_K\boldsymbol{\Sigma}_K$ is the required $N \times K$ matrix consisting of first K principal components.

3.5.3. Independent Component Analysis

ICA is a method for obtaining statistically independent variables from a mixture of them. Finding hidden factors in data to be investigated, or decomposition of data to the source signals are some of the widespread applications. ICA is based on the assumption that each observed signal $\boldsymbol{a_i}$ is a mixture of a set of N unknown independent source data $\boldsymbol{s_i}$ which are linearly integrated by means of an unknown mixing matrix $\boldsymbol{X}$. In ICA $\boldsymbol{a_i}$ and $\boldsymbol{s_i}$ are combined to build the rows of $\boldsymbol{A}$ and $\boldsymbol{S}$ matrices respectively. Thus, the model is:

$$\boldsymbol{A} = \boldsymbol{X}\boldsymbol{S} \tag{3.15}$$

For the ICA analysis, The data vectors having dimension of $|V| = M$ dictionary size are the lexicographically ordered terms t_i's. In short, the aim of ICA is finding a linear transformation $\boldsymbol{W}$ for the inputs that minimises the statistical dependence among the output components $\boldsymbol{a}_i$, the and being estimates of the hypothesised independent sources $\boldsymbol{s}_i$:

$$\boldsymbol{S} \cong \boldsymbol{W}\boldsymbol{A} \tag{3.16}$$

More explicitly, the author data matrix $\boldsymbol{A}$ will be $N \times M$ dimensional VSM. Decomposition of the matrix into N independent source components gives us $\boldsymbol{s}_i$, which take place through the rows of the output matrix $\boldsymbol{S} = \boldsymbol{W}\boldsymbol{A}$. Each row of the mixing matrix $\boldsymbol{X}_{(N\times N)}$, consists of weighting coefficients specific to a given author. Afterwards, for the author $\boldsymbol{a}_i$, the i^{th} row of $\boldsymbol{X}$ will build an N-dimensional feature vector.

In the recognition phase, if we assume the test set follows the same model for synthesis, we project a test author $\tilde{\boldsymbol{a}}_{test}$, onto the set of previously determined basis functions and compare the resulting vector of projection coefficients given by:

$$\tilde{\boldsymbol{a}}_{test} = \boldsymbol{a}_{test}\boldsymbol{S}^T(\boldsymbol{S}\boldsymbol{S}^T)^1 \tag{3.17}$$

Finally, the unknown author to be tested is simply recognised as the author i if $\tilde{\boldsymbol{a}}_{test}$ is closest to the feature vector $\tilde{\boldsymbol{a}}_i$.

3.6. Supervised Learning Methods

Extreme learning machine (ELM) proposed by Huang *et al.* [60] are feed-forward neural network used for regression or classification with a single-hidden-layer of nodes, and very fast to train in comparison to other conventional well-known learning methods. The computation speed is high, because the weights of the first hidden layer are randomly assigned and not iteratively tuned [61]. The second layer is then analytically solved. In the literature, Zheng *et al.* used ELM for text information retrieval purposes, but it has not been applied to author attribution before [62].

In our proposed IBA pipeline, we use multi-quadratic kernel ELM, and during optimum parameter search, we optimize the number of hidden layers (λ) in the range of $[100 - 800]$, mixing coefficient (α) and width coefficient (ω) between $[0 - 1]$ on the validation set.

One of the widespread learning algorithms for authorship attribution tasks is the support vector machine (SVM). That is why we have also used it to compare our approach with other benchmarks in the literature. In spite of the powerfulness of SVM for supervised classification purposes, its iterative learning mechanism can make training process slower with the increasing number of features and sample instances. In order to get higher performance by using SVM, optimising hyper-parameters is very crucial. For that reason, grid-search for optimisation can take more computational time.

Moreover, for training and prediction of authorship, we have also applied to the following approaches by using scikit-learn [63]: i) Multilayer Perceptron which is a feedforward neural network model, ii) Random Forest which is an ensemble learning method for classification, and iii) Multinomial Naive Bayes classifier, in which the naive Bayes algorithm is applied for multinomially distributed data.

4. BENCHMARK METHODOLOGIES IMPLEMENTED

As we summarised in Table 2.3, variety of techniques for author profiling exists in the literature. We have selected some of the common approaches among them, and we have implemented these benchmark methodologies in order to compare their author attribution performances with the proposed approaches mentioned in Section 3. These benchmarks are common n-grams [21], source code author profiling [28], recentered local profile [51], stylometry based attribution methods [23,29,41], and local histograms on character n-grams [34].

4.1. Common N-grams

In this methodology, an author profile is described with set of L most frequent n-grams generated from the all documents of the author. In other words, an author profile is a set of L pairs $(n_1, f_1), (n_2, f_2), (n_3, f_3), ...(n_L, f_L)$ where n_is are n-gram features and f_is are their normalised frequencies. In order to measure dissimilarity between two different author profiles, following formula is used:

$$Dissimilarity(P(x), P(y)) = \sum_{n \in P(x) \cup P(y)} \left(\frac{f_x(n) - f_y(n)}{\frac{f_x(n)+f_y(n)}{2}} \right) \tag{4.1}$$

where $f_x(n)$ and $f_y(n)$ are the n-gram frequencies for $P(x)$ and $P(y)$ profiles respectively [21].

4.2. Source Code Author Profile

Source code author profile (SCAP), which is a simplified version of common n-grams, counts only common character n-grams in profiles compared, such that:

$$Dissimilarity(P(x), P(y)) = 1 - \frac{|P(x) \cap P(y)|}{L} \tag{4.2}$$

4.3. Recentered Local Profile

The common n-grams (CNG) method uses the relative distance between two documents (or author profiles), and serves as a basis for RLP. However, the most noticeable difference is that RLP measures the profile similarity according to most distinctive features, rather than the most frequently used features, using the standardised language profile approach described below:

$$Dissimilarity(P(x), P(y)) = \sum_{n \in P(x) \cup P(y)} \frac{(f_x(n) - E(n)).(f_y(n) - E(n))}{||f_x(n) - E(n)||.||f_y(n) - E(n)||} \tag{4.3}$$

where $f_x(n)$ and $f_y(n)$ are n-gram frequencies for $P(x)$ and $P(y)$ profiles respectively, and E is the language profile which is extracted from the entire training set as an approximation to the absolute language profile. Because of the flexibility inherent in natural languages, extracting the absolute profile of a language is impossible. For this reason, all the normalised author profiles in the training set are combined to extract a standardised language profile.

4.4. Stylometry-based Attribution

Although various authorship attribution methods were applied with changing set of stylometry features, the fundamental principle of such systems is to train a supervised model via these features for predicting authorship, or to measure dissimilarity among author profiles as summarized in Table 2.3.

In this study, a set of 95 linguistic features consisting of 4 character, 79 lexical word and 12 syntactic features have been used for stylometry-based author attribution as given in Table 4.1. In our pipeline, we have normalised them by removing the mean and scaling to unit variance, and false discovery rate was used to select more discriminative features.

Table 4.1. Stylometry features

Character-based features	20. # of short unique words (1-to-3 letters) starting with vowels/ of words
1. # of letters / # of characters	21. # of middle-sized unique words (4-to-7 letters) starting with vowels/ # of words
2. # of capital letters / # of characters	22. # of long unique words (more than 7 letters) starting with vowels/ # of words
3. # of capital letters / # of letters	23. # of hapax dislegomena / # of words
4. # of digits / # characters	24. Guirad's R [29]
Word-based features	25. Duggast's U [29]
5. average word length	26. Brunet's W [29]
6. # of unique words / # of words	27. Herdan's C [29]
7. # of X-letter words / # of words (where X in [1-15])	**Syntax-based features**
8. # of X-letter words starting with vowel / # of words (where X in [1-15])	28. # of punctuations / # of words
9. # of X-letter unique words / # of words (where X in [1-15])	29. # of punctuations / # of characters
10. # of X-letter unique words starting with vowel / # of words (where X in [1-15])	30. # of marks (#, ?, &, %, !, $,) / # of words
11. # of short words (1-to-3 letters) / # of words	31. # of marks (#, ?, &, %, !, $,) / # of characters
12. # of middle-sized words (4-to-7 letters) / # of words	32. # of marks (, ; . : \' / *) / # of words
13. # of long words (more than 7 letters) / # of words	33. # of marks (, ; . : \' /,*) / # of characters
14. # of short words (1-to-3 letters) starting with vowels/ # of words	34. # of three points (...) / # of words
15. # of middle-sized words (4-to-7 letters) starting with vowels/ # of words	35. # of three points (...) / # of characters
16. # of long words (more than 7 letters) starting with vowels/ # of words	36. # of positive emoticons (:), :-D, etc.) / # of words
17. # of short unique words (1-to-3 letters) / # of words	37. # of positive emoticons (:), :-D, etc.) / # of characters
18. # of middle-sized unique words (4-to-7 letters) / # of words	38. # of negative emoticons (:(, :-S, etc.) / # of words
19. # of long unique words (more than 7 letters) / # of words	39. # of negative emoticons (:(, :-S, etc.) / # of characters

4.5. Local Histograms on Character N-grams

Local histograms are combination of term position weighting and term frequency weighting over the terms in vocabulary, such that:

(i) Let $\boldsymbol{d}_i = [x_{i,1}, ..., x_{i,|\boldsymbol{V}|}]$ represent the document i of an author, where $\boldsymbol{V}$ is the vocabulary and $|\boldsymbol{V}|$ is the number of elements in $\boldsymbol{V}$;

(ii) Let $W_i = \{w_{i,1}, ..., w_{i,N_i}\}$ represent the terms used by an author in i^{th} document in order of appearance, where N_i is the number of terms that exist in document

i, and $w_{i,j} \in \boldsymbol{V}$ is the term positioned at j;

(iii) Let $\boldsymbol{v}_i = \{v_{i,1}, ..., v_{i,N_i}\}$ be the set of indexes in the vocabulary $\boldsymbol{V}$ of the terms shown in W_i;

(iv) Let $\boldsymbol{s} = \{s_1, ..., s_{N_i}\}$ be set of scalars determining intervals such that each s_j can be linked to a position in W_i;

Given a kernel smoothing function:

$$K_{\mu,\sigma}(x) = \begin{cases} \frac{N(x;\mu,\sigma)}{\psi\left(\frac{1-\mu}{\sigma}\right)-\psi\left(\frac{-\mu}{\sigma}\right)} & \text{if } x \in [0,1] \\ 0 & \text{otherwise} \end{cases} \tag{4.4}$$

where $\mu \in [0, 1]$ is the location parameter, σ is the scale parameter and $\psi(x)$ is the cumulative distribution function. Thus, a local histogram for each position $\mu_j \in \{\mu_1, ...\mu_k\}$ is computed as follows:

$$\boldsymbol{h^j}_{i,\{v_{i,1},\ldots,v_{i,N_i}\}} = \boldsymbol{d}_{i,\{v_{i,1},\ldots,v_{i,N_i}\}} \times K_{\mu_j,\sigma}(\boldsymbol{s})(4.5)$$

Thus, a set $\boldsymbol{h}_i^{\{1,..,k\}}$ of k local histograms are extracted for i^{th} document. By means of each histogram, information about the distribution of the terms at certain positions can be stored; in other words, sequential information of the author's document is gathered.

Using these local histograms as a feature representation of each document by an author, building a multi-class SVM classifier gives us an instance-based authorship attribution method [34].

5. EXPERIMENTAL DATA

We have used two datasets in Turkish language for our evaluations, which we detail in this chapter. Additionally, we test the generalisation of the proposed approaches on corpora of English and Portuguese news articles.

5.1. Turkish Corpora

5.1.1. COPA

The proprietary CCSoft Okey Player Abuse (COPA) Database, consisting of demographics, statistics, game records, interactions and complaints of thousands of players [4] is used in our study. The database is acquired from a commercial Okey game over a six months period, and incorporates roughly 100,000 unique players, who played the game at least once. All the player identification information is deleted to protect player privacy. In the mentioned period, a total of 800,000 Okey games were recorded along with the player interactions in the chat area and the dataset contains Turkish chat inputs from more than 30,000 user accounts.

The database is particular in that messages are always written in a multiparticipant fashion (there are always four players in a game); they are unedited (except for a black-list that contains the most frequently attempted insults); and they are spontaneously produced. The number of chat and game records per player vary greatly. Consequently, we have pre-selected a subset of the dataset for the problem of chat biometrics before any research or modelling took place. We sorted chat participants according to the number of unique words used by each, and eliminated participants who had vocabulary sizes less then 100 unique words. This is a very coarse pre-processing, but people with very limited vocabulary might be easier to identify, and might positively bias the results. The remaining users are sorted in decreasing order according to number of active chat sessions, and the most active users are selected for building a chat biometrics benchmark database. With 403 users, this database is one order of magnitude bigger than the most relevant work from the literature.

Table 5.1 describes the properties of the database. Since we planned to use 5-fold cross validation, as well as to assess the effect of the number of chat entries per user, we required that at least 5 chat sessions per user should exist for each of five folds.

Table 5.1. Statistics for the chat biometrics subset of the COPA database.

Corpus Characteristics	**Value**
# of users	403
# of chat sessions per user	261
# of chat returns per user	3,251
# of unique words per user	2,375
# of words per user	10,933
# of letters per user	39,494
# of capital letters per user	149
# of emoticons per user	288
# of digits per user	162
# of punctuations per user	679

5.1.2. Ekşisözlük

Ekşisözlük is an online dictionary in Turkish which is built on collaborative user contribution. Nevertheless, it is not a literally dictionary; users are not obliged to write right information. It is actually one of the biggest online communities in Turkey with more than 400,000 registered users. It has about 54,000 writers. As an online public platform, Ekşisözlük is used for sharing information on variety of subjects ranging from science to daily life issues, as well as applied as a virtual socio-political community to communicate controversial political contents and to expound personal opinions [64].

In our study, we have randomly selected 252 authors to create a test corpus from Ekşisözlük, each of whom has more than 1,000 entries in different topics. Some of the entries contains only a few number of characters such as reference or redirection contents, as well as very long entries also exist with hundreds of words. In order to distinguish short and long entries,a threshold character length for an entry is determined as 140. Afterwards, some entries are randomly removed from the corpus until each author has at least 250 long entries. For instance, an author with initial 1,000 entries has become totally 467 short and 250 long entries after the elimination. Table 5.2 shows short summary of Ekşisözlük database properties.

Table 5.2. Statistics for the chat biometrics subset of the Ekşisözlük database.

Corpus Characteristics	**Value**
# of users	252
# of entry per user	491
# of unique words per user	9,273
# of words per user	23,223
# of letters per user	142,927
# of capital letters per user	0
# of emoticons per user	8
# of digits per user	1,271
# of punctuations per user	5,890

5.1.3. Comparison of Turkish Datasets

If we take a glance on most frequent words, in the COPA database, many conversations are specific to games played by the participants and mainly contain greeting and gratitude expressions as seen in Table 5.3. On the other hand, in the Ekşisözlük database, pronouns, adjectives, adverbs and conjunction words are mostly preferred by users and they do not imply any specific purpose, contrary to the COPA database. The frequency-rank distribution of words vary between the corpora, but they roughly obey the Zipf's Law for the first 10 words.

When we try to find an optimum dictionary subspace by means of cut-off threshold in order to comprise significant number features extracted in a dataset, we first need to look at dictionary coverage, i.e. the ratio of total number of terms represented by the dictionary subspace to the actual total number of terms in the textual dataset.

Conversations in the COPA database (in which game participants have many idiosyncratic behaviours including spelling mistakes and shortenings) are dominated by

Table 5.3. Top-10 words with normalised frequencies in COPA and Ekşisözlük.

