AUTHOR AND LANGUAGE PROFILING OF SHORT TEXTS

by

Dijana Kosmajac

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

at

Dalhousie University
Halifax, Nova Scotia
March 2020

Table of Contents

List of Tables

List of Figures

Abstract

Over the past couple of decades, the advancement and growth of digital information and communication technologies have resulted in information explosion and these technologies are profoundly changing all aspects of modern society. The popularization of the Internet and mobile technologies fueled the rise of social media, providing technological platforms for information spreading, content generation, and interactive communication, which has been contributing to the global data growth. Additionally, social media have become one of the main outlets for obtaining information about latest news, people, businesses, services, etc. The research on it has gained traction having in mind the growing interest in the applications and related technical and social science challenges and opportunities. One of the big challenges of the widespread online textual data is the structure and size. Structurally, it is not in proper grammatical form, has slang, emoticons, improper sentences, which is the standard way we communicate daily. Size-wise, the text is usually very short. However, this is not only the case with the online data; medical notes, open-ended survey questions, various old-school maintenance reports are just some of the examples. We particularly focus on the problem of author profiling on short texts in three different domains. Automatic author profiling is a set of methods to determine an author's (or group of authors') gender, age, native language, personality type and similar, which can be useful in different application contexts such as forensics, security, marketing, product personalisation, socio-demographic analyses and so on. In the first task, we explore fine-grained language dialect/variety identification and propose a new feature weighting scheme. In the second task, we work on bot detection on social media and propose a simple, but efficient method based on statistical diversity measures. In the third task, we present some interesting findings on topic modelling in relation to author on open-ended survey questions from the Canadian Longitudinal Study on Aging (CLSA).

List of Abbreviations and Symbols Used

χ^2 chi square.

AAC Arithmetic Average Centroid.

AP Author Profiling.

AVITM Autoencoder Variational Inference for Topic Models is a neural network-based inference algorithm for topic models first introduced in paper [263].

Bi-LSTM Bidirectional Long Short Term Memory.

Bi-RNN Bidirectional Recurrent Neural Network.

BM25 Best Match 25.

BMM Bernoulli Mixture Model is a special case of Mixture Model where variables are binary: 0 or 1..

BoW Bag of Words.

CFC Class Feature Centroid.

CNG Common N-Grams is a distance-based method for classification of sequence data, introduced by authors [135].

CRF Conditional Random Fields.

DARPA Defense Advanced Research Projects Agency is a US Government agency responsible for the development of emerging technologies for use by US military.

DBM Deep Boltzmann Machines - A Boltzmann machine is a network of symmetrically coupled stochastic binary units first introduced in 1983 [112]. DBMs are deep multilayer version of BMs [240].

Dirichlet distribution It is a multivariate probability distribution that describes $k \geq 2$ variables $X_1, ..., X_k$, such that each $x_i \in \{0, 1\}$ and $\sum_{i=1}^{N} x_i = 1$, that is parametrized by a vector of positive-valued parameters $\alpha = (\alpha_1, ..., \alpha_k)$. They are commonly used as prior distributions in Bayesian statistics.

DM Data Mining.

DWT Discrete Wavelet Transform, usually used for analysis, de-noising and compression of signals and images.

earth mover's distance It is a measure of the distance between two probability distributions. In mathematics it is know as Wasserstein metric..

GCN Graph Convolutional Network is a special case of Graph Neural Networks (GNN) which approximate graph structure using convolution. It was originally proposed in the paper [137].

Generalized Pólya Urn It is, in the statistics, a method for sampling. Common example is given with coloured balls in a urn, where the probability of seeing a ball of each color is linearly proportional to the number of balls of that color in the urn. In this model, when a ball of a particular color is sampled, a number of balls of similar colors are put back along with the original ball and a new ball of that color.

i-vector It is an low-dimensional, speaker- and channel-dependent vector representation used in speech recognition..

ICF Inverse Class Frequency.

IDF Inverse Document Frequency.

IR Information Retrieval.

IT Information Technology.

LBCA Lexicon Based Coefficient Attenuation weighting scheme using document length and average document length ratio, proposed in paper [119].

LBFGS Limited Memory Broyden-Fletcher-Goldfarb-Shanno algorithm.

LCS Longest Common Substring is a sequence that appears in the same order and necessarily contiguous in two comparing strings.

LDA Latent Dirichlet Allocation.

LIGA Language Identification using Graph-based Approach.

LIWC Linguistic Inquiry and Word Count is a tool for extracting the various emotional, cognitive, and structural components present in individuals' verbal and written speech samples.

LVI Language Variety Identification.

MI Mutual Information.

ML Machine Learning.

MT Machine Translation.

NHST Null Hypothesis Significance Testing.

NLI Native Language Identification.

NLP Natural Language Processing.

OSN Online Social Network.

PAN Evaluation lab on Plagiarism analysis, Authorship identification, and Near-duplicate detection.

RNN Recurrent Neural Network.

SGD Stochastic Gradient Descent.

SK String Kernels are functions that measure the similarity of string pairs at lexical level.

SNS Social Networking Site.

spike-and-slab It is a Bayesian variable selection technique that is known to be useful when the number of possible predictors is larger than the number of observations. The prior consists of a mixture between two components: the spike, a discrete probability mass at zero; and the slab, a density (typically uniformly distributed) over a continuous domain [188].

SR Speech Recognition.

steganography The practice of concealing a file, message, image, or video within another file, message, image, or video.

SVM Support Vector Machines.

tf-idf Term Frequency – Inverse Document Frequency.

Acknowledgements

Firstly, I would like to express my sincere gratitude to my advisor Prof. Vlado Kešelj for the continuous support of my Ph.D study, for his patience, motivation, and great knowledge. His guidance helped me in the time of research and writing of this thesis. I would also like to thank the rest of my thesis committee: Prof. Evangelos Milios, Prof. Stan Matwin, and Prof. Robert Mercer for their insightful comments and encouragement, but also for the hard questions which incentivized me to widen my research from various perspectives.

Lastly, I am very grateful to my family: my parents Milomir and Branka, my sister Sladjana, and my partner Benjamin for their unconditional love and support.

Chapter 1

Introduction

1.1 Motivation

Over the past couple of decades the advancement and growth of digital information and communication technologies have resulted in an information explosion and they are profoundly changing all aspects of modern society. The society where the creation, distribution, use, integration and manipulation of information has a significant economic, political, and cultural impact is called in literature "information society" [174, 283]. Ubiquitous access is a key characteristic of the underlying enabling platform — the Internet. Further, rapid growth of mobile device technology and increase in processing capabilities are leading to a new era of omnipresent communications systems. It is the era where users are able to obtain the information at any time and any place relying on different electronic devices. Hence, mobile devices are becoming adapters between sources of information (e.g. sensors) and global Internet mobile services (health, education, government, etc.). In such environment, development of architectures, algorithms and protocols is necessary to make the Internet capable of supporting users in access to information. The rapid growth of number of users of Internet is correlated with the amount of data generated. Companies capture huge amounts of information about clients and business operations, and millions of networked sensors are being embedded in the physical world in different types of devices and transport vehicles, sensing, creating, and communicating data. Moreover, the popularization of the Internet and mobile technologies fueled the rise of social media, providing technological platforms for information spreading, content generation, and interactive communication, which has been contributing to the global data growth.

The research interest in machine learning applications to the related technical and social science challenges and opportunities is constantly growing. The research spectrum is very wide and interdisciplinary. Social Networking Sites (SNSs, often

in literature called Online Social Networks — OSNs) and social media have become some of the main outlets for obtaining information about latest news, people, businesses, services, and interacting with other users via messaging, sharing, etc. Such a complex system is interesting to businesses because it gives them an unprecedented opportunity to connect with customers and prospects. Lately, use of social media for political purposes (campaigning, opinion sharing, promoting, etc.) has uncovered a research on its impact on a countries' political processes [29].

Understanding user behaviour on SNSs has become a fundamental issue in social network analysis. Hence, "user profiling" involves building semantics-based user profile (basic information, socio-demographic characteristics, opinions, interests, etc.) from noisy and unstructured (loosely structured) data. The constructed user profiles can have many different applications. For instance, recommender systems can benefit from high quality profile database to provide good recommendations. For companies, user profiling is important for locating potential customers. Another related term, "author profiling" is used in literature [227] to describe a method or set of methods to determine an author's gender, age, native language, personality type and similar. Author profiling is usually used in different application contexts such as forensics, security and marketing. For example, being able to determine the linguistic profile of the author of a problematic text by analyzing the text could be extremely valuable for evaluating suspects in court cases. As mentioned earlier, companies may be interested in predicting, based on the analysis of blogs and online product reviews, what types of clients like or dislike their products. However, use cases involving automated analysis of author traits can have some ethical implications, but this complex and sensitive topic is out of the scope of the thesis.

Speaking broadly and considering very active research in the Natural Language Processing (NLP) domain, there are many "off-the-shelf" tools and frameworks available for a fairly big set of problems. The accuracy of these tools is proven to perform relatively well on the benchmark datasets, depending on the complexity of the problem. Because these methods usually rely on the probability (or frequencies) of the text features, they require long texts to "learn" representative distributions. However, when we apply these methods to short and noisy texts, current NLP tools usually perform worse, additionally because they are characterized as informal, not carefully

edited, and contain grammatical errors, slang, abbreviations, emoticons, etc. This is very different for the corpora built in the NLP tools — they are usually grammatically and structurally consistent. Therefore, the methods need to be adapted to boost the performance on noisy texts. There are known ways to adapt NLP methods to this kind of texts. One way is to perform text normalisation, in such way that it becomes structurally closer to the formal texts. Another way is to retrain the existing models on annotated noisy texts. Depending on application, third way would be the combination of the two approaches.

In this work we present three projects, each aimed to address the aforementioned challenges in the respective domains. The first project focuses on language and variety identification of short texts. First we examine similarities between 44 (40) European languages by employing *Common N-Gram (CNG)* distance method [135]. We additionally investigate the effect of various feature weighting schemes on language identification performance and propose a new one based on CNG. In the second project we explore the problem of bot detection on social media, with the focus on Twitter. We propose simple, yet effective method for bot detection based on statistical diversity measures. The motivation behind it is that genuine human accounts use more diverse type of messages compared to automated accounts. In the third project we move from supervised to unsupervised learning, where we conduct a set of experiments on an open-ended question from survey conducted on 50,000 elderly Canadians who were asked: *"what in their opinion are important factors to age gracefully?"*. We conduct topic modelling techniques and clustering to draw the potential relations between the opinions and different demographic variables, such as age, sex, health conditions and similar.

1.2 General Background

We tackle all the research areas important for this study, going through known methods and approaches for the similar problems. We outlined the relevant work in the domain of short and noisy data, with the accent on social media user-generated data. Graph-based text mining is one of the focal points of the study, as well as the domain adaptation techniques, because the tools that are built for regular, semantically coherent, and error-free texts usually underperform in the case of short texts.

1.2.1 Mining with Short and Noisy Textual Data

Noise in text can be defined as a difference in the surface form from the intended, grammatically correct or original text. Noise in text can be induced in two ways. First, noise can be introduced during an automated conversion process to textual representation from some other form. Conversion from printed or handwritten documents, spontaneous speech, and camera-captured scene images, are some of the examples where computer algorithm results in noisy text. The characteristic of noise in these cases is the deviation of the converted text from the representation of the original signal. Second, noise can be introduced when the text is produced in digital form. Online instant messaging, SMS (Short Messaging Service or texting), emails, social media, blogs and forums, open-ended survey questions, are some of the examples where users produce noise in text. Such text contains grammatical errors, special characters such as emoticons, non-standard or slang word forms, word forms from multiple languages etc.

A considerable amount of research is conducted related to short and noisy texts. One of the reasons is the increasing popularity of short messaging on the Internet. Moreover, many platforms are built around the idea of sharing short snippets of information. Sentiment analysis and opinion mining [158, 178], text classification [260], and text normalisation [15, 105] are some of the very active general short text research topics. It has been observed that algorithms that perform well for larger bodies of text might have decreased performance for short and noisy texts. Derczynski *et al.* [65] presented a part-of-speech (PoS) tagger solution for Twitter posts. They claim that taggers trained for long and grammatically correct texts perform poorly in the conditions of brevity and noise. To enrich the context of a short text (tweet), Guo *et al.* [103] presented a solution where they linked the context of a tweet to the related news.