COPA Database			Ekşisözlük Database		
Word Used	**Meaning in English**	**Frequency**	**Word Used**	**Meaning in English**	**Frequency**
slm	abbr. of "hello"	0.01299	bir	one/some	0.02172
ben	I	0.01159	de/da	so/also/too/either	0.01627
ne	what	0.01021	ve	and	0.01149
sen	you	0.00857	bu	this	0.01071
yok	not	0.00835	bkz	redirection abbr.	0.00476
bu	this	0.00799	o	he/she/it/that	0.00469
ya	or	0.00729	çok	much/many/very	0.00450
tşk	abbr. of "thanks"	0.00649	için	for/so	0.00434
evet	yes	0.00643	ne	what	0.00412
tbr	abbr. of "congrats"	0.00639	ama	but	0.00403

a limited number of unique words with very high frequency. For that reason, coverage rate of COPA is higher than that of Ekşisözlük until number of unique terms (k) reaches 3,000, and after that it becomes flatter, as shown in Figure 5.1. Conversely, Ekşisözlük database, which contains relatively more daily life topics and less grammatical errors, draws a higher slope after (k) reaches 3,000, because vocabulary usage is richer and has a more balanced distribution. These inter-domain differences among the datasets illustrate the potential benefit of tailoring parameters for a given domain.

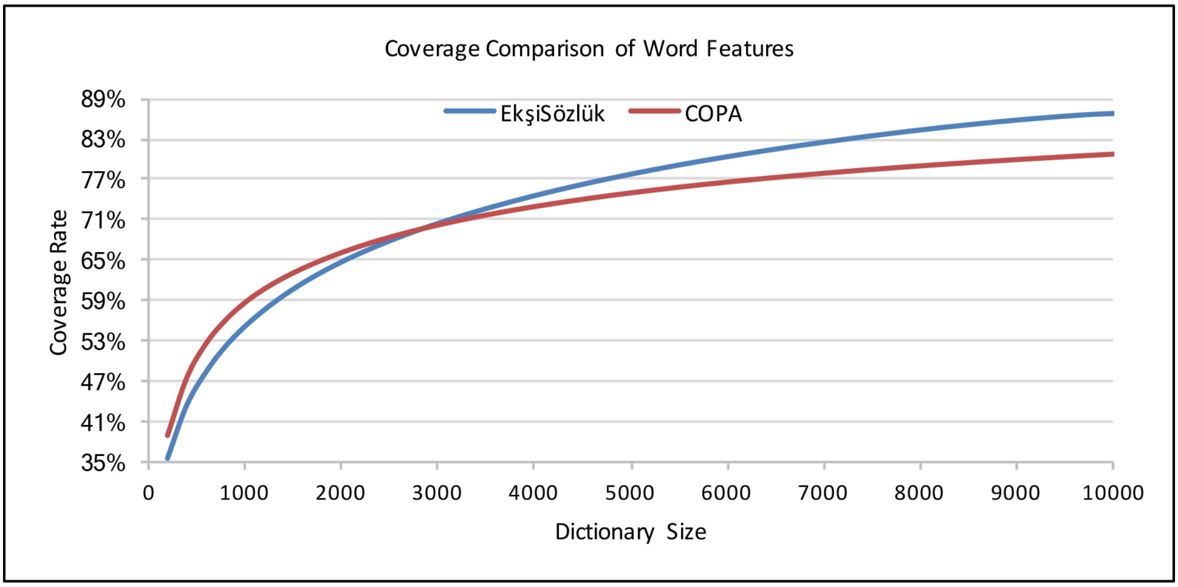


Figure 5.1. Inter-domain variation: Comparison of dictionary coverage on COPA and Ekşisözlük datasets.

If we look at the coverage rate of n-grams for Ekşisözlük dataset in order to visualise intra-domain variances, we see that coverage rate for an n-gram feature representation reaches at saturation very fast in case of decreasing n values as shown in Figure 5.2. In other words, with the increasing n value, coverage rate tends to change slowly, which requires higher cut-off thresholds to extract an optimum dictionary subspace.

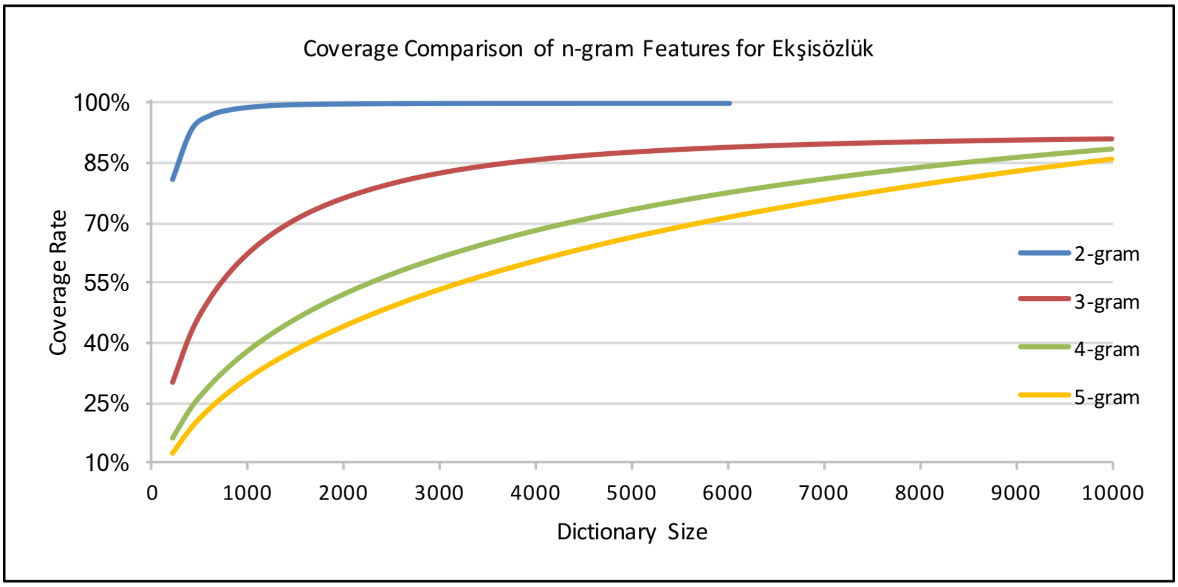


Figure 5.2. Coverage rate comparisons of features for intra-domain changes on Ekşisözlük.

5.2. Non-Turkish Corpora

5.2.1. C10 Database

C10 database contains a subset of English News from the Reuters Corpus Volume I (RCV1) which has over 800,000 manually categorised news-wire. The subset for author attribution is composed of 10 candidate authors, each of whom has 100 texts labelled in Corporate/Industrial (CCAT) group of RCV1 [65]. It was used for comparative evaluation of the most 15 influential author identification methods by reproducing the proposed approaches in the literature [66]. For that reason, C10 dataset is a benchmark point for author attribution efforts.

5.2.2. Portuguese News

The Language of the dataset is Brazilian Portuguese. It is composed of 3,000 documents, from 100 different authors derived from online newspapers and blogs. These are seperated in 10 categories according to the subject: Gastronomy, Literature, Politics, Health, Technology,Law, Economy, Sports, and Tourism. The purpose of choosing these subjects is to have a data which are mostly referred in daily newspapers. If a subject of a derived document would not fit in these categories, the document is left with an “Unspecified Subject”. This dataset has being used elsewhere in authorship attribution works [67] [35]. The documents have an average size of 2989 bytes, with a 1531 standard deviation. Each document has, in average, 486 tokens and 286 hapax legomena (words occurring just once).

6. EXPERIMENTS AND RESULTS

6.1. Experimental Protocol

We have used 5-fold cross validation in all the experiments, where the text produced by each author is divided into non-overlapping folds. The feature extraction is performed separately for each fold in order to guarantee that the test data are held out of the entire process.

In C10 and Portuguese News datasets, essays are divided into folds for each author without concatenating them because they are used as is in other benchmark studies. In other words, each separate document is an instance for an author in these datasets. On the other hand, in COPA, the accumulated chat records of a user in one session are used as an instance; for Ekşisözlük, each individual entry of a user is assumed as an instance. Thus, 50 instances of a user are concatenated in order to create a more representative profile for that author in training stage. While testing, the number of instances per query is increased from 1 to 50 for COPA and Ekşisözlük in order to analyse how performance changes with respect to the number of samples per author.

A small part of the raw data on COPA was normalised by using the web API of a Turkish NLP tool [68], whereby intentional or accidental misspellings were replaced with correct forms. Since Turkish has flexible sentence structure, as well as agglutinative word forms, the normalisation affects the identification performance significantly. For instance the raw sentence “bütttttttüüüünnnn insnlar e$it dogaaarr” (“all people are born equal”) is normalised as “bütün insanlar eşit doğar”. Normalisation changes the distribution of the features. Hence, COPA-NORM dataset is created to understand the effect of normalisation on authorship analysis by using a 83 author subset (due to query limitation of the NLP tool).

6.1.1. Performance Measures

For recognition, recognition rate (accuracy), F_1-measure, logarithmic loss, and cumulative match score (CMC) curve have been provided. In order to explain these measures in more detail, we should give some definitions in advance:

True positive (TP): Text queries that were correctly attributed to the author,
True negative (TN): Text queries that were correctly not attributed to the author,
False positive (FP): Text queries from the other authors that were incorrectly attributed to the author,
False negative (FN): Text queries of the author that were incorrectly not attributed to the author.

6.1.1.1. Recognition Rate (Accuracy). Recognition Rate is calculated by dividing the number of true classified tests to the number of all tests. In other words,

$$RecognitionRate = \frac{TP + TN}{TP + TN + FP + FN} \tag{6.1}$$

6.1.1.2. F_1-measure. Harmonic mean of precision and recall gives us F_1-measure:

$$F_1 - measure = 2.\frac{precision.recall}{precision + recall} \tag{6.2}$$

where

$$Precision = \frac{TP}{TP + FP} \tag{6.3}$$

$$Recall = \frac{TP}{TP + FN} \tag{6.4}$$

6.1.1.3. Logarithmic Loss. Accuracy which is the count of predictions where the predicted value equals to the actual value is not always a good indicator because of its yes or no nature. On the other hand, Logarithmic loss takes into account the uncertainty of the prediction based on how much it varies from the actual label.

The logarithmic loss metric is the negative log likelihood of the prediction model in which each observation of test is independently selected from a distribution that puts the submitted probability mass on the related class for each observation, such that:

$$logloss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{i,j} log(p_{i,j}) \tag{6.5}$$

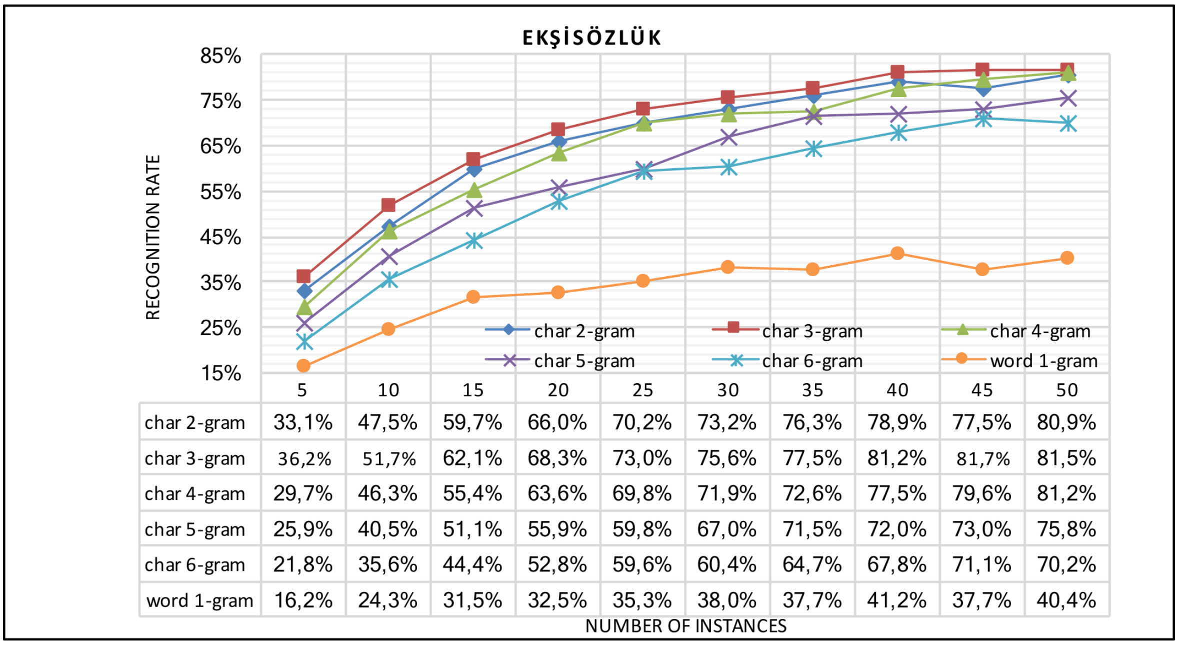
where N and M represent observation number, and number of class labels respectively, log represents the natural algorithm, $y_{i,j}$ equals to 1 if observation i in class j, and equals to 0 otherwise, and $p_{i,j}$ represents the predicted probability of observation i in class j.

6.1.1.4. Cumulative Match Characteristics. CMC is used to measure accuracy of a biometric system in the closed set recognition task. In this measurement, a ranking is applied on the queries in the test set and the probability that the correct attribution has a rank equal to or less than some value is plotted over the size of the test set. Namely, CMC illustrates how frequently a query appears in the ranks $(1, 2, 3, ..., 100,$ etc.) compared to recognition rate [69].

6.2. Fine-tuning on the Pipeline

6.2.1. TEST-1: Effect of Feature Type

6.2.1.1. Comparison of character - word features. The application scenario we mainly focus on is closed-set recognition. Firstly, we have compared author recognition rates of the two Turkish datasets by using character n-gram and word frequency features on the proposed instance-based pipeline. For both databases, 3-gram and 4-gram character features give better recognition rates than their alternatives, as shown in Figure 6.1 and Figure 6.2. Higher order n-grams for words are not feasible, as they are computationally very expensive.

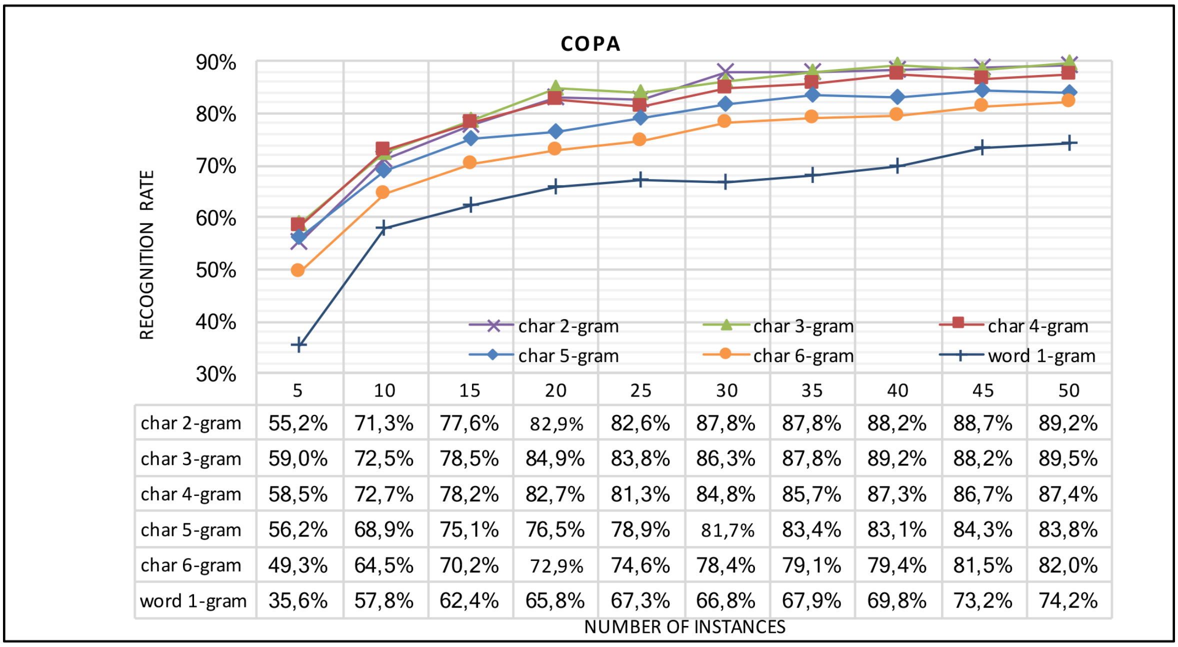


	5	10	15	20	25	30	35	40	45	50
char 2-gram	33,1%	47,5%	59,7%	66,0%	70,2%	73,2%	76,3%	78,9%	77,5%	80,9%
char 3-gram	36,2%	51,7%	62,1%	68,3%	73,0%	75,6%	77,5%	81,2%	81,7%	81,5%
char 4-gram	29,7%	46,3%	55,4%	63,6%	69,8%	71,9%	72,6%	77,5%	79,6%	81,2%
char 5-gram	25,9%	40,5%	51,1%	55,9%	59,8%	67,0%	71,5%	72,0%	73,0%	75,8%
char 6-gram	21,8%	35,6%	44,4%	52,8%	59,6%	60,4%	64,7%	67,8%	71,1%	70,2%
word 1-gram	16,2%	24,3%	31,5%	32,5%	35,3%	38,0%	37,7%	41,2%	37,7%	40,4%

Figure 6.1. Comparison of lexical features for Ekşisözlük. (Dictionary size: 5,000 with local frequent ranking, no weighting, ELM params: $\lambda = 250$, $\alpha = 0,5$, $\omega = 0,5$.)