1.2.2 Author Profiling

The Author Profiling (AP) task, as mentioned earlier, is concerned with determining specific person's characteristics such as gender, age, native language and similar, by analyzing the language usage in groups of authors. It is applied in different

domains such as psychology [213], social media [211, 247], socio-demographic analyses [198, 237], etc. The common processing pipeline consists of three steps: a) textual features extraction, b) documents representation using these features, and c) training a classification model of documents. The first step has received most of attention, where features fall into two groups: content features (words such as nouns, adjectives, verbs, etc.) and stylistic features (function words, PoS tags, punctuation, etc.) [202]. Rosso *et al.* [235] is an excellent survey on author profiling, with the particular focus on Arabic language. However, they give a good overview of the recent and influential developments in the author profiling tasks. They split author profiling into three subtopics: age and gender detection, native language and dialect/variety identification and the deception, irony and sarcasm detection. In Chapters 2 and 3 we will focus more on the research related to language and gender identification.

1.3 Contributions

Having all that laid out, we explore user profiling in three different domains. Under domain-specific settings, the goal is to find alternative (presumably better, in a way) solutions to existing, well-established approaches. The following subtopics are used as guidelines:

Subtopic 1: In the task on Similar Language Identification we tackle a few questions:

- Can a distance-based classification model be used to analyze the training set and provide more insights about the similarity among related languages? Is this automatic model potentially suitable as a tool for comparative linguistics?
- Do different weighting schemes significantly affect the quality of a model?
- Is a distance based weighting scheme appropriate (superior?) compared to traditional weighting schemes?

Subtopic 2: In the task on Bot Identification on Social Media, we explore:

- Can simple measures from Information Theory (Shannon's, Simpson's indices) be used to model user behaviour and to develop a supervised method to distinguish between genuine users and bots?

Subtopic 3: In the task on Topic Modelling on Open-ended Survey Questions, we try to answer:

- Is an Information Retrieval (IR) based method suitable for topic modelling?
- Can it be used to automatically create a participant-to-topic mapping which is suitable for further analyses?

Workshop and conference papers that were the result of this study: [140, 141, 142, 143, 144, 145].

A paper that is in the process of publishing:
"Language Distance using Common N-Grams Approach", INFOTEH 2020 (Jahorina, Bosnia).

1.4 Outline

This rest of this thesis is structured as follows.

Chapter 2 First, we briefly discuss related work, mainly in the domain of language, dialect and language variety identification for short texts. Second, we explore measuring similarity among a set of languages spoken in Europe using Common N-Grams distance [135]. Third, we conduct a comprehensive study on weighting techniques and their impact on classification performance across a set of classifiers. We give task-specific conclusions and potential extensions and questions to investigate in the future.

Chapter 3 In this chapter we conduct a study on bot identification on social media. We discuss the relevant work and try to outline most important directions of research in this domain. Having that the bot and fake news detection solutions on online social networks are increasingly in demand by the companies, diversity in approaches is vast. We address a small fraction of this big set of problems and show that using

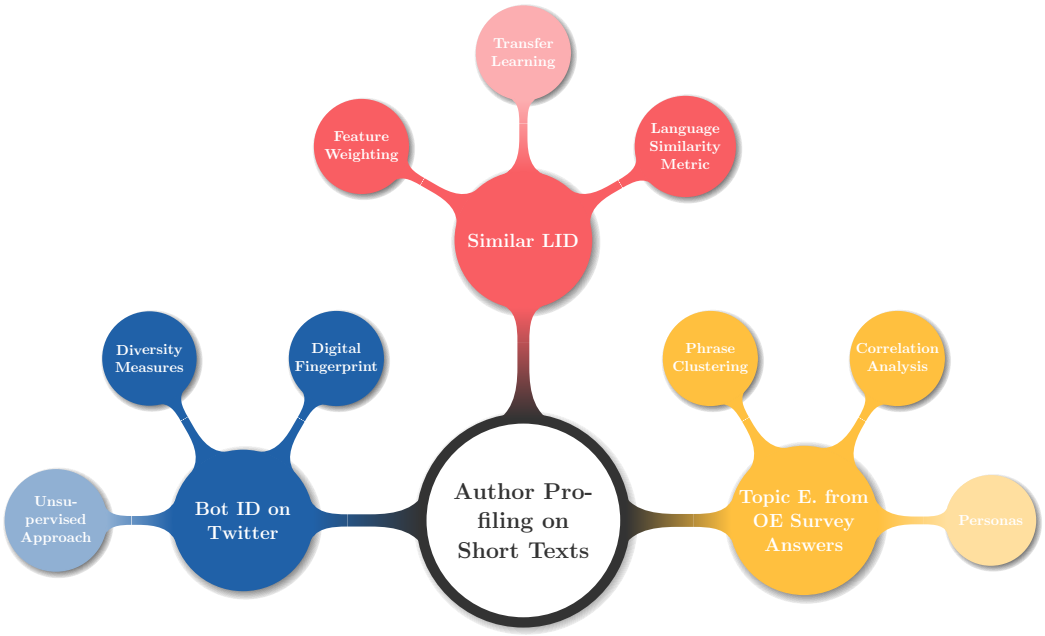

Figure 1.1: Thesis mind map.

simple statistical measures can help in identifying automated accounts. At the end, we lay out the conclusions and propose potential extensions and questions for future work.

Chapter 4 In this chapter we use unsupervised algorithms to explore topics in open-ended survey questions. We compare topic modelling techniques to clustering techniques for the case of very short texts. Although, there have been topic modelling methods specifically developed for this kind of data, we show that clustering combined with knowledge transfer demonstrates better results. The meaning of the "better results" can be questioned, especially in the case when we do not have a gold standard to compare to. Our conclusion is mainly driven by the fact that the result obtained was deemed more appropriate by the domain experts for the purpose of further statistical analyses. At the end, we lay out the conclusions and propose potential extensions and questions for future work.

Chapter 5 In Chapter 5 we give general summary of the study and future directions of the conducted experiments.

The thesis mind map is given in Fig. 1.1.

Chapter 2

Language Identification on Social Media

2.1 Introduction

The unique definition of language does not exist. A couple of them are cited very frequently. Language scholar Henry Sweet defines it as "*the expression of ideas by means of speech – sounds combined into words. Words are combined into sentences, this combination answering to that of ideas into thoughts.*" Two other linguists Bernard Bloch and George L. Trager agreed that "*a language is a system of arbitrary vocal symbols by means of which a social group cooperates.*" According to a famous linguist Noam Chomsky, a language is "*a set (finite or infinite) of sentences, each finite in length and constructed out of a finite set of elements.*" He further claims that all natural languages have "*a finite number of phonemes (or letters in its alphabet) and each sentence is representable as a finite sequence of these phonemes (or letters).*"

Regardless of not having a unique definition, the fact is that natural languages are fundamental components of individual and human heritage. They are the enabling "tool" for expressing identity, sharing ideas, and in a broader sense, achieving political, educational and economic autonomy, as well as promoting peace and sustainable human development [289]. Since Information Technologies (IT) became the vessel for all aspects of social, cultural, economic and political life, it is essential to ensure that everyone has access and can contribute with their own content to the multilingual Internet. Hence, ITs can be considered as a tool for promotion of linguistic diversity. In general, the Internet is open to all languages of the world, but only when certain conditions are met, such as having enough human and financial resources. To this end, many world organizations, including UNESCO[1], La Francophonie[2], Union Latine[3] and ANLoc[4] are committed to promoting multilingualism on the Internet.

[1]United Nations Educational, Scientific and Cultural Organization, `http://unesco.org/`
[2]`http://www.francophonie.org/`
[3]`http://www.unilat.org/`
[4]The African Network for Localization, `http://www.africanlocalization.net/`

As previously explained, linguistic diversity on the Internet is of crucial interest. This poses a challenge from the perspective of computer processing in an automated way. Automatic Language Identification (LID) is a task of automatically identifying the language of a spoken utterance or text. It is a very active area of research due to its application in computational sciences, particularly Machine Translation (MT), Speech Recognition (SR) and Data Mining (DM). LID is often the prerequisite for accurate text analytics because natural language models are governed by language-specific interrelated systems such as phonology, morphology, syntax, the lexicon, and semantics. A statistical language model represents a probability distribution over sequences of tokens $(w_1, ...w_n)$. Given the sequence of length n it assigns a probability $P(w_1, \ldots, w_n)$ to the whole sequence. Some of the well-known statistical models include n-gram, maximum entropy and neural network models.

For a human listener (reader) familiar with the language of interest recognizing utterances or texts comes naturally. The aim of automatic LID is to mimic this human ability. Indeed, a number of automatic approaches have been developed that are able to identify language without human intervention. The advantage of automatic systems over an average human is speed and the ability to recognize a handful of languages, like a trained linguistics expert. There has been a significant effort to document and create resources for as many world languages as possible, such as the Ethnologue project [71]. As of 2019, Ethnologue lists 7,111 registered languages (Fig. 2.1). On the other hand, BabelNet [193], a large multilingual semantic network, covers only 284 languages. This illustrates how many of them are poorly documented and underresourced, which may lead to the eventual disappearance of a language. There are other efforts to document languages in a systematic way such as WordNet [185] (English only) and Universal Knowledge Core (UKC) [98]. It is worth mentioning that UKC is an ambitious project to develop a large scale linguistic resource which aims to cover all registered languages of the world.

Different systems of communication are building blocks of different languages; the degree of difference needed to establish a different language is hard to quantify. In practice, the systems of communication are recognized as different if the parties

[5]Source: `https://www.ethnologue.com/guides/how-many-languages`

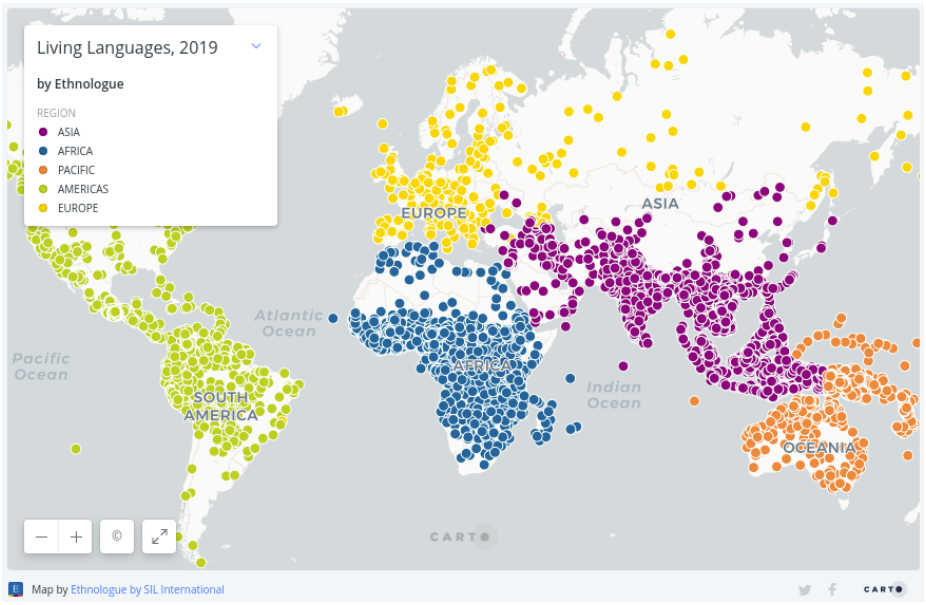

Figure 2.1: Geo-spatial locations of 7,111 world languages and dialects.[5]

cannot understand each other without some learning. Actually, mutual comprehension cannot be expressed in a discrete way, but rather on a scale. To that end, Gamallo *et al.* [93] explore the distance between 44 different European languages and varieties in a quantitative way. Fig. 2.2 from their work, clearly shows the clusters of closely related languages. The measure they used for generating the graph is based on a perplexity measure shown in Eq. (2.1). Perplexity is an evaluation metric for language models used to measure fitness of test data built with n-grams which was, in this case, adapted to measure distance between languages. Let M be a language model with n-gram probabilities $P(\cdot)$ and set of character sequences $T = \{t_1, t_2, ..., t_n\}$.

$$PP(T, M) = \sqrt[n]{\frac{1}{\prod_i^n P(t_i|t_1^{i-1})}} \tag{2.1}$$

N-gram probabilities are defined as in Eq. (2.2). The probability is calculated by dividing the observed count $C(\cdot)$ of a particular character sequence by the count of the prefix of the same sequence (lower rank n-gram):

$$P(t_n|t_1^{n-1}) = \frac{C(t_1^{n-1}t_n)}{C(t_1^{n-1})} \tag{2.2}$$

Final distance matrix is generated by applying Eq. (2.1) pairwise between all languages in the observed dataset $Dist(L_1, L_2) = PP(T_{L_1}, M_{L_2})$. Finally they compared

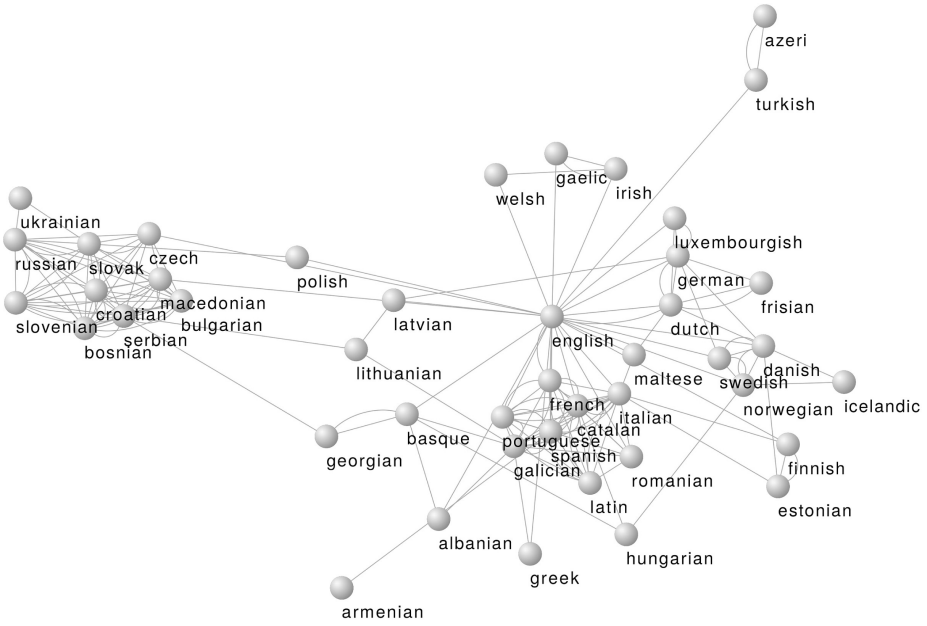

Figure 2.2: Language distances represented as graph. The nodes (languages) are connected by the edges calculated perplexity-based distance on n-gram language models [93].

the results with ground truth (language similarities rated by a linguistics expert) and found a high level of agreement (accuracies in the range from 82.50% to 85% for 2 datasets).

In the sea of challenges and tasks related to human language we particularly choose to explore the possibility of distinguishing closely related languages in short texts. In this chapter we explore the impact of different weighting techniques on performance of classifiers for discriminating similar languages. Although weighting methods were used in this scenario by many authors [315, 54] there is no comprehensive study on feature weighting for this particular task. Hence, our aim is to test the significance of a set of weighting techniques on several datasets used in recent years for LID.

2.2 Related Work

LID have been studied for more than 50 years. At the time of writing there has been several survey papers that cover most relevant research in this domain. Most recent ones are by Garg *et al.* [96], by Qafmolla [224] and the most extensive one by Jauhiainen *et al.* [126].

Mustonen [190] tackled the problem of LID in 1965. They describe statistical methods for discrimination between Swedish, Finnish and English words. The method they used is based on Multiple Discriminant Analysis (MDA) with carefully curated character level features (set of letters that are exclusive to the respective languages, presence of diphthongs, etc.). The accuracy of the approach was 76%, however, it is not clear what training and tests sets consist of, therefore the result is not reproducible. In 1974, Rau explores LID in his Master thesis [229]. The features they used are relative character unigram and bigram frequencies. The final classifier was ensemble of seven simple classifiers using majority voting. The classifiers are built on two measures, Kolmogor-Smirnov's Test [139] and Yule's K characteristic [314]. The accuracy of the system was 89% distinguishing between English and Spanish.

Early works didn't clearly distinguish between written text and spoken utterances. House *et al.* [116] uses phonetic information on transcribed utterances to identify spoken English, Russian, Hindi, Greek, Chinese, Korean and Japanese. They explore different aspects such as the necessary minimum length of a utterance to identify a language. Church *et al.* [52] try to assign accents to words. To apply their method to loan words, they developed a character trigram Bayesian model to identify a language and then apply corresponding syllabic rules. Although they used character n-grams, Beesley [20] were the first to identify it as such.

Likely the most known early work on LID is by Cavnar *et al.* [43]. Their approach is based on language profiles and an "out-of-order" similarity metric on n-gram ranks. They conducted the experiments on 3,478 documents in 8 different languages on the articles collected from Usenet newsgroups. The reported accuracy is 99.8%, which is very high. It is worth mentioning that there is an "off-the-shelf" implementation of their method available online called `TextCat` which currently supports 69 different languages. A part of the reason why their research got high exposure is due to the fact that the implementation is suitable to be used as a baseline.

2.2.1 Text Representation and Features

The usual format of the text for a LID task is a stream of characters. How the text is stored can make a big difference. Character encoding is a way to represent characters

by a standardized encoding system. For example, Braille or Morse are examples of encoding systems. In computer science, some of the most common encodings are ASCII which uses fixed number of bytes (one byte) to represent a character, and UTF-8 which uses variable length to accommodate a larger number of characters. The UTF family of encoding systems are, in general, designed to fit as many languages as possible. The encoding matters a lot when it comes to data pre-processing. Kikui *et al.* [136] use the encoding information as a feature to identify a language. They conducted the experiments on three Eastern Asian languages and six Western European languages. They show that there was no confusion between European and Asian categories, but there was some confusion within the categories. With the respect to encoding, later research has addressed this issue in different ways. Some [17, 146] developed systems that handle multiple encodings, while others [165, 257, 271] used encoding per language as separate models. However, most of research does not take into account the potential issues with the encoding with the assumption that the whole corpus is encoded in the same system.

Words and Characters

Some of the research [111, 244] exploited the fact that some different languages (assuming that they are in the same script family) have language-specific letters. Features such as capitalization, the frequency of punctuation, length of words, *hapax (dis) legomena* are used in distinguishing closely related languages (or dialects) [150, 256]. Character frequency or probability is often used as a feature [229]. Tran *et al.* [282] and Windisch *et al.* [303] used frequency of prefix and suffix characters, respectively. Another approach used is based on weighted character frequency [199, 273]. The approach incorporates variants of Inverse Document Frequency weighting (IDF), which they call inverse class frequency (ICF), Arithmetic Average Centroid (AAC) and Class Feature Centroid (CFC). Takçi *et al.* [273] also use Mutual Information (MI) and chi square (χ^2) as weighting techniques. They [272] also explored the relative character frequencies with discriminating weights. However, most of the recent research focuses on using character sequences rather than single characters. Baldwin *et al.* [17] showed that, in general, character sequences perform better than using statistics of single characters.

Character N-grams and Character Co-occurrences Co-occurrences is another well-explored feature in LID tasks. Lee *et al.* [288] conducted a case study where they used the ratio of question, exclamation characters and the total number of the end of sentence punctuation as features with different ML algorithms. Franco-Salvador *et al.* [88] applied `fastText`[6] character n-gram embeddings [31] in conjunction with neural network classifier. Use of characters (including letters and some non-letter characters) and complex character repetitions (handled with regular expressions) are common character-based features found in literature [18, 73, 255]. Barman [19] used a combination of lexicon words, word lengths and character n-grams, where $n \in \{1, 2, 3, 4, 5\}$.

Character n-grams are another way of text representation that is used very often in the NLP tasks, especially the ones that are low level (where there is no need for semantic inference, or complex parsing). Character n-grams represent a continuous sequence of characters with the length of n. They can be overlapping or non-overlapping. For example, the word *"sample"* can be represented as [“sa'', “mp'', “le''] if the n-grams are non-overlapping, or more common in the literature are overlapping ones, and the example is [“sa'', “am'', “mp'', “pl'', “le'']. Zamora *et al.* [315] showed that feature weighting scheme based on MI applied on character n-grams worked well on the evaluation dataset provided by the TweetLID workshop [325] organizers.

Word N-grams and Word Co-occurrences Words are also commonly used as features in LID tasks in various ways. A position of a word (as a part of *CRF* model) has been used in the case of code-switched corpora [70, 147]. Basic dictionary is often used as a training corpus [293]. The dictionaries were constructed in various ways, such as using only stop words [300], or most relevant language words [230], function words [269], words that convey modality [5], etc. In general, many researchers agree that a combination of word and character features can boost the performance [126].

Word and Character Embeddings Word and character embeddings became a popular (and successful) choice for text representation in many NLP tasks because of its ability to encode high level relations between the tokens (semantic, syntactic similarity). It particularly became prominent with the breakthrough of

[6] `https://github.com/facebookresearch/fastText`

"word2vec" *et al.* [183], paving the way towards developing better performing representations [31, 153, 259]. There are a number of recent studies that take advantage of word and character embedding representations. Jaech *et al.* [121] used character embedding ("char2vec") representation. The character vectors are learned for each Unicode code point that appears at least twice in the training data, including punctuation, emoji, and other symbols. Kocmi *et al.* [138] followed a similar setup for input representation, with the difference then they used moving window of 200 characters of input text. More recently, Zhang *et al.* [322] explored LID for code-mixed sentences, the feature embeddings consist of a few separate feature matrices: character, script and lexicon features. Wan *et al.* [297] explored the usage of word embeddings to cluster words per language (German, French, Italian, English and Romanian). Franco-Salvador *et al.* [88] use embeddings of the text based on subword character n-grams using `fastText`. However, some report [54] that the embeddings did't provide any significant improvement over traditional representations. Our intuition is that the success of embeddings largely depends on the quality and size of the training corpora.

2.2.2 Approaches

A number of recent studies used readily available classifier implementations and rather focus on feature engineering and report the performance in the contexts of their studies. In the following subsections we discuss some of the most common approaches for LID tasks.

Probabilistic methods

One of the simplest methods found in the literature is use of so-called positive and negative decision rules. In the case of positive decision rule, the language is identified if a unique character or character n-gram was found. For the negative decision rule, if a character or character n-gram that was found that does not exist in a language, then that language is not considered in identification [70, 107]. Decision trees (DT) and random forests (RF) (ensemble decision trees) with information gain-based decision are shown to be successful [45, 75]. In some LID setups [89], it is demonstrated that a simple dictionary scoring can discriminate between unrelated language groups,

but more sophisticated methods are necessary when the languages are closer in the linguistic tree.

Franco-Salvador *et al.* [87] combined String Kernels (SK) and word embeddings, which capture different characteristics of texts. They run the experiments on two subproblems of LID, called Language Variety Identification (LVI) and Native Language Identification (NLI) with the aim to generalize the approach over related tasks. One of the datasets they used is also used in our study (Section 2.8.1).

Zampieri *et al.* [317] presented some supervised computational methods for the identification of Spanish language variants. The methods include character n-grams (2–5), word unigrams, word bigrams, PoS and morphological features as well as the additional lexicons. In a different study [316], they discussed about efficiency of the n-gram method compared to the efficiency of the Bag-of-Words (BoW) approach. The experiments have shown that the n-gram based algorithm gives better results that the Bag-of-Words approach.

Tromp *et al.* [284] discussed an n-gram graph-based method for LID of short and noisy text which they call LIGA. The results were relatively good; however, language identification task was conducted on significantly different languages. Vogel *et al.* [294] extended the LIGA method, by introducing some improvements with using word length information, reduction of the weight of repeated information, using median scoring, and using log frequencies. Ljubesic *et al.* [160] presented a variation of the task: distinguishing similar Slavic languages/dialects. Two methods are combined: most frequent words and character n-grams.

LID on the idiomatic microblog language is more challenging than on formal texts of equal length[42]. Carter *et al.* [42] go beyond text of the tweet: mentions, hashtags were used as additional features. They opted for using an n-gram approach to LID. They used, as they call, semi-supervised priors to address the sparsity and imbalance if the data. Vatanen *et al.* [291] discussed character-based language identification with n-gram language models. The approach is shown to be well suited to LID tasks that have dozens of languages, little training data and short test samples.

Gamallo *et al.* [92] (TweetLID 2014 workshop [325]) presented the systems which are based on two different strategies: ranked dictionaries and Naïve Bayes classifiers. The ranked dictionaries method outperformed the Naïve Bayes method on a small

training data, while Naïve Bayes algorithm shows better results with larger training data. The experiments have shown that word unigrams perform better than character n-grams. In the same workshop, Porta *et al.* [222] described a system based on Support Vector Machines (SVMs) and Rational Kernels.