We note that the recognition rates on Ekşisözlük dataset with word frequency features is much lower than with character n-grams, while word frequency of COPA gives similar patterns with character n-grams. The reason is the amount of grammatical mistakes prevalent in social chat (COPA), which makes misspellings into distinctive indicators for their authors.

6.2.1.2. Comparison of character n-grams and stylistic features. In this experiment, the style-based features mentioned in Table 4.1 and character n-gram features are extracted separately for Ekşisözlük, C10 and Portuguese datasets. For n-grams, dictionary size is limited to 2,500 after GFR ranking and $n = 5$. Each 50 documents are concatenated to get enriched instances for Ekşisözlük, and each single document is used as an instance for C10 and Portuguese News. By training SVM, authorship recognition performances are compared for both separate feature set.



	5	10	15	20	25	30	35	40	45	50
char 2-gram	55,2%	71,3%	77,6%	82,9%	82,6%	87,8%	87,8%	88,2%	88,7%	89,2%
char 3-gram	59,0%	72,5%	78,5%	84,9%	83,8%	86,3%	87,8%	89,2%	88,2%	89,5%
char 4-gram	58,5%	72,7%	78,2%	82,7%	81,3%	84,8%	85,7%	87,3%	86,7%	87,4%
char 5-gram	56,2%	68,9%	75,1%	76,5%	78,9%	81,7%	83,4%	83,1%	84,3%	83,8%
char 6-gram	49,3%	64,5%	70,2%	72,9%	74,6%	78,4%	79,1%	79,4%	81,5%	82,0%
word 1-gram	35,6%	57,8%	62,4%	65,8%	67,3%	66,8%	67,9%	69,8%	73,2%	74,2%

Figure 6.2. Comparison of lexical features for COPA. (Dictionary size: 5,000 with local frequent ranking, no weighting, ELM params: $\lambda = 250$, $\alpha = 0,5$, $\omega = 0,5$.)

Table 6.1 summarises the superiority of n-gram features over stylometric features among 3 different authorship corpora with 3 different languages. Moreover, following Figures 6.3, 6.4 and 6.5 also illustrate the same result.

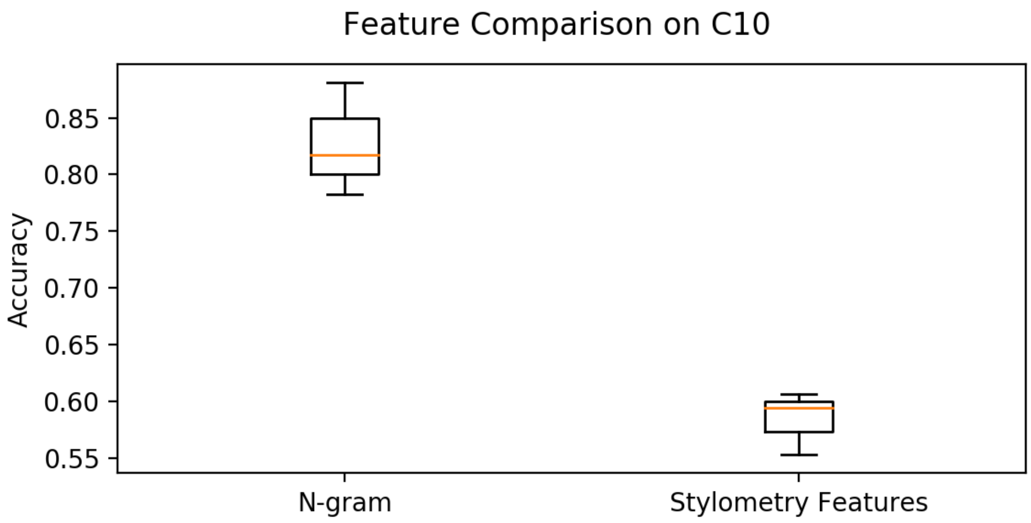


Figure 6.3. Box plot of authorship attribution accuracy for n-gram and stylistic features on C10 dataset (linear SVM is used to predict authorship).

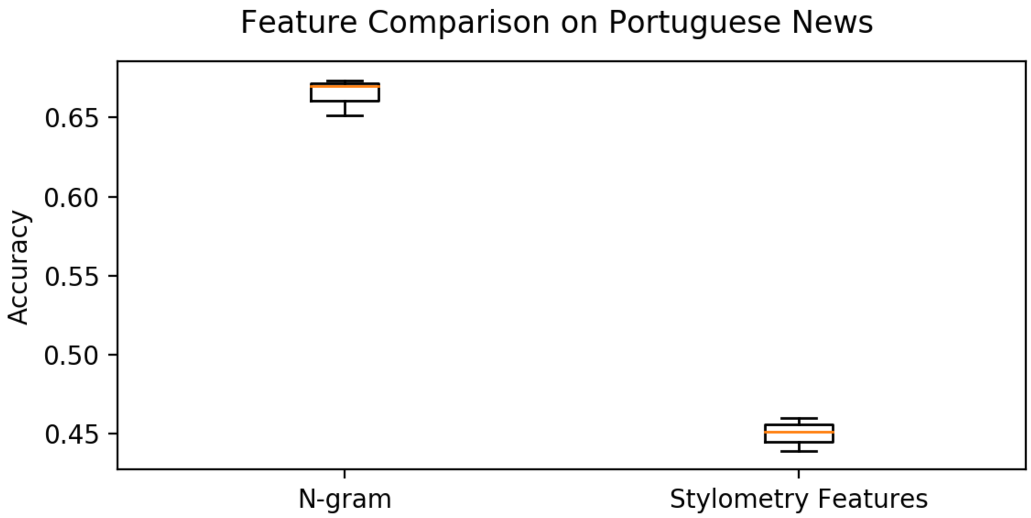


Figure 6.4. Box plot of authorship attribution accuracy for n-gram and stylistic features on Portuguese News dataset (linear SVM is used to predict authorship).

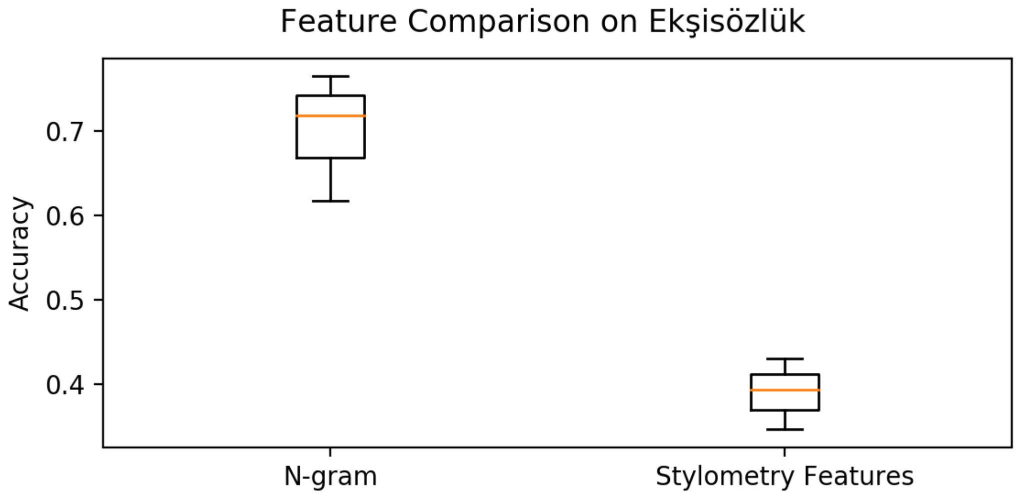


Figure 6.5. Box plot of authorship attribution accuracy for n-gram and stylistic features on Ekşisözlük dataset (linear SVM is used to predict authorship).

Table 6.1. IBA cross validation accuracies of authorship recognition for different feature types.

	C10	**Portuguese News**	**Ekşisözlük**
Character *n*-gram features	82.7% ±2.0	66.5% ±0.5	69.2% ±3.8
Style-based features	58.2% ±1.1	44.9% ±0.5	39.0% ±1.7

6.2.1.3. Comparison of character histograms and n-grams. In this test, we have compared how LOWBOW histograms and raw n-grams have different effects on an instance-based author recognition pipeline, such that:

(i) 5-gram character features are extracted for each documents of authors by limiting the dictionary size to $2,500$ after GFR ranking. Thus the frequency of 5-gram terms are fed into linear SVM in order to create a prediction model for authorship attribution.

(ii) LOWBOW histograms are extracted for 5-gram character sequences. Here, 12 histograms are computed for each documents of authors by limiting the dictionary size to $2,500$ after global most frequent ranking. The combined histograms are used to train linear SVM, and unknown document queries are attributed to an author with the SVM model.

Extracting LOWBOW has high computational complexity, and it is not feasible on COPA and Ekşisözlük corpora, both of which have huge document volumes. For that reason, C10 and Portuguese News datasets are preferred for the test.

As shown in Figure 6.6, inner quartile ranges for n-gram and LOWBOW features have similar distributions over accuracy. Indeed, the accuracies are $81.97\% \pm 2.85$ for n-gram and $82.22\% \pm 1.30$ for LOWBOW, which means that the difference between their accuracies is not significant in terms of paired t-tests, $p = 0.0075$.

In order to seek for which one is better on C10 dataset, logarithmic loss is measured for the features as shown in Figure 6.7, because it can give hints about the uncertainty of the recognition as described in Section 6.1.1.3. In this case, loss values become -0.668 ± 0.043 for n-gram and -0.724 ± 0.041 for LOWBOW which significantly shows the superiority of n-grams over LOWBOW features.

On the other hand, when testing on Portuguese News dataset, n-gram features give better authorship accuracy and smaller logarithmic loss value than LOWBOW features give as illustrated in Figures 6.8 and 6.9 respectively. The accuracy results are 66.50%±0.49 and 61.51%±0.56, as well as logarithmic loss values are -2.026 ± 0.025 and -2.334 ± 0.031 for n-gram and LOWBOW features in order.

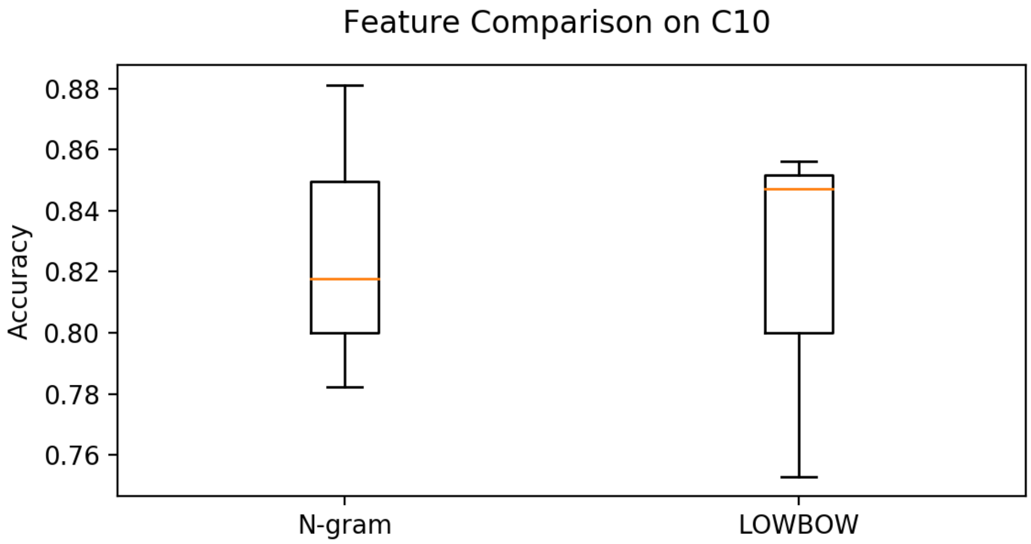


Figure 6.6. Box plot of authorship attribution accuracy for n-gram and LOWBOW features on C10 dataset (orange line represents median value).

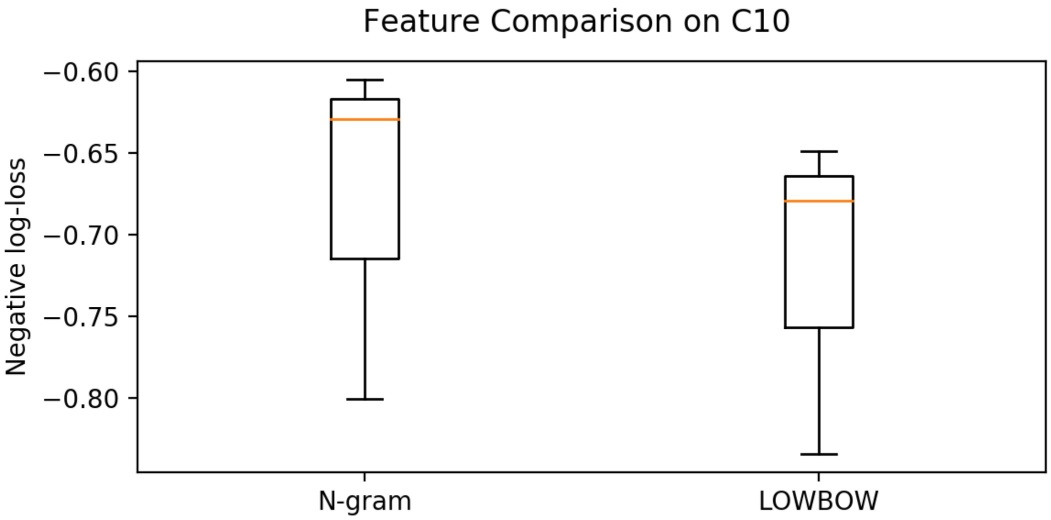


Figure 6.7. Box plot of logarithmic loss values while predicting authorship for n-gram and LOWBOW features on C10 dataset.

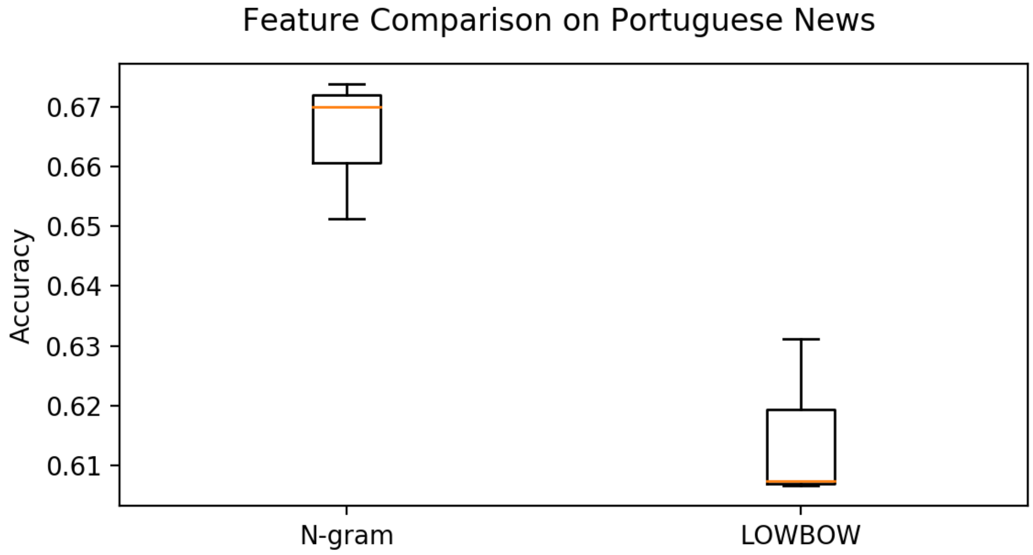


Figure 6.8. Box plot of authorship attribution accuracy for n-gram and LOWBOW features on Portuguese News dataset.