Neural-network-based methods

MacNamara *et al.* [168] experimented on LID using Neural Networks (NNs) and showed that 3-gram language model was superior compared to a shallow recurrent neural network (RNN) and exceeded the accuracy for 4%. The experiments with NNs [54] conducted on one of the datasets used in our study (Section 2.8.1) also show that their simple NN architecture performed worse compared to the linear SVM model. However, they also stated that there is still a potential in using NNs for LID using better architectures, more data, and performing exhaustive parameter tuning. On the other hand, there haave been a couple successful studies on NNs and LID. Geng *et al.* [97] use attention layer with Bidirectional Long Short Term Memory Neural Network (Bi-LSTM) to identify utterances in different languages. Note that the dataset is not in a textual form, but represented as *i-vector* [63]. I-vector is a vector representation of a given speech utterance and it is widely used representation in the area of speech recognition. In general, it is built using a simple factor analysis and it is characterized by being low-dimensional, speaker- and channel-dependent. Cai *et al.* [40] showed that using a convolutional layer in conjunction with Bi-LSTM and attention layers can improve the performance (the experiments are conducted on the same dataset as in the previously mentioned study [97]). Kocmi *et al.* [138] described and experimented with a method based on Bidirectional Recurrent Neural Networks and they found that the system performs well in monolingual and multilingual language identification. They covered 131 languages. They also showed that their system works well on short texts. Jaech *et al.* [121], using Bi-RNN took advantage of hierarchical input text representation. First, they trained "char2vec" representations of words and then used Bi-RNN layer to represent temporal characteristic of the word sequence. One of the datasets they used is also used in our study (Section 2.8.1). However, although they showed comparable results to traditional probabilistic methods, they did not provide the results of significance tests.

2.2.3 Language Analysis

One of the early statistical linguistic analyses is presented in 1949 by Zipf [324]. Zipf formulated an empirical law stating that given a large sample of words used, the frequency of any word is inversely proportional to its rank in the frequency table.

Human languages exhibit a wide spectrum of similarities and differences in structure. Ferrer *et al.* [82] analysed syntactic dependency networks for three languages: Czech, German and Romanian. Similarly, Liu *et al.* [159] built fifteen linguistic complex networks based on the dependency syntactic treebanks of fifteen different languages. Gao *et al.* [95] generated independent word co-occurrence network for Arabic, Chinese, English, French, Russian and Spanish languages. They found several interesting results. First, English language network is more dense which means, they concluded, the English language is more flexible and powerful in expression. Second, Spanish and French is more constrained by the rules. Arabic and Russian have many inflections, and the networks are very sparse. Chinese shows that less characters express more meaning than other languages.

Asgari *et al.* [14] built word co-occurrence network for fifty languages. The edges between nodes are weighted use cosine similarity between word embeddings. Additionally, they perform word alignment between two graphs, which means that the words from different language networks are aligned by semantic similarity. Their results show that they were able to show that clustering follows the expert knowledge on linguistic similarity. This approach is powerful, but there are a few drawbacks. First, large amount of data is required to train word vectors for each language. Second, training high quality word vectors can take long time. Last, vector alignment is also time consuming algorithm and human intervention is required to set the seed word mappings among languages. To that end, Gamallo *et al.* [93] that we mentioned earlier provided less computationally expensive approach.

2.2.4 "Off-the-Shelf" LID tools

"Off-the-shelf" LID tools have been evaluated by many researchers. One of the most recent ones [210] tested 13 languages on multilingual comments and identifiers in the documentation of software projects. Tools that were available at the time of writing are listed in Table 2.1.

Table 2.1: List of available "off-the-shelf" tools for language identification.

LID tool	Lang. #	Description
`CelLang`[7]	100	Hybrid ad-hoc ranking combined with dictionary and Naïve Bayesian classifier.
`CLD`[8]	83	Naïve Bayesian classifier.
`fastText LID`[9]	176	Word embeddings and sub-word information by Joulin *et al.* [131].
`GuessLanguage`[10]	64	Dictionary-based.
`LangDetect`[11]	53	Naïve Bayesian classifier.
`LangId`[12]	97	Naïve Bayesian classifier by Lui *et al.* [166].
`LDig`[13]	17	Logistic regression classifier by Okanohara *et al.* [205].
`LingPipe`[14]	trainable	n-gram models.
`OpenNLP`[15]	103	Maximum entropy, perceptron or Naïve Bayesian classifiers.
`TikaIdentifier`[16]	18	Euclidean distance between n-gram profiles.
`TextCat`[17]	69	n-gram ranking-based model by Cavnar *et al.* [43].
`whatlang-rs`[18]	84	n-gram ranking-based model by Cavnar *et al.* [43].

2.3 Standardized Evaluation Metric

LID task us commonly considered as a document-level classification problem. Given a set of labeled evaluation documents ("gold-standard"), and labels predicted by a model, the document-level accuracy (Eq. (2.3)) is the ratio of the correctly labeled documents over the entire evaluation collection. Often authors provide fine-grained

[7] `https://code.google.com/archive/p/language-identification/`
[8] `https://github.com/CLD2Owners/cld2`
[9] `https://fasttext.cc/docs/en/language-identification.html`
[10] `https://bitbucket.org/spirit/guess_language`
[11] `https://code.google.com/p/language-detection/`
[12] `https://github.com/saffsd/langid.py`
[13] `https://github.com/shuyo/ldig`
[14] `http://alias-i.com/lingpipe/index.html`
[15] `http://opennlp.apache.org/models.html`
[16] `https://tika.apache.org/`
[17] `http://odur.let.rug.nl/vannoord/TextCat/`
[18] `https://github.com/greyblake/whatlang-rs`

per-language results. Precision (Eq. (2.4)) and recall (Eq. (2.5)) are the measures that are usually used to express this.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.3}$$

where TP is the number of true positive examples; FN is the number of false negative examples; FP is the number of false positive examples; and TN is the number of true negative examples.

$$P = \frac{TP}{TP + FP} \tag{2.4}$$

$$R = \frac{TP}{TP + FN} \tag{2.5}$$

F-score (Eq. (2.6)) is also a common way to evaluate the performance of a system and it is expressed as the harmonic mean of precision and recall. The F-score was developed in IR to measure the effectiveness of retrieval with respect to a user who attaches different relative importance to precision and recall.

$$F_\beta = (1 + \beta^2) \times \frac{P \times R}{\beta^2 \times P + R} \tag{2.6}$$

β is the parameter which regulates the importance of precision and recall. For example, $\beta = 2$ will weigh more recall, and $\beta = 0.5$ will weigh more precision. In LID, commonly used value is $\beta = 1$ (Eq. (2.7)).

$$F_1 = 2 \times \frac{P \times R}{P + R} \tag{2.7}$$

For multiclass classification problems, two strategies are used to compute global F_1, P and R measures and those are macro-averaging and micro-averaging. We used macro-average for multiclass problems (common choice in evaluation labs).

Statistical Significance New approaches and variations of similar ideas to existing problems are a common occurrence in machine learning. Hence it is important to be able to determine which of them work better in practice. Accuracy measure alone is not enough to report and compare the performance of systems. To ensure the validity of such claims, a couple of papers addressed this problem [64, 69] by laying out the common statistical methodology applicable in all scientific areas and relying on empirical observations called Null Hypothesis Significance Testing (NHST). We

use exact binomial two-tailed McNemar's test for comparisons of error rates on gold standard tests among classifiers, as described by Dietterich [69]. For significance testing on F_1 scores across 10 folds on identical splits across different classifiers, we use Wilcoxon signed-rank test. It is a non-parametric statistical hypothesis test used to compare, among other things, repeated measurements on a single sample to assess whether their population mean ranks differ. It is considered as a weaker test compared to student's t-test. Because of student's t-test strong assumptions, it is not always applicable, and the Wilcoxon test is recommended [69].

Benavoli *et al.* [21] questioned the adequacy of NHST and laid out its main drawbacks. They stated that the frequentist reasoning has great disadvantages and is often improperly used in publications. Instead, they proposed using the Bayesian paradigm to analyse the results. However, this is out of scope of this study and will be considered as a part of future work.

2.4 The Common N-Grams Language Distance Measure

The Common N-Grams (CNG) text classification is an algorithm which in its core compares the frequencies of character n-grams (strings of characters of length n) that are the most common in the considered documents and classes of documents [135]. In other words, the algorithm solves the task of labeling a document (a text message) with a single label from a given fixed set of labels (assigning the text to one class from a fixed set of classes). The document is classified based on the closest similarity measure between the document being tested and the training profiles of different classes. This classification method could be interpreted as a k nearest neighbours classification method with $k = 1$ and where instead of the standard Euclidean distance, a modified Euclidean distance is used, known as the CNG distance (2.8). The CNG distance is used in various text classification tasks. In the original dissimilarity distance proposal, Keselj *et al.* [135] applied it to the authorship attribution task: given a predefined set of author names the algorithm is designed to label an unseen document with one of these names. Using the same method, Jankowska *et al.* [124, 125] conducted a thorough analysis on authorship attribution task in different languages and developed a visualization tool for feature analysis. The model is built from training data which consists of documents with designated class membership. All training documents

belonging to the same class are merged into one document with one class membership. Hence, for each defined class and a new unlabeled document, the algorithm builds a class profile, which consists of frequencies of the most common character n-grams with the length of n. The n-gram frequencies are normalized: n-gram counts divided by the total number of n-grams; i.e., they are estimates of n-gram probabilities based on the training documents.

The k nearest neighbours classifier with CNG ($k = 1$), where only one neighbour (profile) votes on class membership of a new sample, as stated earlier, is based on the CNG dissimilarity distance between n-gram profiles from the profile p_1 of an unlabeled document, to the profile p_2 — the summary profile of all training documents in a certain class — defined with the Eq. (2.8). The total dissimilarity $D(p_1, p_2)$ (as called by authors, since a higher number means lower similarity; i.e., higher dissimilarity or higher distance) between two profiles is calculated by summing squares of relative distances of individual n-gram frequencies over all n-grams:

$$D(p_1, p_2) = \sum_{x \in (p_1 \cup p_2)} \left(\frac{2 \cdot (f_{p_1}(x) - f_{p_2}(x))}{f_{p_1}(x) + f_{p_2}(x)} \right)^2 \tag{2.8}$$

where x are all n-grams in the profiles p_1 and p_2, and $f_{p_i}(x)$ ($i \in \{1, 2\}$) is frequency of n-gram x in profile p_i. If an n-gram x is not present in a profile p_i, we take $f_{p_i}(x) = 0$. The classifier has been also reported as a successful method for various other classification tasks: effective identification of software source code authors [90], genome classification [280], page genre classification [176], determination of composers of musical works [304], and recognition of computer viruses [2]. There are two hyper-parameters that can be set: the n-gram length, and the profile length. The profile length is a cutoff value for selecting the number of most frequent n-grams across all profiles.

2.5 The Effect of Different Feature Weighting Techniques

Term weighting techniques have a wide range of applications in IR and ML domains. It has been thoroughly explored in ongoing evaluation forums including TREC [106] since 1992, NTCIR [134] since 1999, INEX [101] since 2002, and FIRE [171] since 2008. Term weighting aims to evaluate the relative importance of different features

Table 2.2: Local term weighting.

#	Local w.	Expression	Description
1	tf	tf	Raw term frequency.
2	tf_b	$\begin{cases} 1, & \text{if } tf > 0 \\ 0, & \text{otherwise} \end{cases}$	Binary term presence.
3	tf_a	$k + (1-k)\frac{tf}{max_t(tf)}$	Augmented term frequency, $max_t(tf)$ is the maximum frequency of any term in the document, k is a parameter (0.5 for short documents) [242].
4	tf_s	$log(1 + tf)$	Sublinear term frequency.
5	tf_{bm25}	$\frac{(k_1+1)tf}{tf+k_1(1-b+b\frac{dl}{dl_{avg}})}$	BM25 (Best Match) tf, dl_{avg} is the average document length in the corpus. Default k_1 is 1.2 and b 0.95 [129].

within a dataset. There are three components in a term weighting scheme: local weight, global weight and normalization factor (Eq. (2.9)) [149, 242].

$$x_{ij} = l_{ij} \times g_i \times n_j \tag{2.9}$$

where x_{ij} represents the final weight of i-th term in the j-th document, l_{ij} represents the local weight of i-th term in the j-th document, g_i is the global weight of the ith term, and n_j is the normalization factor for the j-th document.