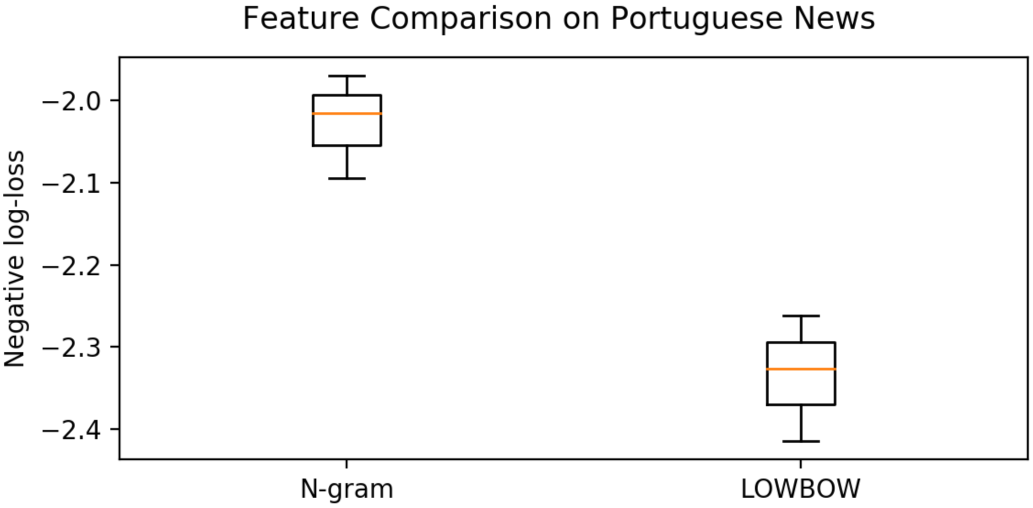


Figure 6.9. Box plot of logarithmic loss values while predicting authorship for n-gram and LOWBOW features on Portuguese News dataset.

6.2.2. TEST-2: Dictionary Feature Selection

If computational complexity is deemed to be important for the system, the dimensionality of the dictionary should be reduced. In this case, how the subspace of full dictionary is determined is an optimisation issue. As seen in Figure 6.10, GFR, LFR, and LDR are compared with each other on the proposed instance-based approach, while number of unique term (k) is increased from 1,000 to 10,000. LFR and LDR are more robust compared to GFR (the lines above the bars) on changing dictionary coverage (the blue bars). On the other hand, the results show that when dictionary size is reached 5000, which covers 39% of all terms existing in the dataset, ranking methods for representing VSM have no superiority to each other (paired t-tests, $p > 0.0001$).

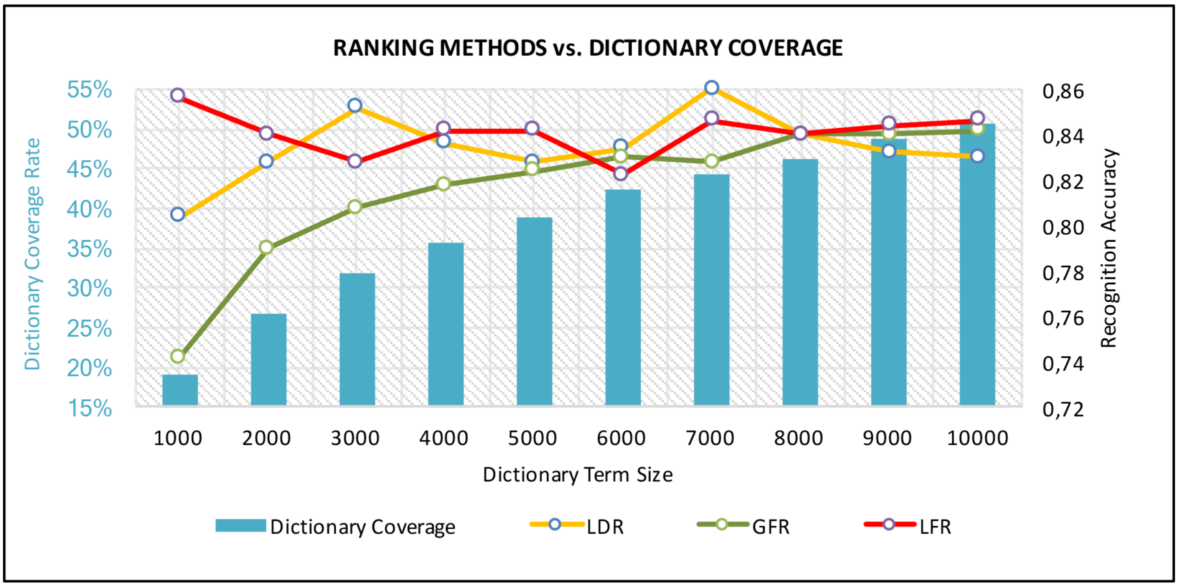


Figure 6.10. Average cross-validation accuracies with different rankings & dictionary size for C10 dataset (6-gram char. features, TF-IDF weighting, ELM parameters: $\lambda = 230$, $\alpha = 0,7$, $\omega = 0,9$).

6.2.3. TEST-3: Feature Weighting

Weighting schemes used in the experiments are (i) term frequency - inverse document frequency (TF-IDF), (ii) sublinear term frequency - inverse document frequency (sTF-IDF), and (iii) entropy - logarithmic term frequency (Entropy-Log). For author attribution, Layton *et al.*'12 [51] used TF-IDF weighting, namely the inverse author frequency (IAF) scheme and reached promising results. In a similar manner, VSM weighting with sTF-IDF gives better cross-validation accuracy results on C10 database compared to the test pipeline without weighting. On the other hand, TF-IDF outperforms other weighting methods for Ekşisözlük dataset which means that performance of VSM weighting is strongly dependent upon dataset, as shown in Table 6.2. Moreover, we observe on COPA dataset that test pipeline with weighting methods don't outperform the non-weighted one in case of data concatenation for each author. The reason for this, global weighting reduces the effect of terms used by many authors, and as corpus size is increased, even rare words are used by multiple authors, thus reducing their discriminativeness. For instance, about 85% of terms in COPA are weighted with 0 in our case. On the other hand, if a training corpus has a limited amount of text for each author, as C10 or Ekşisözlük dataset have, weighting scheme may lead to remarkable improvement on the recognition rate.

Table 6.2. Comparison of weighting schemes with changing character n-grams on C10 and Ekşisözlük datasets. (Dictionary size: 5,000 with local frequent ranking, ELM parameters: $\lambda = 230$, $\alpha = 0,7$, $\omega = 0,9$)

	C10 Database				**Ekşisözlük Database**			
	n=3	**n=4**	**n=5**	**n=6**	**n=3**	**n=4**	**n=5**	**n=6**
No Weighting	0.834±0.027	0.848±0.034	0.838±0.031	0.838±0.021	0.646±0.060	0.668±0.054	0.644±0.062	0.605±0.062
TF-IDF	0.820±0.026	0.844±0.029	0.846±0.031	0.844±0.031	**0.719**±0.066	**0.715**±0.070	**0.683**±0.069	**0.654**±0.071
sTF-IDF	**0.856**±0.029	**0.850**±0.036	**0.864**±0.027	**0.858**±0.032	0.581±0.061	0.546±0.070	0.515±0.061	0.499±0.056.
Log-Entropy	0.826±0.037	0.840 ±0.038	0.850±0.036	0.848±0.036	0.578±0.058	0.516±0.062	0.517±0.060	0.509±0.064

6.2.4. TEST-4: Subspace Projection and Dimensionality Reduction

In order to determine appropriate number of components without missing significant data, we created scree graphs for the datasets, as shown in Figure 6.11. By taking the top 50 eigenvectors, more than 99.95% of data can be represented in the projected subspace for both COPA and Ekşisözlük. However, this reduction causes a significant performance loss for proposed approaches as examined below:

6.2.4.1. Dimensionality Reduction for IBA. As explained in the pipeline shown in Figure 3.1 and in the Algorithm A.2, we are projecting VSM of the author dataset into a subspace by means of LSA. Let a baseline pipeline be built without LSA. Thus, including the LSA with changing number of components into the pipeline has following effects on Ekşisözlük and C10 datasets shown in Figure 6.12 and 6.13:

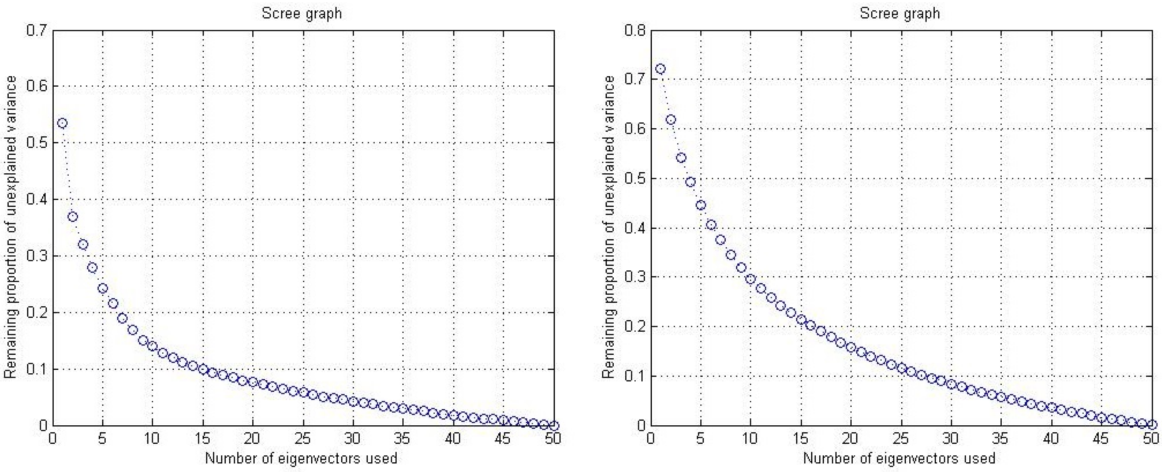


Figure 6.11. Scree graph for Ekşisözlük (left) and COPA (right).

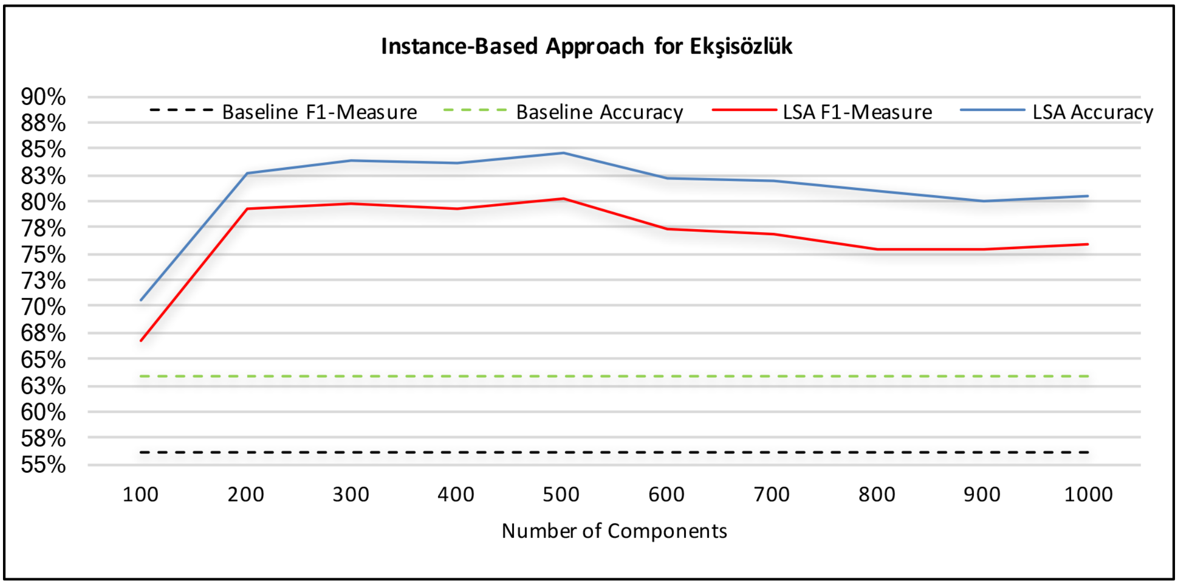


Figure 6.12. Accuracy comparison for varying number of LSA components on Ekşisözlük (Dictionary size=$3,000$ with LFR, n=4, no-weighting, ELM: $\lambda = 250$, $\alpha = 0,5$, $\omega = 0,5$).

For Ekşisözlük, the best accuracy and F_1-measure are accomplished if the number of component used for projecting VSM into a lower dimension is set to 500, where the test accuracy is 84.2% and F_1-measure is 80.3% for the given pipeline parameters in Figure 6.12. On the contrary, if the number of LSA components is below 200, significant performance loss occurs.

Similarly, for C10 dataset, even though the existence of LSA increases the recognition performance compared to the baseline pipeline, reducing number of LSA components results in diminishing accuracy and F_1-measure curves as illustrated in Figure 6.13. The best interval for the number of LSA components is $300 - 700$, where the recognition score draws a relatively flat curve. Moreover, the optimum number of components is 700, where the recognition accuracy is 79.0% for the given pipeline parameters.

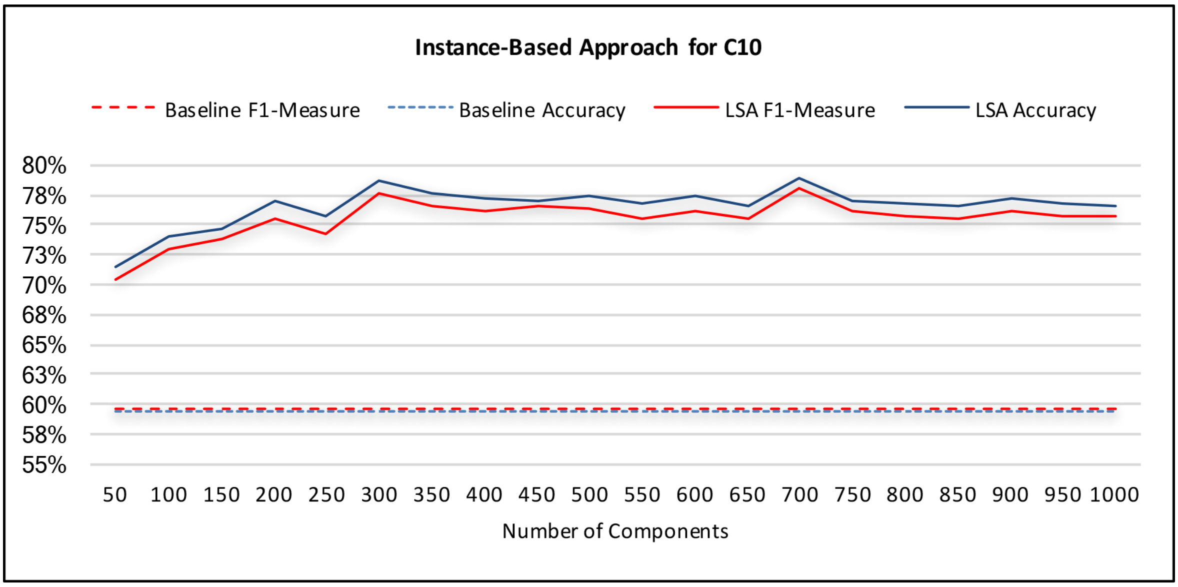


Figure 6.13. Accuracy comparison for varying number of LSA components on C10 (Dictionary size=$5,000$ with LFR, n=6, no-weighting, ELM: $\lambda = 250$, $\alpha = 0,5$, $\omega = 0,5$).