Local term weights are built upon frequencies within a document. The most common used in practice are laid out in Table 2.2 is derived only from frequencies within the document.

The most used representation, tf, counts how many times the term occurs in a document which means the weight is higher for the terms that appear more frequently. The representation tf_b does not take into account the frequency of the term, but instead it records presence or absence of the term. This is useful when the frequency is not important. tf_a is known as augmented term frequency proposed by authors [242]. It gives weight to all terms that appear and then give some additional weight to terms that appear frequently. Sublinear term frequency tf_s is designed to adjust within document frequency. That means that it will assign similar weights to less frequent terms. tf_{bm25} uses the information about document length and it has adjustable parameter k_1. It is important to note that we did not present the experiments with

Table 2.3: Global term weighting. a - number of training documents in the positive category containing term t_i; b - number of training documents in the positive category which do not contain term t_i; c - number of training documents in the negative category containing term t_i; d - number of training documents in the negative category which do not contain term t_i.; N - total number of documents in the training document collection, $N = a + b + c + d$; N^+ - number of training documents in the positive category, $N = a+b$; N^- - number of training documents in the negative category, $N = c + d$.; p^+ - probability of document belonging to positive category, $p^+ = \frac{a/N^+}{a/N^+ + c/N^-}$.; p^- - probability of document belonging to negative category, $p^- = \frac{c/N^-}{a/N^+ + c/N^-}$.

#	Global w.	Expression	Description
1	idf	$log_2(\frac{N}{a+c})$	Inverse term frequency [261].
2	idf_p	$log_2(\frac{N}{a+c} - 1)$	Probabilistic inverse term frequency [306].
3	$bm25$	$log_2\left(\frac{b+d+0.5}{a+c+0.5}\right)$	BM25 *idf* [129].
4	ig	$\frac{a}{N} log_2\left(\frac{aN}{N^+(a+c)}\right) + \frac{b}{N} log_2\left(\frac{bN}{N^+(b+d)}\right) + \frac{c}{N} log_2\left(\frac{cN}{N^-(a+c)}\right) + \frac{d}{N} log_2\left(\frac{dN}{N^-(b+d)}\right)$	Information gain.
5	gr	$\frac{\texttt{idf}_{ig}}{-\frac{N^+}{N} log_2\left(\frac{N^+}{N}\right) - \frac{N^-}{N} log_2\left(\frac{N^-}{N}\right)}$	Gain ratio.
6	mi	$log_2\left(max(\frac{aN}{N^+(a+c)}, \frac{cN}{N^-(a+c)})\right)$	Mutual information.
7	mi'	$log_2\left(max(2p^+, 2p^-)\right)$	Modified mutual information [305].
8	χ^2	$\frac{N(ad-bc)^2}{N^+N^-(a+c)(b+d)}$	χ^2-based weighting.
9	$delta$	$log_2(\frac{N^-a}{N^+c})$	Delta *idf* [175].
10	Δ_{sm}	$log_2(\frac{N^-(a+0.5)}{N^+(c+0.5)})$	Smoothed delta *idf* [207].
11	Δ_{sm2}	$log_2(\frac{(N^- - c)(a+0.5)}{(N^+ - a)(c+0.5)})$	Smoothed delta *idf* [207].
12	Δ_{bm25}	$log_2(\frac{(N^- - c + 0.5)(a+0.5)}{(N^+ - a + 0.5)(c+0.5)})$	BM25 delta *idf* [207].
13	rf	$log_2(2 + \frac{a}{max(1,c)})$	Relevance frequency [149].
14	ne	$1 + p^+ log_2 p^+ + p^- log_2 p^-$	Natural entropy [305].
15	re	$b_0 + (1 - b_0) idf_{ne}$	Regularized entropy [305].

different variants of term weighting for the reason that all of our datasets are short texts and we found no significant impact on the performance.

Term Frequency-Inverse Term Frequency (tf-idf) is one of the best-known weighting algorithms. Several newer methods adapt *tf-idf* for use as part of their process, and many others rely on the same fundamental concept. *idf*, being the measure's key part, was introduced in 1972 by Karen Spärck Jones [261]. Around the same time, Robertson *et al.* [232] examined statistical techniques for exploiting relevance information to weight search terms in documents and developed a weighting algorithm called *Best Match 25 (BM25)*. Rousseau *et al.* [236] conducted experiments with *IDF* and *BM25* variants on IR tasks, and found that the weighting techniques significantly outperform non-weighted setup, but there is no significant difference among the weighting approaches. Another study [285] tested different variants of BM25 on IR tasks and found that adaptive BM25 yields the best performance. For our study, we consider only original BM25 and *BM25 delta* which was proposed by Paltoglou [207] and tested on text classification tasks.

2.6 Common N-Gram-based Feature Weighting Scheme

We introduce two new supervised term weighting schemes for text classification based on Euclidean and CNG distance measures. The motivation behind this is to calculate distance of positive class term profile and negative class term profile. Following the same notation as in Table 2.3, the normalized positive class term frequency can be expressed as: $f^+ = a/N^+$ and the normalized negative class term frequency can be expressed as: $f^- = c/N^-$ (a - number of training documents in the positive category containing term t_i, c - number of training documents in the negative category containing term t_i, N^+ - number of all training documents in the positive category, N^- - number of all training documents in the negative category, K - classes). Hence, the distance measure between positive and negative class term profiles can be expressed as in Eq. (2.10) using the Euclidean metric and as in Eq. (2.11) using CNG.

$$euclidean = \sum_{K} \left(f^+ - f^-\right)^2 \tag{2.10}$$

$$cng = \sum_{K} \left(2 \cdot \frac{f^+ - f^-}{f^+ + f^-}\right)^2 \tag{2.11}$$

We evaluate these two weighting schemes along the ones in Table 2.3.

2.7 Evaluation on 44 (40) European Languages

2.7.1 Dataset

Phylogenetic studies within historical linguistics [215] are interested in finding a good measure for language similarity. A study [93] that was mentioned in the introductory part focused on testing a distance measure (perplexity) between languages. They collected two datasets, each consisting of long articles in 44 European languages. The first dataset is corpus collected from Web pages in the respective languages. The texts are heterogeneous. The second corpus consists of parallel translations of the Bible. The authors provided the dataset and the results of their study, however, the dataset is missing complete training samples for some of the languages that were reported as part of the results. Authors did provide the actual similarity matrix, so we were able to remove the missing languages from it and compare our results. Information about the number of tokens per language, per corpus and missing languages can be found in Tables A.1 and A.2 in Appendix A.

2.7.2 Methodology

We used the CNG algorithm described in Section 2.4. Using the training subsets we prepared language profiles using character n-grams within the range 3–7 and word unigrams. We evaluated the test subsets by preparing test profiles and applied pairwise CNG between each train-test language pair. The gold standard was provided by the original author (it was created by a language expert), and it is in the following format:

- `Portuguese Galician 1`
- `Portuguese Spanish 2`
- `Catalan Spanish 3`
- `Bosnian Serbian 3`
- `...`

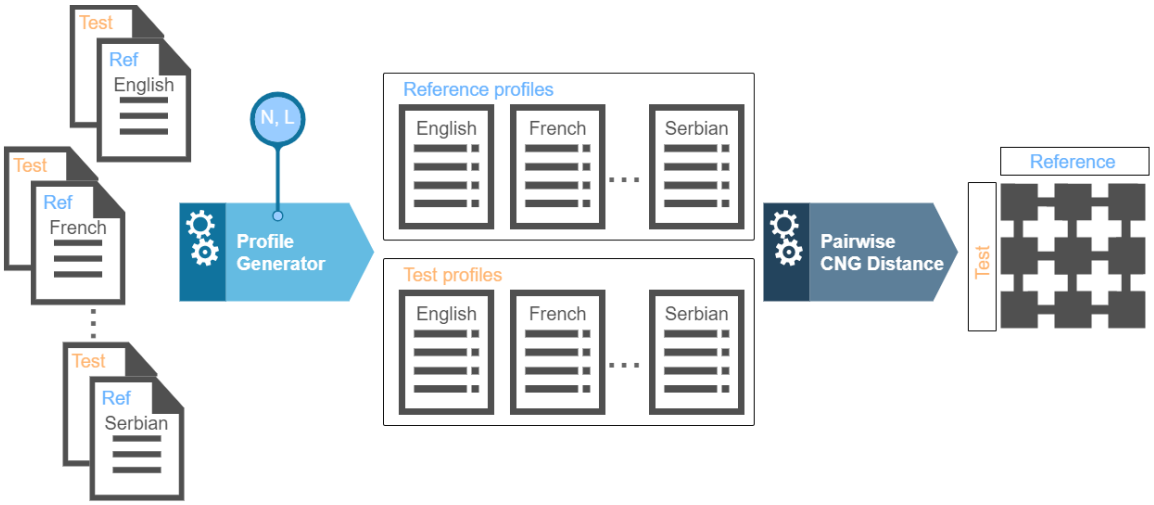

Figure 2.3: Experimental setup for language distance task.

The number represents how far apart are the languages. For example, Portuguese and Galician should be the closest (first neighbours). Portuguese and Spanish should be first or second neighbours. Catalan and Spanish should be first, second or third neighbours, and so on. The gold test consists of 40 language pair distances. For the Web corpus, we had to remove 2 labels, and for the Bible corpus, 4. The system architecture is shown in Fig. 2.3.

For the profile length hyperparameter, for all experiments we used the the maximum length of the smallest profile in the dataset. We evaluated the test subsets by preparing test profiles and applied pairwise CNG between each train-test language pair. The result of this is an asymmetrical distance matrix among train and test language profiles. Then, using the evaluation method described by Gamallo *et al.* [93] we obtain accuracies in reference to the gold standard. To evaluate the significance of the results, we employ McNemar's test. To compare the stability among different methods and setups, we use the Spearman correlation coefficient. The Spearman correlation coefficient is a special case of the Pearson correlation coefficient where the variables are the rank variables.

For a sample of size n, the n raw scores X_i, Y_i are converted to ranks rgX_i, rgY_i and r_s is computed as:

$$r_s = \rho_{rg_X, rg_Y} = \frac{cov(rg_X, rg_Y)}{\sigma_{rg_X} \sigma_{rg_Y}} \tag{2.12}$$

Table 2.4: 40 European languages. Adjusted accuracies are calculated with removed languages that were missing in the test set. $^{\downarrow}$significantly higher ratio of errors compared to adjusted perplexity (highlighted row). $^{\uparrow}$significantly lower ratio of errors compared to adjusted perplexity (highlighted row). McNemar's test with $p < 0.05$.

	Corpus	
Method	**Web**	**Bible**
Rank reported	82.50	82.50
Perplexity reported	85.00	85.00
Rank adjusted	84.21	83.33
Perplexity adjusted	**86.84**	83.33
CNG 3-gram	78.95$^{\downarrow}$	91.67$^{\uparrow}$
CNG 4-gram	78.95$^{\downarrow}$	88.89$^{\uparrow}$
CNG 5-gram	81.58$^{\downarrow}$	88.89$^{\uparrow}$
CNG 6-gram	81.58$^{\downarrow}$	**94.44**$^{\uparrow}$
CNG 7-gram	81.58$^{\downarrow}$	88.89$^{\uparrow}$
CNG w 1-gram	86.11	86.11$^{\uparrow}$

where ρ denotes the Pearson correlation coefficient applied to the rank variables, $cov(rg_X, rg_Y)$ is the covariance of the rank variables, σ_{rg_X} and σ_{rg_Y} are the standard deviations of the rank variables. In our case, raw values are the CNG distances between each language. That means that the Spearman coefficient can show us how much the ranks change across different approaches (experimental hyperparameter settings).

2.7.3 Results and Discussion

Table 2.4 shows the results of the CNG algorithm. Significantly better results were obtained on the Bible corpus with every character n-gram profile length in the observed range (3–7). CNG with 6-grams performed the best with an accuracy of 94.44%, which is better than 83.33% and even reported 85.00%. However, CNG performed significantly worse on the Web corpus, where the best accuracy is 81.58% compared to 86.84%. Word unigrams provided the same accuracy for both datasets, where for the Web dataset it was not significantly different from the authors', and for the Bible dataset was better than the reported, but worse than our character n-gram setup. The CNG method seemed to favour the parallel corpus with only 2 errors out of 36 available labels. The misses were Catalan-French and Russian-Ukrainian. The Web corpus was harder and resulted in somewhat lower accuracies.