6.2.4.2. Dimensionality Reduction for PBA. As mentioned in the pipeline shown in Figure 3.2 and in Algorithm A.1, the pipeline of the profile-based approach consists of a subspace projection step via PCA or ICA. If we define a baseline pipeline by removing subspace projection and using only the following variable (i) dictionary size of $10,000$, (ii) n-gram size of 4, and (iii) local distinctive ranking, it is actually the same as the RLP method [36]. Thus, we can observe the performance improvements as a result of PCA and ICA on datasets by comparing them with the baseline RLP.

In case of the COPA dataset, the baseline recognition rate is 95.8%. Adding PCA to the pipeline with all the eigenvectors results in 98.5% recognition rate; adding

ICA with all independent components gives 96.3% recognition rate, as illustrated in Figure 6.14. For PCA, after 200 eigenvectors the recognition performance saturate. On the other hand, for ICA, 150 independent components are sufficient to get a good result without performance loss due to dimensionality reduction. On COPA dataset, PCA is prominently better than ICA for dimensional sub-projection.

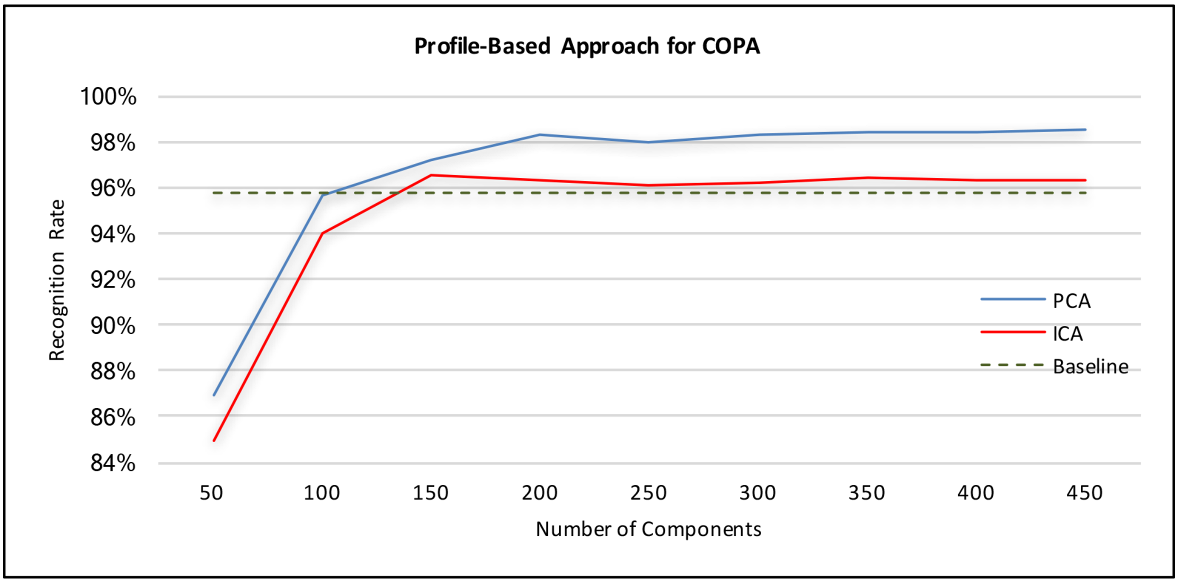


Figure 6.14. Comparison of PCA and ICA on COPA dataset for varying number of components (Dictionary size=10,000 with local distinctive ranking, n=4).

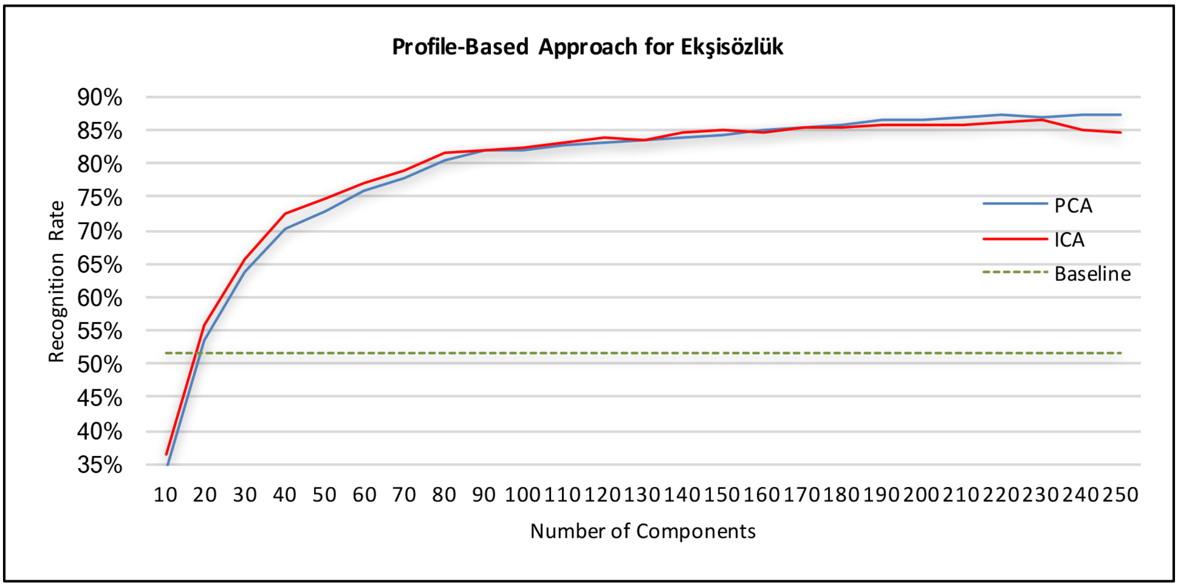


Figure 6.15. Comparison of PCA and ICA on Ekşisözlük dataset for varying number of components (Dictionary size=10,000 with local distinctive ranking, $n = 4$).

For Ekşisözük, the baseline recognition result is 51.6%, and projecting the terms space to a lower dimensionality via PCA or ICA accomplishes remarkable improvements, as seen in Figure 6.15. On the other hand, there is no significant difference between PCA and ICA (in terms of paired t-test, $p > 0.0001$). When the number of components is below 80, ICA shows a better curve, but after 80 components, they have similar patterns. They both reach a saturation level after 200 components. In case of using all components for dimensional projection, PCA and ICA have 87.38% and 84.68% recognition rates, respectively.

6.2.5. TEST-5: Supervised Classifiers

<u>6.2.5.1. Model selection for proposed IBA.</u> Despite preferring ELM in our instance-based pipeline due to its rapid training time, we also compare Naive Bayes classifier, Multi-layer Perceptron, and Support Vector Machines, which are commonly used in the literature. In order to observe performance variations on language domain changes, we conduct experiments on both C10 and Ekşisözlük datasets. During grid search for supervised classifiers, character n-gram type, weighting methods, existence of LSA, number of unique terms (k) in dictionary, and hyper-parameters of each classifier are optimised to get the best performances. As seen in Table 6.3, ELM gives higher average accuracy than other supervised learning methods on Ekşisözlük dataset. On the other hand, both MLP and ELM have the same accuracy, outperforming other learning methods on C10 dataset.

Table 6.3. Cross validation accuracies for classifiers, and the parameters which give the best recognition results.

	Ekşisözlük Database		**C10 Database**	
	Best Accuracy	**Optimum Parameters**	**Best Accuracy**	**Optimum Parameters**
ELM	0.**864**±0.027	ELM:Multiquadric, $\lambda = 240$, $\alpha = 0,2$, $\omega = 1,0$ $n = 4$, $k = 3,000$ weight:TF-IDF, LSA:Yes	**0.876**±0.023	ELM:Multiquadric, $\lambda = 250$, $\alpha = 0,5$, $\omega = 0,7$ $n = 6$, $k = 5,000$, weight:sTF-IDF, LSA:Yes
SVM	0.700±0.030	SVM: linear model, $C = 1$, $n = 5$ $k = 2,000$ weight:TF-IDF, LSA:Yes	0.858±0.029	SVM: linear model, $C = 0.9$, $n = 5$ $k = 5,000$ weight:sTF-IDF, LSA:Yes
NB	0.732±0.055	NB: multivariate Bernoulli, $n = 5$ $k = 2,000$, weight:TF-IDF, LSA:Yes	0.844±0.022	NB: multivariate Bernoulli, $n = 5$ $k = 3,000$, weight:No, LSA:No
MLP	0.852±0.038	MLP: Rectified linear unit, $\lambda = 210$, $n = 4$ $k = 4,000$, weight:TF-IDF, LSA:No	**0.876**±0.035	MLP: Rectified linear unit, $\lambda = 240$, $n = 5$ $k = 5,000$, weight:sTF-IDF, LSA:No

6.2.5.2. Model comparison for stylistic features. Syle-based features mentioned in Section 4.4 and their variations have been excessively used in the literature until now. In order to replicate these efforts on different datasets, Naive Bayes classifier, Multi-layer Perceptron, Support Vector Machine, Extreme Learning Machine and Random Forest classifier are used to detect authorship via these features. The supervised algorithms are modeled with their default hyper-parameters in the scikit-learn library, which is a machine learning tool in Python language.

As seen in Figure 6.16, SVM outperforms rest of the learning models in case of author recognition on Ekşisözlük and Portuguese News datasets. Moreover, MLP and SVM give very close accuracy rates for C10 dataset (See Figure 6.18). On the other hand, ELM has very competitive accuracy on C10 and Portuguese datasets, while it gives the worst recognition results on stylistic features of Ekşisözlük.

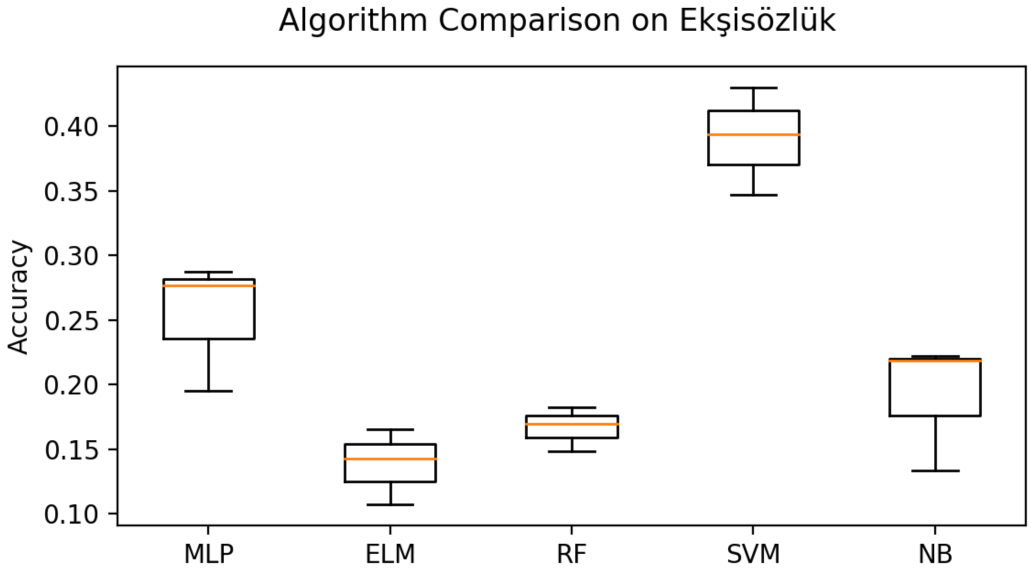


Figure 6.16. Authorship recognition accuracy comparison for Ekşisözlük on changing supervised models with style-based features.

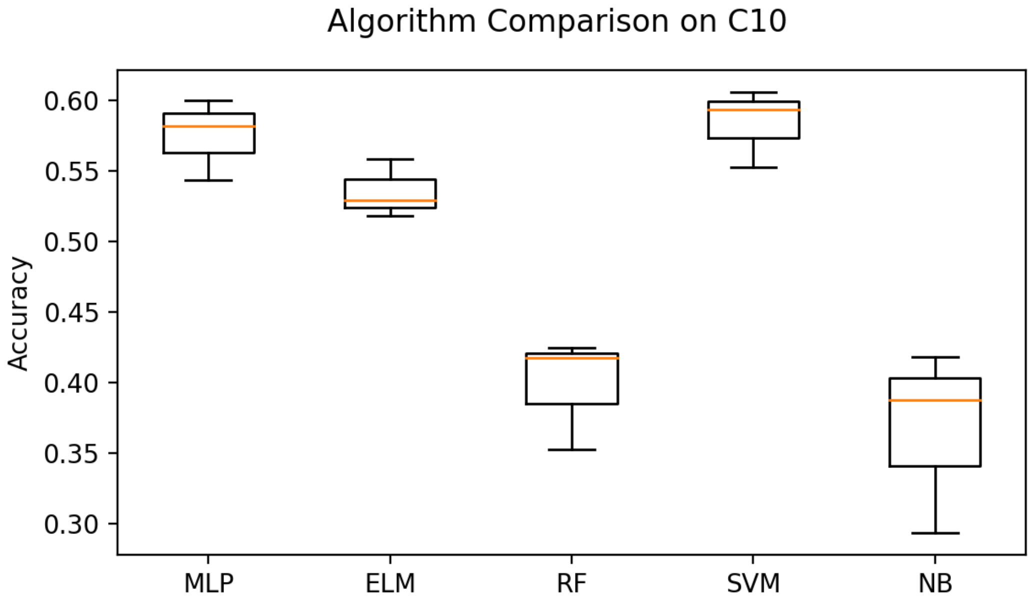


Figure 6.17. Authorship recognition accuracy comparison for C10 on changing supervised models with style-based features.

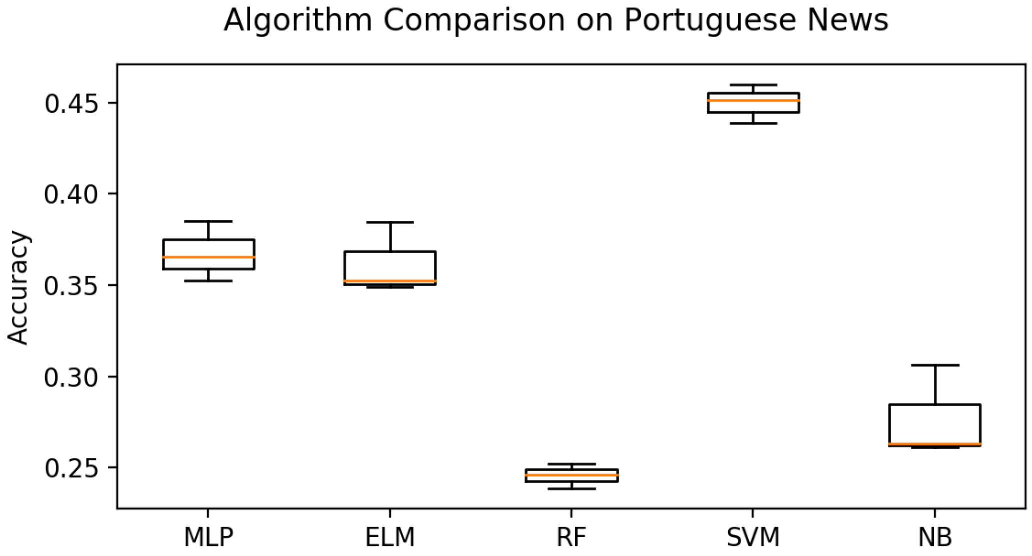


Figure 6.18. Authorship recognition accuracy comparison for Portuguese News on changing supervised models with style-based features.

6.2.6. TEST-6: Benchmarking on the Literature

We validate our recognition methodology both on Turkish and non-Turkish authorship problems. COPA and Ekşisözlük are the two Turkish datasets we use, where

Table 6.4. Cross-validation accuracies for changing supervised models on style-based features of given datasets.

	MLP	**ELM**	**RF**	**SVM**	**NB**
C10	57.7% ±1.2	53.5% ±0.8	38.1% ±2.6	58.4% ±1.1	36.7%±2.1
Ekşisözlük	23.3%±1.8	13.8%±1.1	17.6% ±1.2	39.0%±1.7	19.1%±2.0
Portuguese News	36.7%±0.6	36.1%±0.8	24.5%±0.2	45.0%±0.4	27.7%±1.0

author entries are concatenated to create enriched instances. In these datasets, our profile-based methodology outperforms some well-known profile-based and instance based approaches in the literature, as seen in Table 6.5. Moreover, the superiority of our recognition pipeline over previous efforts in Turkish [24, 29] is also shown.

Table 6.5. Comparison of some profile-based approaches in the literature. (For COPA and Ekşisözlük, each 50 documents are concatenated to get enriched instances; for C10 and Portuguese News, each separate document is a single instance.)

	Proposed PBA		**SCAP** [28]		**CNG** [21]		**RLP** [51]	
	$\mathrm{F}_1 Score$	Accuracy	$\mathrm{F}_1 Score$	Accuracy	$\mathrm{F}_1 Score$	Accuracy	$\mathrm{F}_1 Score$	Accuracy
COPA	**98.4%**	**98.5%**	80.4%	76.6%	95.1%	96.1%	95.6%	95.8%
Ekşisözlük	**89.2%**	**87.9%**	72.9%	72.8%	71.7%	71.5%	45.6%	51.6%
C10	71.3%	73.2%	73.4%	74.2%	71.4%	72.2%	69.6%	70.6%
Portuguese News	82.5%	81.9%	75.9%	73.3%	75.4%	74.5%	73.9%	72.3%

On the other hand, for non-Turkish datasets, where each essay is treated as an instance, very promising results are obtained with our instance-based approach, as shown in Table 6.6. The proposed methodology exceeds the most influential author identification methods reproduced in the works of Potthast *et al.*, where best recognition accuracy (of 15 approaches) on the C10 dataset was noted as 76.6% [66]. We have surpassed the noted results with 87.6%(±2.3%) cross validation accuracy and 81.2% test accuracy on the C10 dataset. Moreover, the success of our profile-based and instance-based approaches is repeated on the Portuguese News database.

The highest accuracies and F_1-measures have been achieved on the proposed PBA and IBA, with the pipeline steps and their values as given in Table 6.7.

Table 6.6. Comparison of some instance-based approaches in the literature. (For COPA and Ekşisözlük, each 50 documents are concatenated to get enriched instances; for C10 and Portuguese News, each separate document is a single instance.)

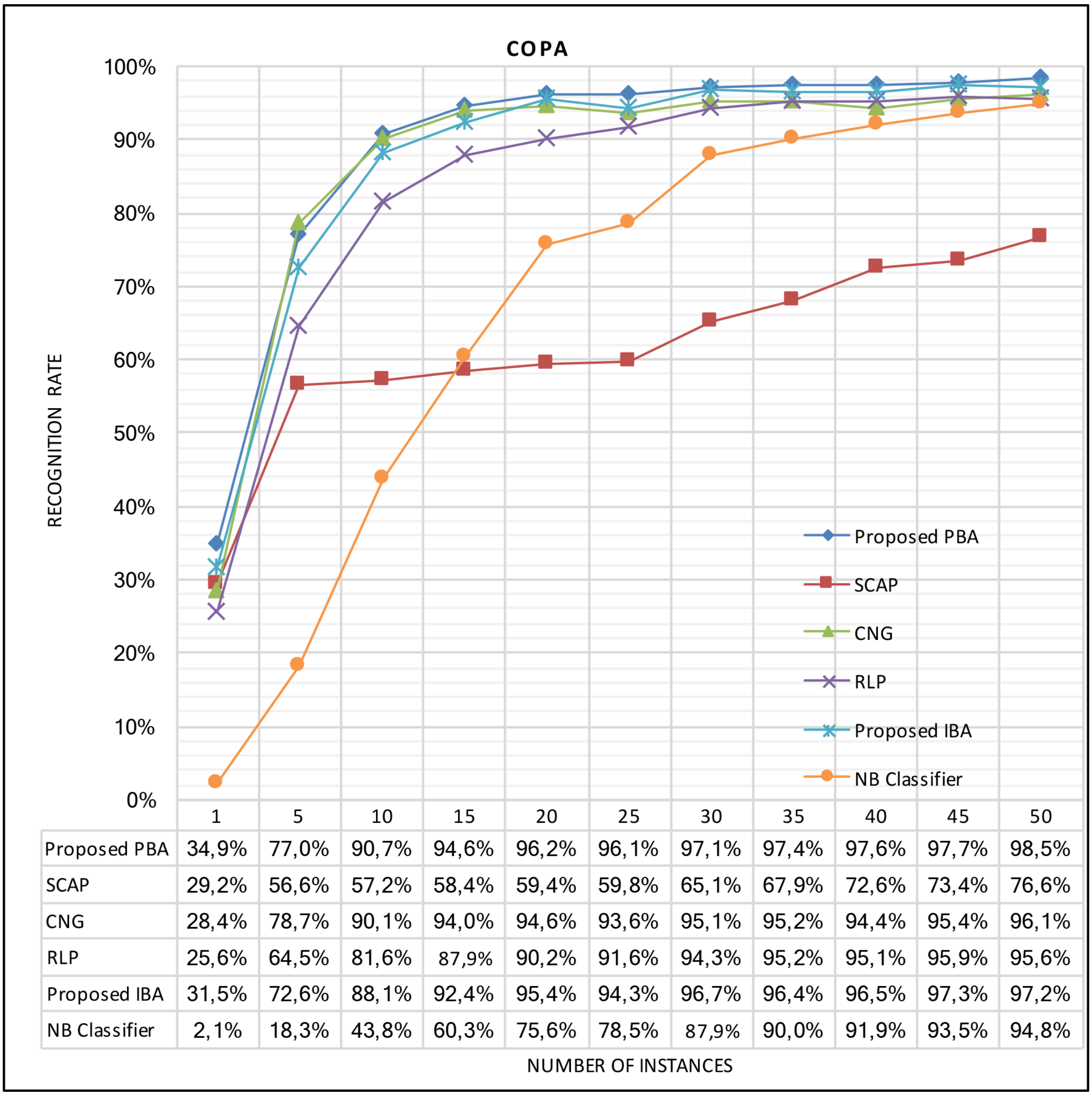
	Proposed IBA		**NB Classifier** [24]		**LOWBOW** [34]		**Stylometry-based Attribution** [29]	
	$F_1 Score$	Accuracy	$F_1 Score$	Accuracy	$F_1 Score$	Accuracy	$F_1 Score$	Accuracy
COPA	96.4%	97.2%	93.3%	94.7%	N/A	N/A	N/A	N/A
Ekşisözlük	84.6%	86.0%	82.3%	82.7%	N/A	N/A	27.0%	34.1%
C10	**81.8%**	**81.2%**	76.9%	77.2%	75.7%	77.4%	51.2%	50.8%
Portuguese News	**83, 7%**	**83.3%**	79.6%	75.9%	65.6%	65.7%	44.1%	46.0%

Table 6.7. The parameters of the pipeline steps for proposed PBA and IBA which give the best author recognition results on the datasets

	Dataset Tested	**Feature Type**	**Dictionary Size**	**Dictionary Selection Type**	**Feature Weighting**	**Subspace Projection**	**Supervised Learning**	**Similarity Measure**
Proposed IBA	COPA	character 3-gram	5,000	Local Frequent Ranking	No	LSA	ELM	N/A
	Ekşisözlük	character 4-gram	3,000	Local Distinctive Ranking	TF-IDF	LSA	ELM	N/A
	Portuguese News	character 6-gram	6,000	Local Distinctive Ranking	TF-IDF	LSA	ELM	N/A
	C10	character 6-gram	5,000	Local Frequent Ranking	sTF-IDF	No	MLP	N/A
Proposed PBA	COPA	character 4-gram	2,000	Local Frequent Ranking	No	PCA	N/A	Cosine
	Ekşisözlük	character 3-gram	2,000	Local Frequent Ranking	No	PCA	N/A	Cosine
	Portuguese News	character 5-gram	3,000	Local Distinctive Ranking	No	PCA	N/A	Cosine
	C10	character 6-gram	3,000	Local Distinctive Ranking	No	PCA	N/A	Cosine

To elaborate on the COPA dataset, the proposed PBA and IBA share similar patterns with CNG and RLP, while concatenated instance size is increased from 1 to 50 as seen in Figure 6.19. These methods can reach a saturation level after instance size of 25, while Naive Bayes classification is slower to reach their levels. Nevertheless, if only one instance is queried, proposed PBA and IBA are significantly superior to other methods: They have 34.9% and 31.5% Rank-1 accuracy, respectively, while these rates are only 29.2%, 28.4% and 25.6% for SCAP, CNG and RLP methods.

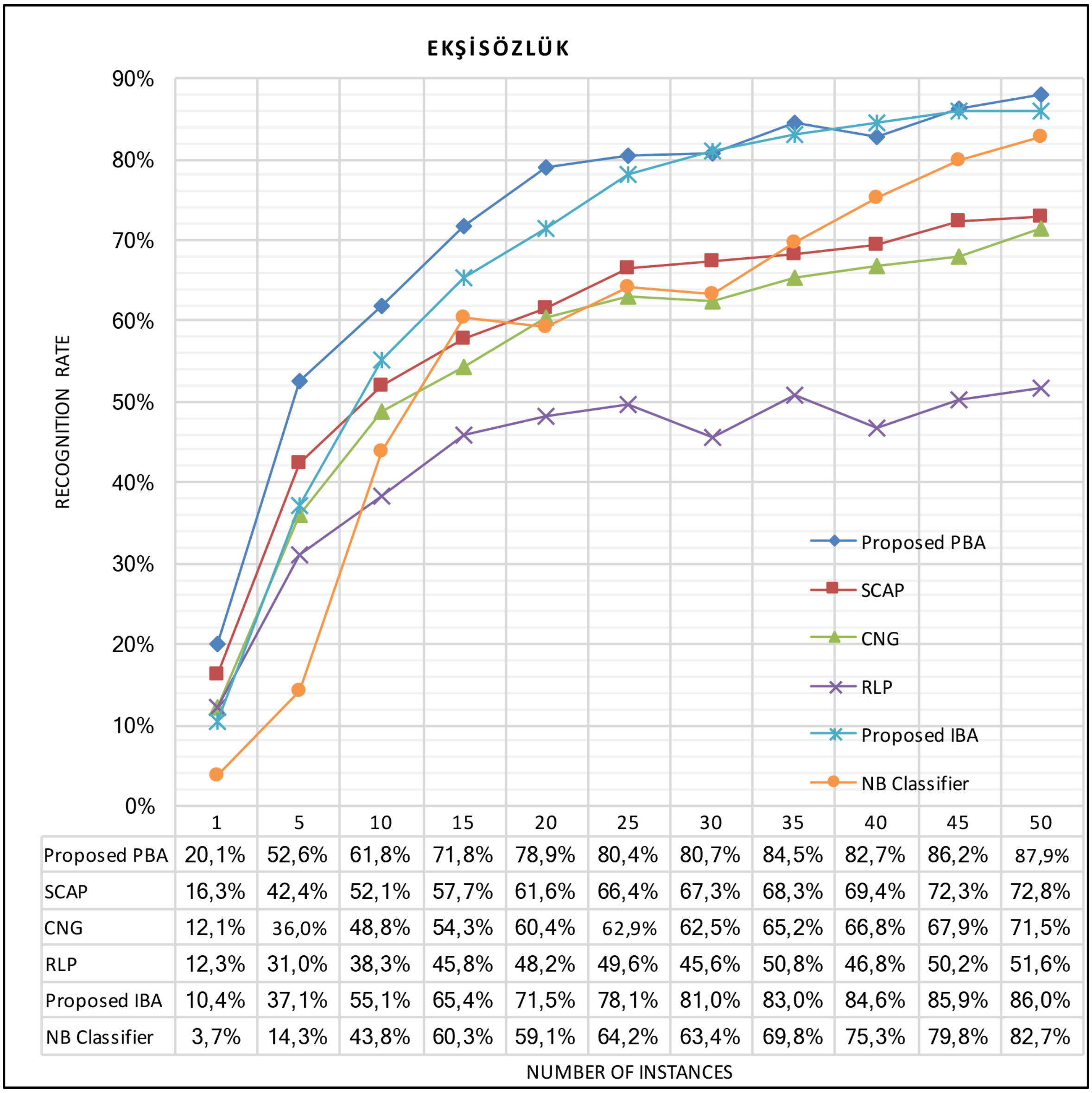
The proposed PBA and IBA approaches surpass the rest of the authorship attribution approaches tested on the Ekşisözlük dataset, as illustrated in Figure 6.20. The interesting thing here is that RLP gives the worst result contrary to its promising pattern on COPA. The reason for this is that the writing purpose of Ekşisözlük authors are generally to define or discuss some phenomena on daily life by using objective



	1	5	10	15	20	25	30	35	40	45	50
Proposed PBA	34,9%	77,0%	90,7%	94,6%	96,2%	96,1%	97,1%	97,4%	97,6%	97,7%	98,5%
SCAP	29,2%	56,6%	57,2%	58,4%	59,4%	59,8%	65,1%	67,9%	72,6%	73,4%	76,6%
CNG	28,4%	78,7%	90,1%	94,0%	94,6%	93,6%	95,1%	95,2%	94,4%	95,4%	96,1%
RLP	25,6%	64,5%	81,6%	87,9%	90,2%	91,6%	94,3%	95,2%	95,1%	95,9%	95,6%
Proposed IBA	31,5%	72,6%	88,1%	92,4%	95,4%	94,3%	96,7%	96,4%	96,5%	97,3%	97,2%
NB Classifier	2,1%	18,3%	43,8%	60,3%	75,6%	78,5%	87,9%	90,0%	91,9%	93,5%	94,8%

Figure 6.19. Recognition rates vs. number of instances concatenated on COPA.

statements, and obeying grammatical rules to some extent. Moreover, the nature of the platform, which results in some writing attitudes and sentence structures to be common among authors, suppress authors from having totally divergent styles. For that reason, RLP, which is based on dissimilarity measurement of local distinctive features, might not extract sufficiently diverse features for author profiles. By depending on the results shown in Figures 6.19 and 6.20, we can also say that the proposed PBA and IBA are more robust to domain changes on authorship attribution, while the accuracy of RLP and CNG are not stable to such changes.



	1	5	10	15	20	25	30	35	40	45	50
Proposed PBA	20,1%	52,6%	61,8%	71,8%	78,9%	80,4%	80,7%	84,5%	82,7%	86,2%	87,9%
SCAP	16,3%	42,4%	52,1%	57,7%	61,6%	66,4%	67,3%	68,3%	69,4%	72,3%	72,8%
CNG	12,1%	36,0%	48,8%	54,3%	60,4%	62,9%	62,5%	65,2%	66,8%	67,9%	71,5%
RLP	12,3%	31,0%	38,3%	45,8%	48,2%	49,6%	45,6%	50,8%	46,8%	50,2%	51,6%
Proposed IBA	10,4%	37,1%	55,1%	65,4%	71,5%	78,1%	81,0%	83,0%	84,6%	85,9%	86,0%
NB Classifier	3,7%	14,3%	43,8%	60,3%	59,1%	64,2%	63,4%	69,8%	75,3%	79,8%	82,7%

Figure 6.20. Recognition rates vs. number of instances concatenated on Ekşisözlük.

6.2.7. TEST-7: Limited Text for Recognition

We investigate the performance of the proposed PBA under the assumption that very limited text exists per gallery author. Figure 6.21 and 6.22 illustrate the CMC curves for the case where a single document is used per author.

On Ekşisözlük dataset, whilst performance of SCAP remained far behind the proposed PBA with the increasing number of concatenated instances as seen in Figure 6.20, they have similar CMC curves after Rank-25 as illustrated in Figure 6.21.

Even so, our approach shows a better curve than RLP and CNG.