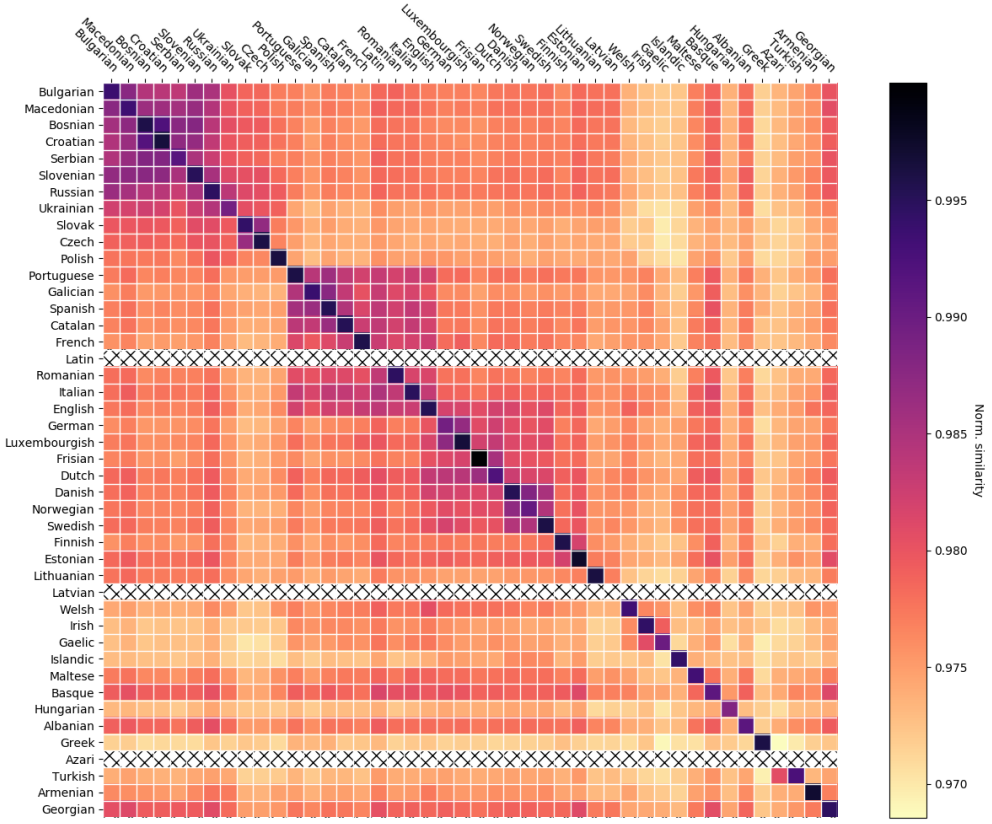

Figure 2.4: Language similarity using CNG and 3-gram features. Web corpus.

Figures 2.4 and 2.5 show the similarity matrix in the form of a heatmap. We can clearly see the big language groups such as Slavic in top left corner, followed by Romantic, then by Germanic. The bottom right corner represents the languages that are unique by its origins, such as Albanian, Basque and Hungarian. One interesting observation is that the Latin language shows higher similarity across all groups, indicating its historic influence.

On Fig. 2.6 we present the results of averaged Spearman correlation among different approaches. Spearman correlation measures statistical dependence between the rankings of two variables. The correlation is stronger if the value is closer to 1, and it is considered weak if it is closer to 0. In our case, we want to measure how constant each language neighbour rankings are. We consider the method more stable if it shows high Spearman correlation among different setups. The Rank approach measures, in general, exhibit weaker stability. CNG-based runs show that the ranks are more stable, but that they drop with the length of n-grams.

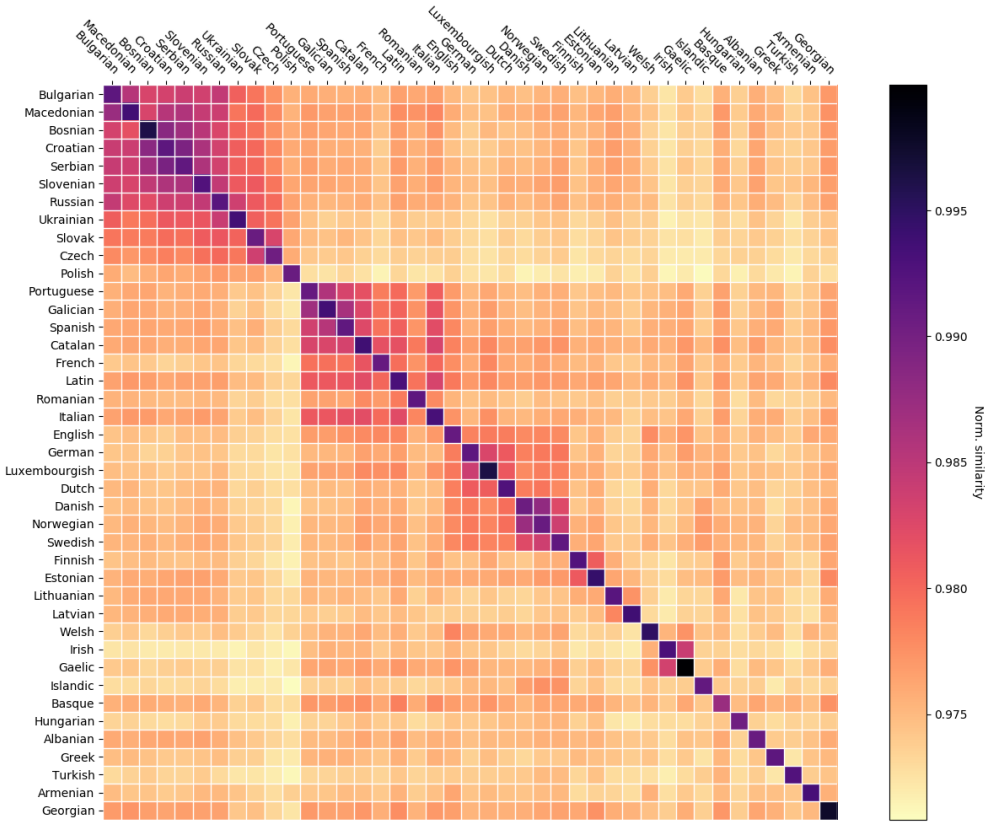

Figure 2.5: Language similarity using CNG and 3-gram features. Bible corpus.

The final language graph generated by CNG method on Bible corpus with 7-grams is shown in Fig. 2.7. The nodes represent languages, the edges represent similarities, and the layout used is force layout. The graph is generates using `networkx` and `matplotlib` Python libraries.

2.8 Evaluation on 7 LID datasets

We considered 7 different datasets from recent evaluation campaigns. The specific details on each dataset are given in the following sections.

2.8.1 Datasets

TweetLID (SEPLN 2014)

The dataset presented in the SEPLN 2014 TweetLID Workshop [325] is an annotated corpus of nearly 35k tweets. The tweets are written in the top five languages of

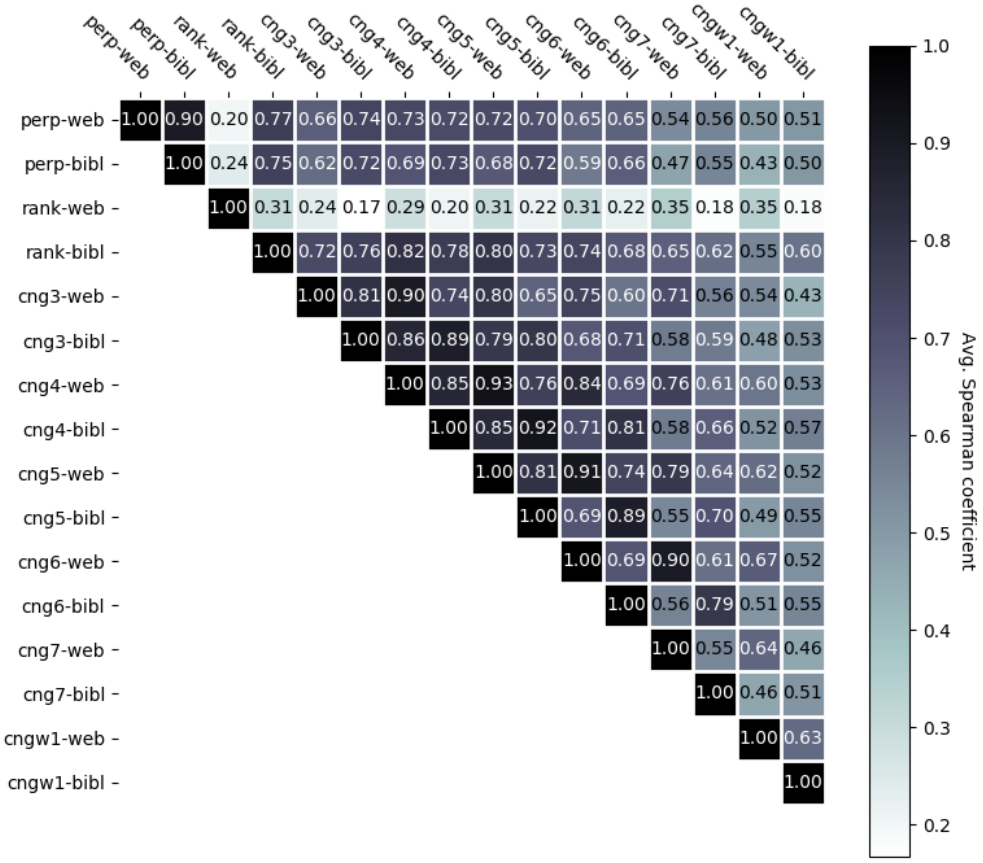

Figure 2.6: Average Spearman coefficient among measures.

Iberian Peninsula which include Castilian, Catalan, Galician, Portuguese, Basque and English (Table 2.5). The dataset also includes some noise tweets, either written in some other language or have more than one label (code-switched text, multilingual). Additionally, some tweets are labeled as "ambiguous", which means that it could not be distinguished as a unique language by a human annotator.

Twitter data consisting of users' comments generally shows several challenging characteristics which we considered in our experiments. First, messages are noisy and likely incorrect in terms of grammar, skewed by informatively irrelevant elements, such as emoticons, hyperlinks, numbers, onomatopoeias and hashtags. In addition to that, messages are rather short due to Twitter message limitation of 140 characters, and extracted features do not provide as much information as longer texts. Another characteristic particular to the dataset is that the similarity between certain classes is very high. A characteristic that is very often found in various classification problems

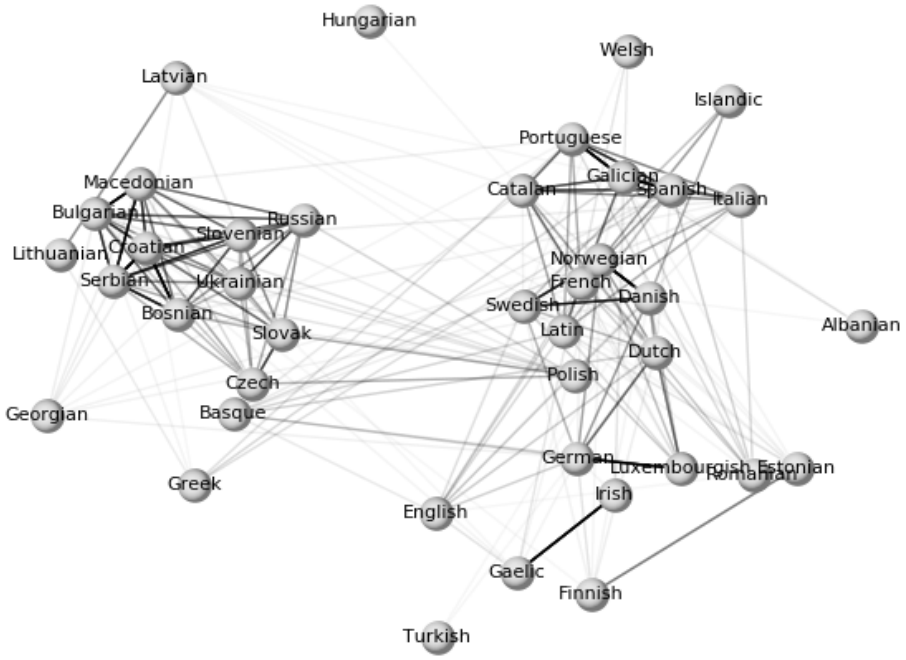

Figure 2.7: Graph generated from CNG similarity matrix.