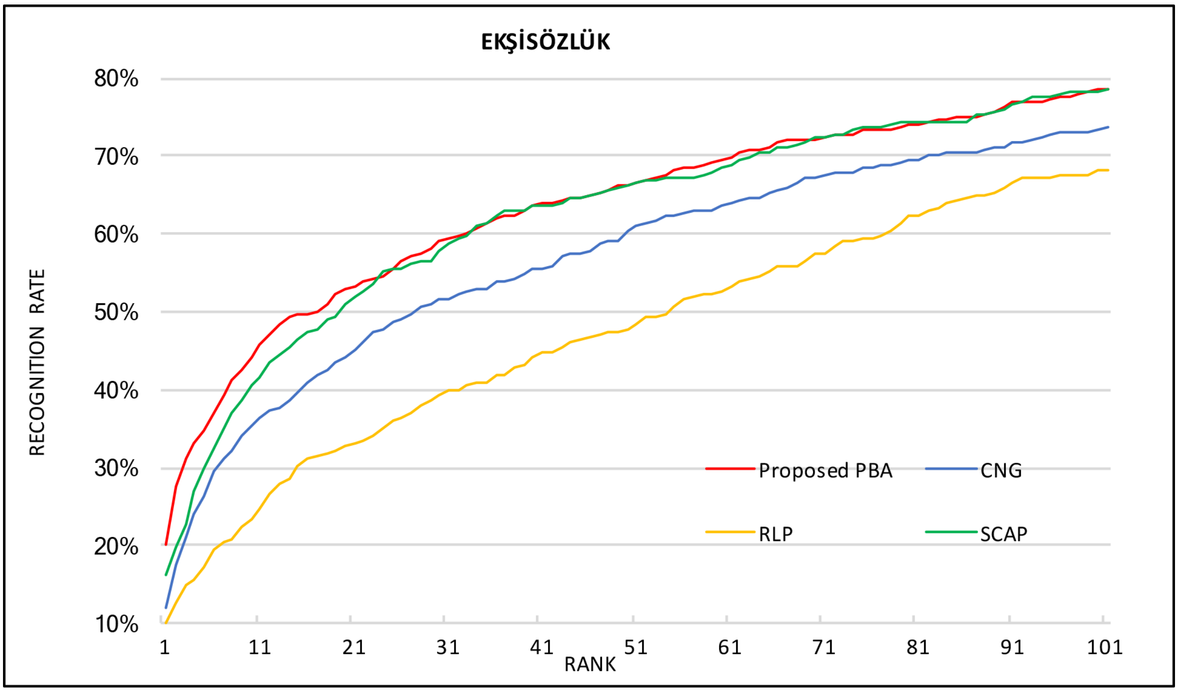


Figure 6.21. CMC curves for Ekşisözlük under the assumption of only one text for each gallery author.

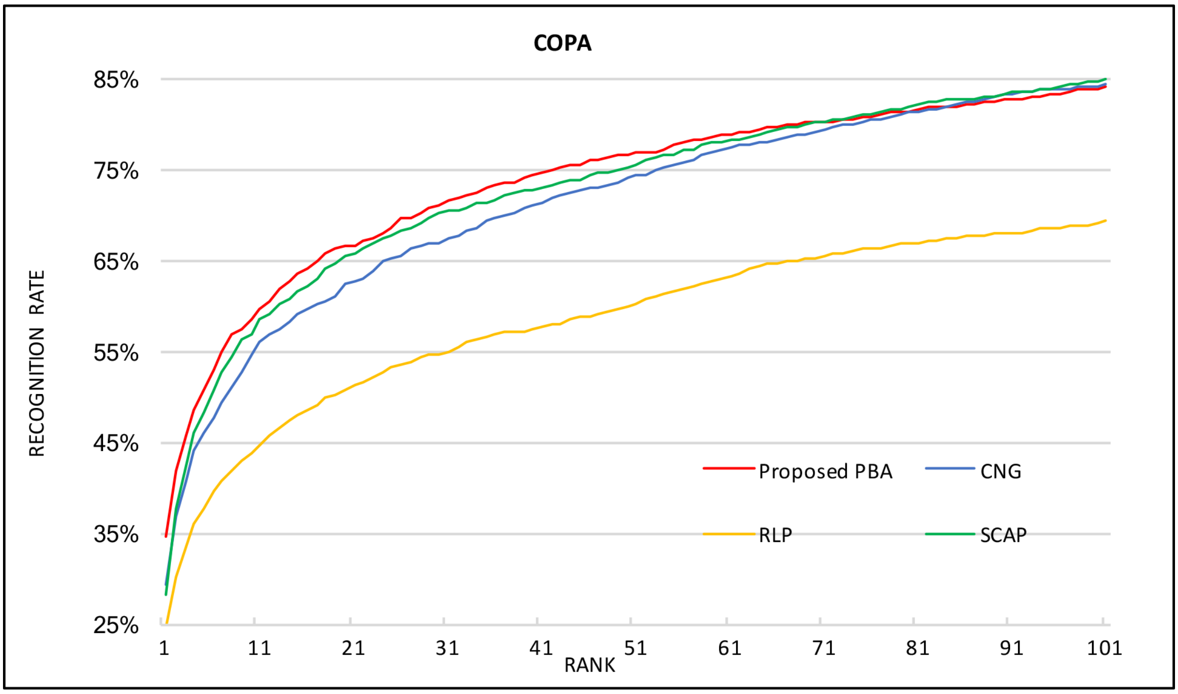
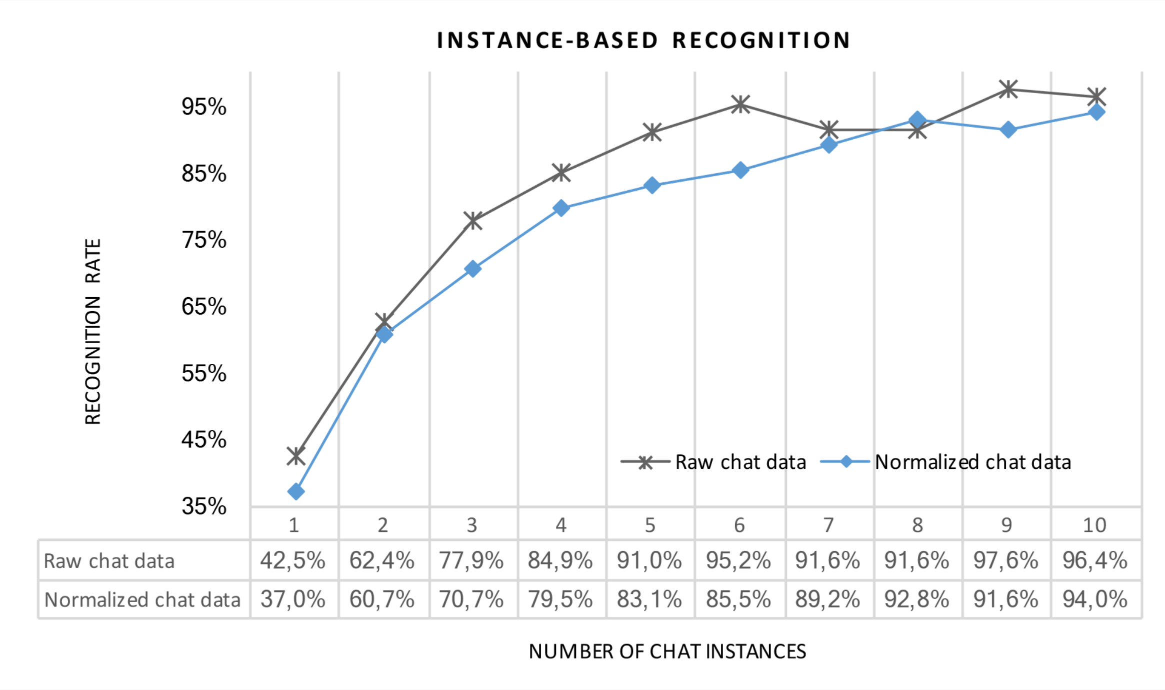


Figure 6.22. CMC curves for COPA under the assumption of only one text for each gallery author.

The proposed approach also has a significantly outperforming curve until Rank-75 on COPA dataset as seen in Figure 6.22. After that, CNG and RLP draw slightly better CMC curves. On the other hand, despite the fact that RLP and our approach have similar recognition patterns in Figure 6.19, CMC performance of RLP is at the lowest degree.

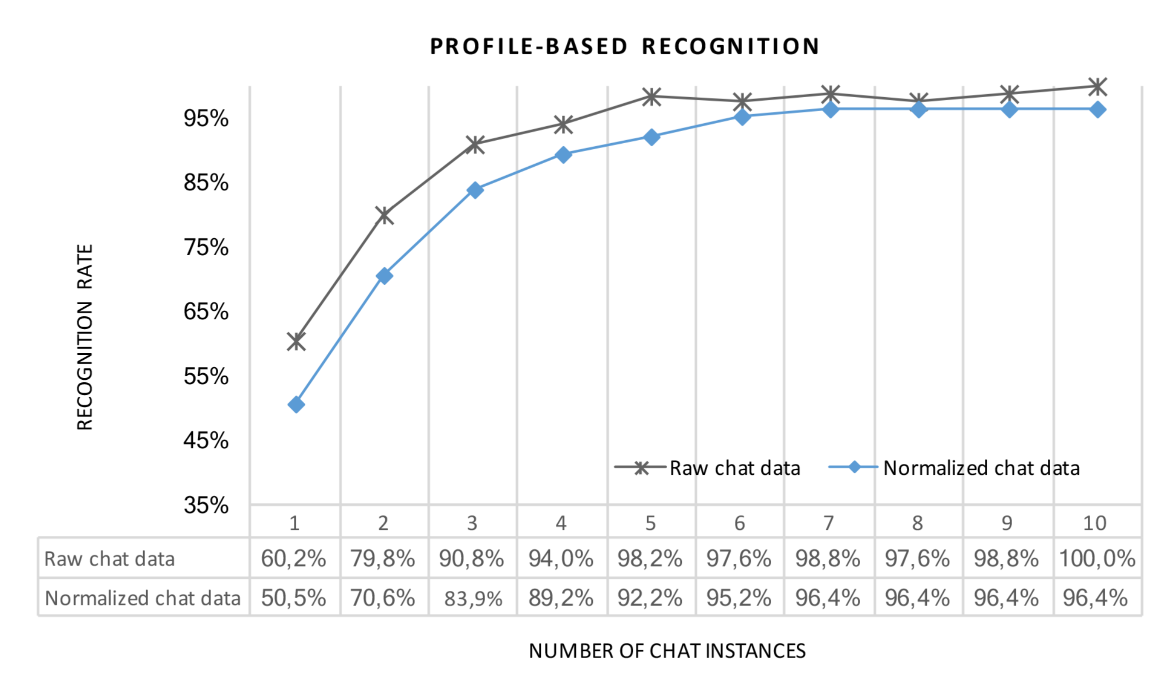
6.2.8. TEST-8: Text Normalization

One of the issues we investigate with this study is how text normalization impacts author identification from Turkish chat records. Our results show that raw chat data is more distinctive than normalised chat data, since intentional misspellings or unconscious typos are some of the most important features for identification. Normalization of text causes loss of these distinguishing features [7]. The impact on the results is evident in the Figures 6.23 and 6.24, which report the raw and normalised versions of each test setting. By performing a paired t-test, we also confirmed that the difference is statistically significant with $p < 0.0001$. On the other hand, the accuracy loss due to normalisation seems to be getting insignificant with the increase of chat instances. We used a small set of users for the normalization experiments, and since the effect was very clear, we did not perform normalization on the entire set of users.



	1	2	3	4	5	6	7	8	9	10
Raw chat data	42,5%	62,4%	77,9%	84,9%	91,0%	95,2%	91,6%	91,6%	97,6%	96,4%
Normalized chat data	37,0%	60,7%	70,7%	79,5%	83,1%	85,5%	89,2%	92,8%	91,6%	94,0%

Figure 6.23. Comparison of recognition rates for COPA-NORM before and after the normalisation on the instance-based approach.



	1	2	3	4	5	6	7	8	9	10
Raw chat data	60,2%	79,8%	90,8%	94,0%	98,2%	97,6%	98,8%	97,6%	98,8%	100,0%
Normalized chat data	50,5%	70,6%	83,9%	89,2%	92,2%	95,2%	96,4%	96,4%	96,4%	96,4%

Figure 6.24. Comparison of recognition rates for COPA-NORM before and after the normalisation on the profile-based approach.

7. CONCLUSION

In this study, we have proposed an approach for authorship recognition within the context of chat biometrics. We performed tests with a large database of multiparty chat records in Turkish, which is available upon request for academic purposes, and a novel database collected from the largest Turkish online social network, as well as a Portuguese and English News database.

Our results illustrate that domain-specific optimisation of dictionary size via local ranking of terms, and LSA/PCA projection on the feature set are both important for obtaining accurate systems. We contrasted lexical word and character based features, as well as effects of feature weighting schemes. Character based features appear to be more scalable for this problem, and produced better results. On the other hand, although stylometric features are commonly referred for such efforts in the literature, they are not as efficient as character based features.

Finally, we tested the robustness of the approach to domain variations, by means of the C10 and Portuguese News datasets. We have reached rank-1 recognition rates up to 98.5% and 87.9% on COPA (403 classes) and Ekşisözlük (252 classes) datasets via profile-based approach. On the other hand, 81.2% and 83.3% accuracy rates are reached on Portuguese (100 classes) and English (10 classes) news datasets via instance-based approach. These results imply that profile-based approach is better for author attribution on informal datasets, while instance-based author attribution method outperforms on well-structured and formal textual data.

Our results also indicate that for moderately sized closed sets (i.e. up to 1000 authors), and with a fairly small amount of query text (e.g. with 50 lines), it is possible to identify authors from their online social communications.

REFERENCES

1. Jain, A. K., A. Ross and S. Prabhakar, “An Introduction to Biometric Recognition”, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 14, No. 1, pp. 4–20, 2004.

2. Delac, K. and M. Grgic, “A survey of Biometric Recognition Methods”, *Electronics in Marine, 2004. Proceedings Elmar 2004. 46th International Symposium*, pp. 184–193, IEEE, 2004.

3. Ali, N., M. Price and R. Yampolskiy, “BLN-Gram-TF-ITF as a New Feature for Authorship Identification”, *Academy of Science and Engineering (ASE) BIGDATA/SOCIALCOM/CYBERSECURITY Conference*, 2014.

4. Koray Balci, A. A. S., “Automatic Analysis and Identification of Verbal Aggression and Abusive Behaviors for Online Social Games ”, *Computers in Human Behavior*, Vol. 53, pp. 517 – 526, 2015.

5. Balci, K. and A. A. Salah, “Automatic Classification of Player Complaints in Social Games”, *IEEE Transactions on Computational Intelligence and AI in Games*, 2017.

6. Kucukyilmaz, T., B. B. Cambazoglu, C. Aykanat and F. Can, “Chat Mining: Predicting User and Message Attributes in Computer-Mediated Communication”, *Information Processing & Management*, Vol. 44, No. 4, pp. 1448–1466, 2008.

7. Kuzu, B. K., Rıdvan S. and A. A. Salah, “Authorship Recognition in a Multiparty Chat Scenario”, *4th International Conference on Biometrics and Forensics (IWBF)*, pp. 1–6, IEEE, 2016.

8. Kuzu, R. S. and A. A. Salah, “Chat Biometrics”, *IET Biometrics*, submitted for publication.

9. Gray, A., P. Sallis and S. Macdonell, “Software Forensics: Extending Authorship Analysis Techniques to Computer Programs”, *in Proceedings of 3rd Biannual Conference International Association of Forensic Linguists (IAFL’97)*, 1997.

10. Mendenhall, T. C., “The Characteristic Curves of Composition”, *Science*, pp. 237–249, 1887.

11. Mosteller, F. and D. L. Wallace, “Inference in an Authorship Problem: A Comparative Study of Discrimination Methods Applied to the Authorship of the Disputed Federalist Papers”, *Journal of the American Statistical Association*, Vol. 58, No. 302, pp. 275–309, 1963.

12. Stamatatos, E., “A Survey of Modern Authorship Attribution Methods”, *Journal of the American Society for Information Science and Technology*, Vol. 60, No. 3, pp. 538–556, 2009.

13. Zheng, R., J. Li, H. Chen and Z. Huang, “A Framework for Authorship Identification of Online Messages: Writing-Style Features and Classification Techniques”, *Journal of the American Society for Information Science and Technology*, Vol. 57, No. 3, pp. 378–393, 2006.

14. Stamatatos, E., N. Fakotakis and G. Kokkinakis, “Automatic Text Categorization in Terms of Genre and Author”, *Computational Linguistics*, Vol. 26, No. 4, pp. 471–495, 2000.