Table 2.5: TweetLID v2.0 dataset overview of the training, development subset.

		Train & Dev.		**Test**	
Language/Variety	**Class**	**Instances**	**Tokens**	**Instances**	**Tokens**
Castilian	`es`	8,562	84,036	12,812	132,382
Catalan	`ca`	1,466	18,383	1,471	18,019
Galician	`gl`	507	4,801	456	4,369
Portuguese	`pt`	2,151	19,991	2,169	19,800
Basque	`eu`	380	2,307	374	2,152
English	`en`	999	9,242	968	9,012
Other	`other`	21	202	420	3,851
Undefined	`und`	188	477	595	2,100
Ambiguous	`amb`	717	5,204	651	4,856
Total		14,991	144,643	19,916	196,541

is the existence of a class representing a set of texts which do not belong to any of the defined classes ("open-world" classification problem). Denoted as "other" (or undefined "und"), this class is the default class, which means if a message does not belong to any of the defined languages, it should be classified as "other". For example, the number of messages in the training set labeled as Spanish ("es") is 61.22% and the messages labeled as undefined is 3.51%.

Table 2.6: DSLCC v3.0 dataset overview of the training and development subset. Information taken from the authors [172].

		Train & Dev.		Test	
Language/Variety	**Class**	**Instances**	**Tokens**	**Instances**	**Tokens**
Bosnian	`bs`	20,000	743,732	1,000	37,630
Croatian	`hr`	20,000	874,555	1,000	42,703
Serbian	`sr`	20,000	813,076	1,000	41,153
Indonesian	`id`	20,000	831,647	1,000	42,192
Malay	`my`	20,000	618,532	1,000	31,162
Brazilian Portuguese	`pt-BR`	20,000	988,004	1,000	49,288
European Portuguese	`pt-PT`	20,000	908,605	1,000	45,173
Argentine Spanish	`es-AR`	20,000	999,425	1,000	50,135
Castilian Spanish	`es-ES`	20,000	1,080,523	1,000	53,731
Mexican Spanish	`es-PE`	20,000	751,718	1,000	47,176
Canadian French	`fr-CA`	20,000	772,467	1,000	38,602
Hexagonal French	`fr-FR`	20,000	963,867	1,000	48,129
Total		240,000	10,346,151	12,000	527,074

DSLCC v3.0 (VarDial 2016)

The dataset released for VarDial Workshop [172] at COLING 2016. Table 2.6 shows the distribution of test, development and training samples, as well as the number of tokens (words) per subset. The dataset consists of three test sets: one in-domain (Table 2.6), and two out-of-domain (Table 2.7). The in-domain test set contains 1,000 instances per language of journalistic data.

The out-of-domain test sets B1 and B2 contain 100 Twitter users per language or variant each, and a varying number of tweets per user. The number of classes for these subsets is reduced and they cover only two groups of closely-related languages: South-Slavic (Bosnian, Croatian, Serbian) and Portuguese (Brazilian and European). The token distribution is shown in Table 2.7.

Fig. 2.8 shows the sample length distribution of test and development sets. All the distributions are relatively similar (majority 100–400) except Malay, Indonesian and Mexican Spanish where samples tend to be in a shorter range (100–300).

DSLCC v4.0 (VarDial 2017)

In 2017, organizers [318] continued the evaluation campaign on DSLCC datasets. This dataset does not have an out-of-domain evaluation test set, but has three additional

Table 2.7: DSLCC v3.0 dataset overview of the test subset. Information taken from the authors [172].

		Out-of-domain test			
Language/Variety	**Class**	**B1**	**Tokens**	**B2**	**Tokens**
Bosnian	bs	100	209,884	100	170,481
Croatian	hr	100	179,354	100	119,837
Serbian	sr	100	181,185	100	124,469
Brazilian Portuguese	pt-BR	100	151,749	100	19,567
European Portuguese	pt-PT	100	134,139	100	13,145
Total		500	856,331	500	323,030

Table 2.8: DSLCC v4.0 dataset overview of the test, development and test subsets. Information taken from the authors [318]

		Train & Dev.		Test	
Language/Variety	**Class**	**Instances**	**Tokens**	**Instances**	**Tokens**
Bosnian	bs	20,000	716,537	1,000	35,756
Croatian	hr	20,000	845,639	1,000	42,774
Serbian	sr	20,000	777,363	1,000	39,003
Indonesian	id	20,000	800,639	1,000	39,954
Malay	my	20,000	591,246	1,000	29,028
Brazilian Portuguese	pt-BR	20,000	907,657	1,000	45,715
European Portuguese	pt-PT	20,000	832,664	1,000	41,689
Argentine Spanish	es-AR	20,000	939,425	1,000	42,392
Castilian Spanish	es-ES	20,000	1,000,235	1,000	50,134
Peruvian Spanish	es-PE	20,000	569,587	1,000	28,097
Canadian French	fr-CA	20,000	712,467	1,000	36,121
Hexagonal French	fr-FR	20,000	871,026	1,000	44,076
Persian	fa-IR	20,000	824,640	1,000	41,900
Dari	fa-AF	20,000	601,025	1,000	30,121
Total		280,000	8,639,459	14,000	546,790

language varieties: Persian and Dari (a variety spoken in Afghanistan, often referred to as Farsi) and Peruvian Spanish. Table 2.8 shows the token distribution across subsets and classes.

Fig. 2.9 shows the sample character length distribution across classes. We can see that most of the classes have the majority of the samples between 50 and 300. For Portuguese variants the length tends to be slightly longer (200–300), and Peruvian Spanish samples are slightly shorter (50–200).

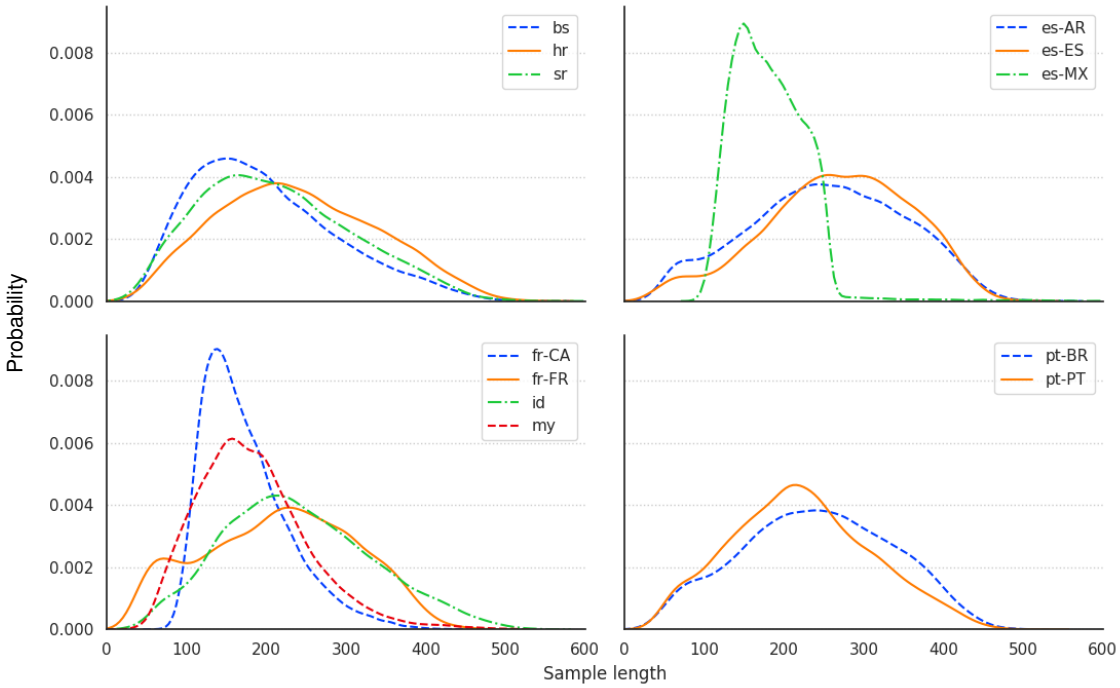

Figure 2.8: Sample length distribution in training & development subsets for DSLCCv3.0.

ILI (VarDial 2018)

VarDial Workshop in 2018 [319] for the first time organized an Indo-Aryan Language Identification task. The target classes are 5 languages and variants from the Indian subcontinent: Awadhi (Awadh region of Uttar Pradesh), Bhojpuri (northern-eastern part of India and the Terai region of Nepal), Braj (northwestern Uttar Pradesh, the eastern extremities of Rajasthan and the southern extremities of Haryana), Hindi (official dialect of India) and Magahi (Bihar, Jharkhand and West Bengal). Table 2.9 shows the number of tokens per language per data subset. Note that Awadhi and Braj are considered as dialects of Hindi, while Bhojpuri and Magahi are different languages, but considered to originate from the same ancient language (Mithila Prakrit or Bengali Prakrit). All languages share the same script - Devanagari.

Fig. 2.10 shows the sample character length distribution across classes. Most of the classes have the majority of the samples between 20 and 150. The texts are sampled mainly from the literature domain, which were published either on the Web or in print.

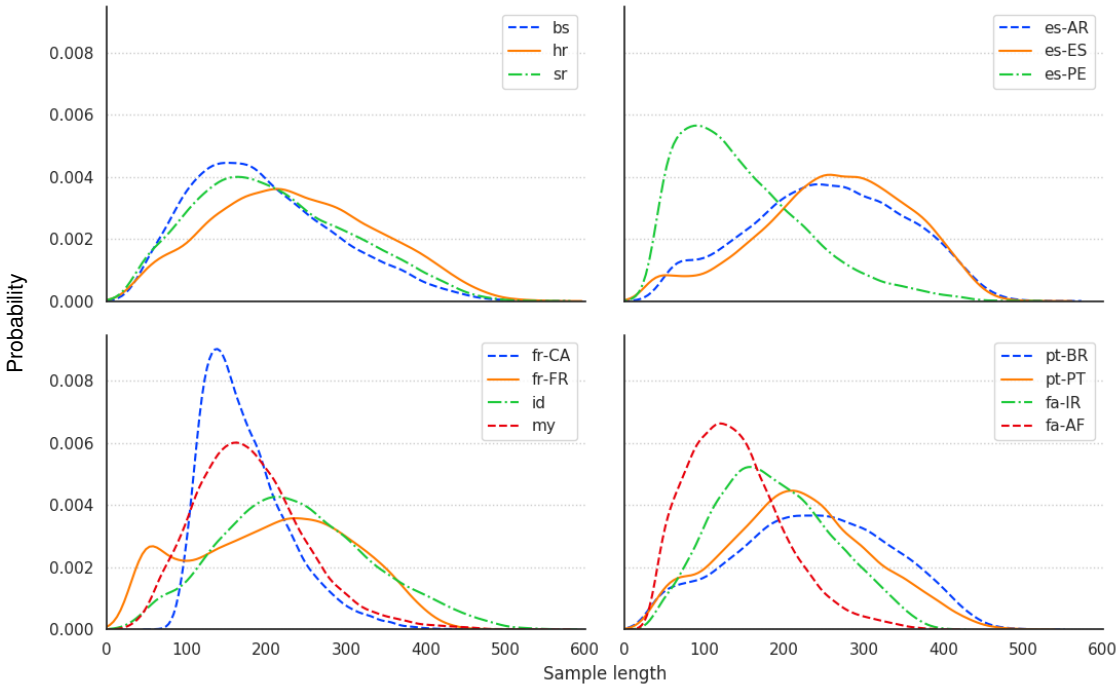

Figure 2.9: Sample length distribution in training & development subsets for DSLCCv4.0.

Table 2.9: ILI dataset overview of the test, development and test subsets.

		Train & Dev.		Test	
Language/Variety	**Class**	**Instances**	**Tokens**	**Instances**	**Tokens**
Awadhi	AWA	10,787	136,893	1,502	22,029
Braj	BRA	17,419	278,438	2,147	30,871
Hindi	HIN	17,895	358,045	1,835	34,888
Bhojpuri	BHO	16,900	304,052	2,006	49,706
Magahi	MAG	17,591	262,337	2,202	33 876
Total		80,592	1,339,765	9,692	171,370

DFS (VarDial 2018)

The same workshop [319] organized the Dutch-Flemish variety identification. The task was defined as a binary classification problem: Dutch (Northern Dutch) and Flemish (a Low Franconian dialect cluster of the Dutch language). The corpus consists of subtitles from articles as samples. These raw subtitles were originally converted into linguistically annotated text in the original SUBTIEL corpus [288]. Table 2.10 shows the token distribution across the subsets and languages.