15. Zu Eissen, S. M., B. Stein and M. Kulig, “Plagiarism Detection Without Reference Collections”, *Advances in Data Analysis*, pp. 359–366, Springer, 2007.

16. Juola, P., “Authorship Attribution for Electronic Documents”, *Advances in Digital Forensics II*, pp. 119–130, Springer, 2006.

17. De Vel, O., A. Anderson, M. Corney and G. Mohay, “Mining e-mail Content for Author Identification Forensics”, *ACM Sigmod Record*, Vol. 30, No. 4, pp. 55–64,

2001.

18. Sanderson, C. and S. Guenter, "On Authorship Attribution via Markov Chains and Sequence Kernels", *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, Vol. 3, pp. 437–440, IEEE.

19. Šarkute, L. and A. Utka, "The Effect of Author Set Size in Authorship Attribution for Lithuanian", *Nordic Conference of Computational Linguistics NODALIDA 2015*, p. 87, 2015.

20. Potha, N. and E. Stamatatos, "A Profile-based Method for Authorship Verification", *Artificial Intelligence: Methods and Applications*, pp. 313–326, Springer, 2014.

21. Kešelj, V., F. Peng, N. Cercone and C. Thomas, "N-Gram-Based Author Profiles for Authorship Attribution", *Proceedings of the Conference Pacific Association for Computational Linguistics, PACLING*, Vol. 3, pp. 255–264, 2003.

22. Clough, Paul and others, *Old and New Challenges in Automatic Plagiarism Detection*, 2013, `http://ir.shef.ac.uk/cloughie/index.html`, accessed at June 2017.

23. Diri, B. and M. Amasyalı, "Automatic Author Detection for Turkish Texts", *Artificial Neural Networks and Neural Information Processing (ICANN/ICONIP)*, pp. 138–141, 2003.

24. Amasyalı, M. F. and B. Diri, "Automatic Turkish Text Categorization in Terms of Author, Genre and Gender", *Natural Language Processing and Information Systems*, pp. 221–226, Springer, 2006.

25. Zhao, Y., J. Zobel and P. Vines, "Using Relative Entropy for Authorship Attribution", *Information Retrieval Technology*, pp. 92–105, Springer, 2006.

26. Sanderson, C. and S. Guenter, "Short Text Authorship Attribution via Sequence Kernels, Markov Chains and Author Unmasking: An Investigation", *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pp. 482–491, Association for Computational Linguistics, 2006.

27. McCarthy, P. M., G. A. Lewis, D. F. Dufty and D. S. McNamara, "Analyzing Writing Styles with Coh-Metrix", *in Florida Artificial Intelligence Research Society Conference*, pp. 764–769, 2006.

28. Frantzeskou, G., E. Stamatatos, S. Gritzalis, C. E. Chaski and B. S. Howald, "Identifying Authorship by Byte-Level N-Grams: The Source Code Author Profile (SCAP) Method", *International Journal of Digital Evidence*, Vol. 6, No. 1, pp. 1–18, 2007.

29. Tufan, T. and A. K. Görür, "Author Identification for Turkish Texts", *Cankaya University Journal of Arts and Sciences*, Vol. 1, No. 7, 2007.

30. Estival, D., T. Gaustad, S. B. Pham, W. Radford and B. Hutchinson, "Author Profiling for English Emails", *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics (PACLING'07)*, pp. 263–272, 2007.

31. Argamon, S., M. Koppel, J. W. Pennebaker and J. Schler, "Automatically Profiling the Author of an Anonymous Text", *Communications of the ACM*, Vol. 52, No. 2, pp. 119–123, 2009.

32. Koppel, M., J. Schler and S. Argamon, "Authorship Attribution in The Wild", *Language Resources and Evaluation*, Vol. 45, No. 1, pp. 83–94, 2011.

33. Solorio, T., S. Pillay, S. Raghavan and M. Montes-y Gómez, "Modality Specific Meta Features for Authorship Attribution in Web Forum Posts", *IJCNLP*, pp. 156–164, 2011.

34. Escalante, H. J., T. Solorio and M. Montes-y Gómez, "Local Histograms of Charac-

ter N-grams for Authorship Attribution", *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pp. 288–298, Association for Computational Linguistics, 2011.

35. Oliveira Jr, W., E. Justino and L. Oliveira, "Authorship Attribution of Documents Using Data Compression as a Classifier", *Proceedings of the World Congress on Engineering and Computer Science*, Vol. 1, 2012.

36. Layton, R., P. Watters and R. Dazeley, "Recentred Local Profiles for Authorship Attribution", *Natural Language Engineering*, Vol. 18, No. 03, pp. 293–312, 2012.

37. Savoy, J., "Authorship Attribution Based on Specific Vocabulary", *ACM Transactions on Information Systems (TOIS)*, Vol. 30, No. 2, p. 12, 2012.

38. Cristani, M., G. Roffo, C. Segalin, L. Bazzani, A. Vinciarelli and V. Murino, "Conversationally-inspired Stylometric Features for Authorship Attribution in Instant Messaging", *Proceedings of the 20th ACM International Conference on Multimedia*, pp. 1121–1124, ACM, 2012.

39. Seidman, S., "Authorship Verification Using the Impostors Method", *CLEF 2013 Evaluation Labs and Workshop-Online Working Notes*, 2013.

40. Inches, G., M. Harvey and F. Crestani, "Finding Participants in a Chat: Authorship Attribution for Conversational Documents", *Social Computing (SocialCom), 2013 International Conference on*, pp. 272–279, IEEE, 2013.

41. Monaco, J. V., J. C. Stewart, S.-H. Cha and C. C. Tappert, "Behavioral Biometric Verification of Student Identity in Online Course Assessment and Authentication of Authors in Literary Works", *Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE Sixth International Conference on*, pp. 1–8, IEEE, 2013.

42. Roffo, G., M. Cristani, L. Bazzani, H. Minh and V. Murino, "Trusting Skype: Learning the Way People Chat for Fast User Recognition and Verification", *Pro-*

ceedings of the IEEE International Conference on Computer Vision Workshops, pp. 748–754, 2013.

43. Brocardo, M. L., I. Traore, S. Saad and I. Woungang, "Authorship Verification for Short Messages using Stylometry", *Computer, Information and Telecommunication Systems (CITS), 2013 International Conference on*, pp. 1–6, IEEE, 2013.

44. Iqbal, F., H. Binsalleeh, B. C. Fung and M. Debbabi, "A Unified Data Mining Solution for Authorship Analysis in Anonymous Textual Communications", *Information Sciences*, Vol. 231, pp. 98–112, 2013.

45. Rappoport, R. S. O. T. A. and M. Koppel, "Authorship Attribution of Micro-Messages", *Conference on Empirical Methods in Natural Language Processing*, Vol. 3, pp. 1880–1891, 2013.

46. Mikros, G. K. and K. Perifanos, "Authorship Attribution in Greek Tweets Using Author's Multilevel N-Gram Profiles.", *AAAI Spring Symposium: Analyzing Microtext*, 2013.

47. Qian, T., B. Liu, L. Chen and Z. Peng, "Tri-Training for Authorship Attribution with Limited Training Data", *Association of Computational Linguistics (2)*, pp. 345–351, 2014.

48. Seroussi, Y., I. Zukerman and F. Bohnert, "Authorship Attribution with Topic Models", *Computational Linguistics*, Vol. 40, No. 2, pp. 269–310, 2014.

49. Segarra, S., M. Eisen and A. Ribeiro, "Authorship Attribution through Function Word Adjacency Networks", *IEEE Transactions on Signal Processing*, Vol. 63, No. 20, pp. 5464–5478, 2015.

50. Overdorf, R. and R. Greenstadt, "Blogs, Twitter Feeds, and Reddit Comments: Cross-domain Authorship Attribution", *Proceedings on Privacy Enhancing Technologies*, Vol. 2016, No. 3, pp. 155–171, 2016.

51. Layton, R., S. McCombie and P. Watters, "Authorship Attribution of IRC Messages Using Iinverse Author Frequency", *Cybercrime and Trustworthy Computing Workshop (CTC), 2012 Third*, pp. 7–13, IEEE, 2012.

52. Piantadosi, S. T., "Zipf's Word Frequency Law in Natural Language: A Critical Review and Future Directions", *Psychonomic Bulletin & Review*, Vol. 21, No. 5, pp. 1112–1130, 2014.

53. Salton, G. and M. J. McGill, *Introduction to Modern Information Retrieval*, McGraw-Hill, Inc., 1986.

54. Kuzu, R. S., A. Haznedaroğlu and M. L. Arslan, "Topic Identification for Turkish Call Center Records", *2012 20th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, IEEE, 2012.

55. Mavroeidis, D. and M. Vazirgiannis, "Stability Based Sparse LSI/PCA: Incorporating Feature Selection in LSI and PCA", *European Conference on Machine Learning*, pp. 226–237, Springer, 2007.

56. Tibaduiza, D., L. Mujica, M. Anaya, J. Rodellar and A. Güemes, "Principal Component Analysis vs. Independent Component Analysis for damage detection", *Proceedings of the Sixth European Workshop on Structural Health Monitoring*, Vol. 2, pp. 3–6, 2012.

57. Wold, S., K. Esbensen and P. Geladi, "Principal Component Analysis", *Chemometrics and Intelligent Laboratory Systems*, Vol. 2, No. 1-3, pp. 37–52, 1987.

58. Hyvärinen, A. and E. Oja, "Independent Component Analysis: Algorithms and Applications", *Neural networks*, Vol. 13, No. 4, pp. 411–430, 2000.

59. Landauer, T. K., P. W. Foltz and D. Laham, "An Introduction to Latent Semantic Analysis", *Discourse Processes*, Vol. 25, No. 2-3, pp. 259–284, 1998.

60. Huang, G.-B., Q.-Y. Zhu and C.-K. Siew, “Extreme Learning Machine: Theory and Applications”, *Neurocomputing*, Vol. 70, No. 1, pp. 489–501, 2006.

61. Huang, G.-B., H. Zhou, X. Ding and R. Zhang, “Extreme Learning Machine for Regression and Multiclass Classification”, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, Vol. 42, No. 2, pp. 513–529, 2012.

62. Zheng, W., Y. Qian and H. Lu, “Text Categorization Based on Regularization Extreme Learning Machine”, *Neural Computing and Applications*, Vol. 22, No. 3-4, pp. 447–456, 2013.

63. Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, “Scikit-learn: Machine Learning in Python”, *Journal of Machine Learning Research*, Vol. 12, No. Oct, pp. 2825–2830, 2011.

64. Wikipedia, *Ekşisözlük*, 2017, `http://en.wikipedia.org/wiki/Eksi_Sozluk`, accessed at June 2017.

65. Lewis, D. D., Y. Yang, T. G. Rose and F. Li, “Rcv1: A New Benchmark Collection for Text Categorization Research”, *Journal of Machine Learning Research*, Vol. 5, No. Apr, pp. 361–397, 2004.

66. Potthast, M., S. Braun, T. Buz, F. Duffhauss, F. Friedrich, J. M. Gülzow, J. Köhler, W. Lötzsch, F. Müller, M. E. Müller *et al.*, “Who Wrote the Web? Revisiting Influential Author Identification Research Applicable to Information Retrieval”, *European Conference on Information Retrieval*, pp. 393–407, Springer, 2016.

67. VARELA, P. J., *O Uso de Atributos Estilométricos na Identificação da Autoria de Textos*, Ph.D. Thesis, Pontifícia Universidade Católica do Paraná, 2010.

68. Eryigit, G., “ITU Turkish NLP Web Service”, *European Chapter of the Association for Computational Linguistics (EACL 2014)*, p. 1, 2014.

69. Bolle, R. M., J. H. Connell, S. Pankanti, N. K. Ratha and A. W. Senior, "The Relation Between the ROC Curve and the CMC", *Automatic Identification Advanced Technologies, 2005. Fourth IEEE Workshop on*, pp. 15–20, IEEE, 2005.

APPENDIX A: ALGORITHMS

```
Require: $D$, a collection of training documents with known authors
Require: $L$, number of n-gram features for representing author profiles
Require: $Projection$, PCA or ICA projection function
Require: $R$, a ranking method
Require: $Profile$, profile extraction function outlined in Algorithm A.3
  $E \leftarrow Profile(D)$ , the language profile
  for each author $A_i$ of documents in $D$ do
    $P_{A_i} \leftarrow Profile(\{d_k \in D : author(d_k) = A_i\}, E, R, L)$, the profile of the author $A_i$
  end for
  $TR \leftarrow Projection(P_A)$, $TR$ is the dimensional transforming function
  $T^{P_A} \leftarrow TR(P_A)$, projected profiles of known authors
  for each testing document $t_i$ do
    $P_{t_i} \leftarrow Profile(t_i, E, R, L)$,
    $T^{P_{t_i}} \leftarrow TR(P_{t_i})$, projected profile of testing document
    $G_{t_i} \leftarrow$ argmin $CosineDissimilarity(T^{P_{A_j}}, T^{P_{t_i}})$,
  end for
  return G, the guesses for each testing document
```

Figure A.1. Generic Profile-Based Recognition Algorithm

```
Require: $D$, a collection of training documents with known authors
Require: $L$, number of n-gram features for representing author profiles
Require: $Projection$, PCA or ICA projection function
Require: $R$, a ranking method
Require: $Profile$, profile extraction function outlined in Algorithm A.3
  $E \leftarrow Profile(D)$ , the language profile
  for each author $A_i$ of documents in $D$ do
    $P_{A_i} \leftarrow Profile(\{d_k \in D : author(d_k) = A_i\}, E, R, L)$, the profile of the author $A_i$
  end for
  $TR \leftarrow Projection(P_A)$, $TR$ is the dimensional transforming function
  $T^{P_A} \leftarrow TR(P_A)$, projected profiles of known authors
  for each testing document $t_i$ do
    $P_{t_i} \leftarrow Profile(t_i, E, R, L)$,
    $T^{P_{t_i}} \leftarrow TR(P_{t_i})$, projected profile of testing document
    $G_{t_i} \leftarrow$ argmin $CosineDissimilarity(T^{P_{A_j}}, T^{P_{t_i}})$,
  end for
  return G, the guesses for each testing document
```

Figure A.2. Generic Instance-Based Recognition Algorithm

```
Require: $D*$, a set of documents
Require: $L$ (optional), number of n-gram features to choose
Require: $E$ (required only if $L$ is given), a language default profile
Require: $R$ (required only if $L$ is given), a ranking method
Require: $Rank$ (optional), ranking function outlined in Algorithm A.4
  for each document $D_i$ in $D*$ do
    for each feature $f$ do
      $P_f \leftarrow P_f + f(D_i)$, the value of feature $f$ for the document
    end for
  end for
  for each feature $f$ do
    $P_f \leftarrow P_f/|D*|$, normalize frequencies
  end for
  if $L$ is not given then
    return $P$, full profile
  else
    return $P* \leftarrow Rank(P, E, R, L)$ , the profile of the set of documents $D*$
  end if
```

Figure A.3. Profiling a set of documents; algorithm $Profile(D)$

```
Require: $P*$, the profile of a set of documents
Require: $L$ (optional), number of n-gram features to choose
Require: $E$ (required only if $L$ is given), a language default profile
Require: $R$ (required only if $L$ is given), a ranking method
  if $R$ is $LocalDistinctiveRanking$ then
    for each feature $f$ do
      $P_f \leftarrow P_f - E_f$, recenter value
    end for
    $limit \leftarrow sorted(\{absolute(P_f) \forall f \in P\})_L$, $L$th highest absolute value
    $P* \leftarrow (\{P_f \forall f \in P : absolute(P_f) \geq limit\})$,
  else
    if $R$ is $GlobalFrequentRanking$ then
      $limit \leftarrow sorted(\{E_f \forall f \in E\})_L$, $L$th highest value in language profile
    else $\{R$ is $LocalFrequentRanking\}$
      $limit \leftarrow sorted(\{P_f \forall f \in P\})_L$, $L$th highest value in local profile
    end if
    $P* \leftarrow (\{P_f \forall f \in P : P_f \geq limit\})$,
  end if
  return P, the ranked profile of a set documents
```

Figure A.4. Ranking profiles; algorithm $Rank(P, E, R, L)$