Fig. 2.11 shows the sample character length distribution across classes. Most of

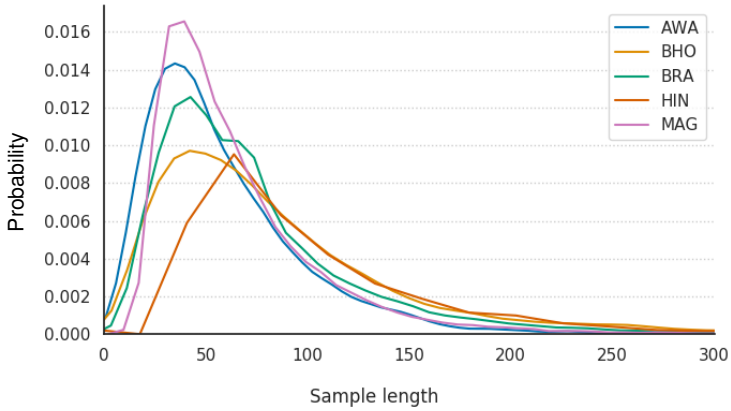

Figure 2.10: Sample length distribution in training & development subsets for ILI'18.

Table 2.10: DFS dataset overview of the test, development and test subsets.

		Train & Dev.		Test	
Language/Variety	**Class**	**Instances**	**Tokens**	**Instances**	**Tokens**
Flemish	BEL	150,250	5,027,941	10,000	334,408
Netherlandic	DUT	150,250	5,104,494	10,000	339,112
Total		300,500	10,132,435	20,000	673,520

the classes have the majority of the samples between 120 and 250.

GDI (VarDial 2018 & 2019)

The German Dialect Identification shared task was first introduced as a part of the VarDial Workshop series in 2017. The dataset is subsampled from the Archimob corpus of spoken Swiss German [243] originally developed for studying linguistic micro-variation and for developing NLP tools. The compilation of this corpus is set in the context of an increasing presence of Swiss German variants in different domains of everyday communication. The subset is also adjusted to the purposes of the task, where the organizers removed information about the authors of the utterances and general metadata related to transcriber, and particular tokens. The task is formulated as a classification problem, where the participants were presented with four language variations to be classified: Bern, Basel, Lucerne and Zurich. Additionally, in the 2018 task, in the test phase, organizers introduced a fifth, unknown language variety, which was not part of the training set. Tables 2.11 and 2.12 show the number of tokens per subset (training and gold), per language variety.

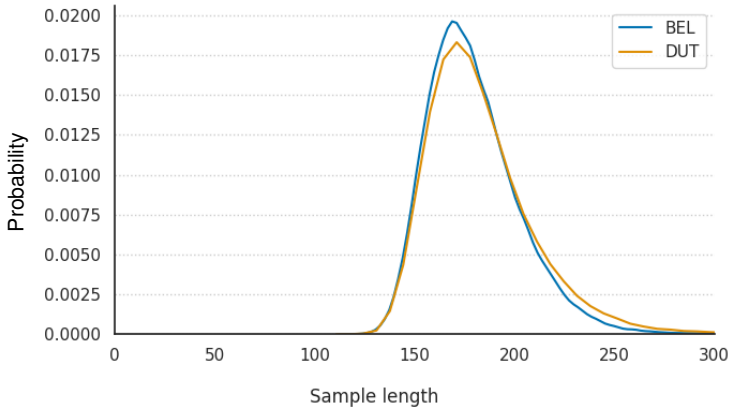

Figure 2.11: Sample length distribution in training & development subsets for DFS'18.

Table 2.11: GDI 2018 dataset overview of the train, development and test subsets.

		Train & Dev.		Test	
Language/Variety	**Class**	**Instances**	**Tokens**	**Instances**	**Tokens**
Bern	BE	4,956	35,962	1,191	12,013
Basel	BS	4,921	36,965	1,200	9,802
Lucerne	LU	4,593	38,328	1,186	11,372
Zurich	ZH	4,834	36,919	1,175	9,610
Unknown	XY	-	-	790	8,938
Total		19,304	148,174	5,542	51,735

Table 2.12: GDI 2019 dataset overview of the train, development and test subsets.

		Train & Dev.		Test	
Language/Variety	**Class**	**Instances**	**Tokens**	**Instances**	**Tokens**
Bern	BE	4,803	35,349	1,191	12,013
Basel	BS	4,797	36,389	1,199	9,803
Lucerne	LU	4,407	37,629	1,176	11,271
Zurich	ZH	4,802	36,919	1,177	9,612
Total		18,809	146,286	4,743	42,699

Fig. 2.12 shows the sample character length distribution across classes. Most of the classes have the majority of the samples between 20 and 80.

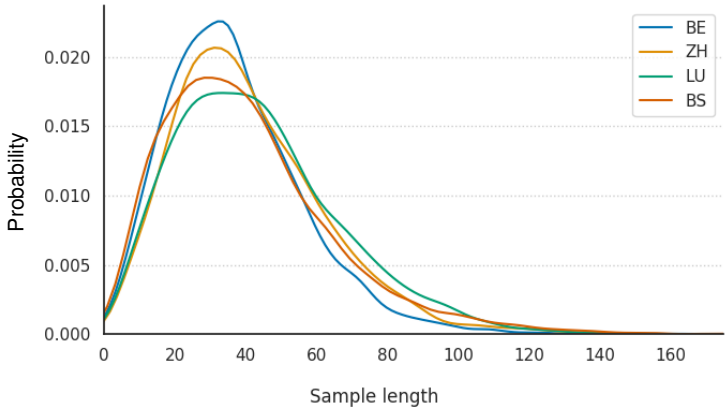

Figure 2.12: Sample length distribution in training & development subsets for GDI'18.

MADAR (ACL 2019)

Organizers presented the MADAR (Multi-Arabic Dialect Applications and Resources)[19] shared task at ACL 2019 [37]. The corpus is constructed from a commissioned translation of the Basic Traveling Expression Corpus (BTEC) [274] sentences from English and French to 26 different Arabic dialects in parallel. The dialects cover 25 different Arabic-speaking cities across North Africa and the Middle East with the addition of Modern Standard Arabic (MSA). Table 2.13 shows the token distribution over the subsets and over the dialects.

Fig. 2.13 shows the sample character length distribution across classes. Most of the classes have the majority of the samples between 10 and 70, and the distribution is even, partially because of the fact that the corpus is parallel.

2.8.2 Methodology

In our experiments, we used three ML algorithms (SVM, Logistic Regression and Multinomial Naïve Bayes) with default parameters in the `scikit-learn` Python library. For SVM we used the `sklearn.svm.LinearSVC` implementation, which is more suitable for bigger datasets because it is an optimization designed for the linear kernel. Default parameters are `C=1.0; loss=squared_hinge; penalty=l2`. For Logistic Regression we used `sklearn.linear_model.LogisticRegression` implementation with the default parameters `C=1.0; solver=lbfgs; penalty=l2`. *LBFGS* is an

[19] `https://camel.abudhabi.nyu.edu/madar/`

Table 2.13: MADAR 2019 dataset overview of the train, development and test subsets.

		Train & Dev.		Test	
Language/Variety	**Class**	**Instances**	**Tokens**	**Instances**	**Tokens**
Rabat (Morocco)	RAB	1,800	13,303	200	1,475
Fes (Morroco)	FES	1,800	13,067	200	1,455
Algiers (Algeria)	ALG	1,800	13,123	200	1,466
Tunis (Tunisia)	TUN	1,800	12,470	200	1,406
Sfax (Tunisia)	SFX	1,800	12,221	200	1,364
Tripoli (Libya)	TRI	1,800	13,003	200	1,463
Benghazi (Libya)	BEN	1,800	13,030	200	1,447
Cairo (Egypt)	CAI	1,800	13,020	200	1,438
Alexandria (Egypt)	ALX	1,800	13,174	200	1,443
Aswan (Egypt)	ASW	1,800	13,389	200	1,484
Khartoum (Egypt)	KHA	1,800	13,269	200	1,451
Jerusalem (S. Levant)	JER	1,800	12,565	200	1,371
Amman (S. Levant)	AMM	1,800	13,246	200	1,452
Salt (S. Levant)	SAL	1,800	12,686	200	1,432
Beirut (N. Levant)	BEI	1,800	12,121	200	1,333
Damascus (N. Levant)	DAM	1,800	12,204	200	1,376
Aleppo (N. Levant)	ALE	1,800	12,185	200	1,340
Mosul (Iraq)	MOS	1,800	12,787	200	1,403
Baghdad (Iraq)	BAG	1,800	12,311	200	1,360
Basra (Iraq)	BAS	1,800	11,901	200	1,298
Doha (Gulf)	DOH	1,800	12,123	200	1,328
Muscat (Gulf)	MUS	1,800	13,050	200	1,456
Riyadh (Gulf)	RIY	1,800	12,640	200	1,405
Jeddah (Gulf)	JED	1,800	12,139	200	1,341
San'a (Yemen)	SAN	1,800	12,763	200	1,431
Modern Standard Arabic	MSA	1,800	14,339	200	1,593
Total		46,800	332,129	5,200	36,811

iterative method for solving nonlinear optimization problems using a limited amount of computer memory. For two datasets (DSLCC), due to the fact that the aforementioned implementation makes a copy of the data in `C++` (underlying implementation `liblinear`[20]) and takes up extra space, we used `sklearn.linear_model.SGDClassifier` with the parameter setting `loss=log`, which is, essentially, logistic regression that instead of minimizing the log-probability uses Stochastic Gradient Descent (SGD) as a solver. For Multinomial Naïve Bayes we used the `sklearn.naive_bayes.MultinomialNB` implementation with default parameters. We did not apply extensive grid search to

[20] `https://www.csie.ntu.edu.tw/~cjlin/liblinear/`

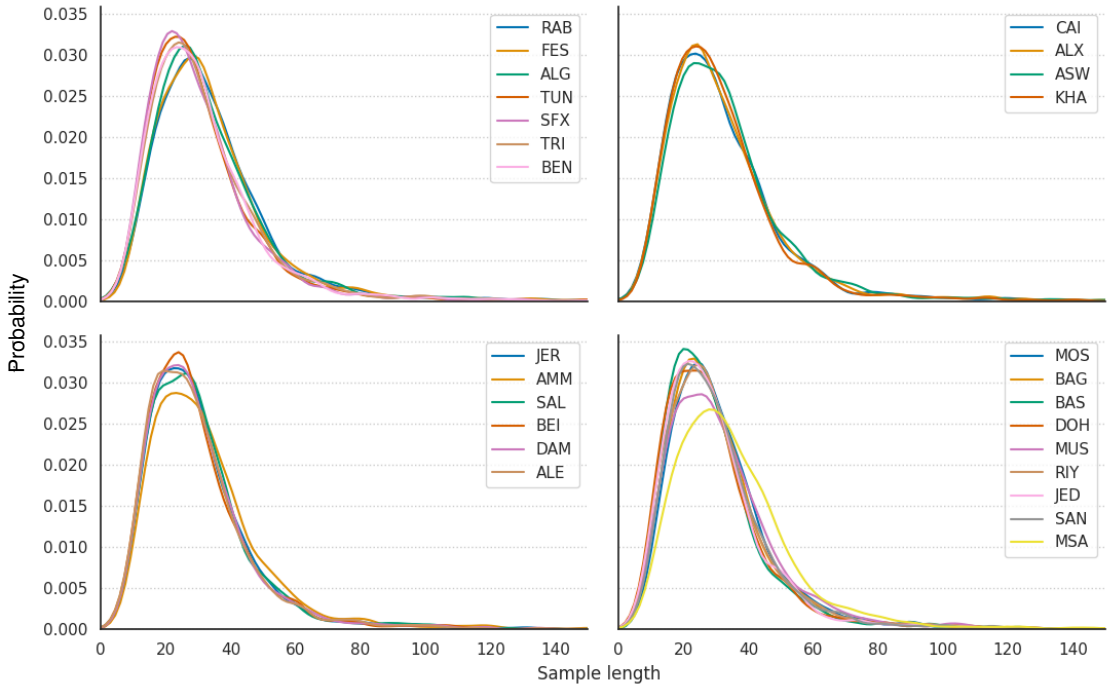

Figure 2.13: Sample length distribution in training & development subsets for MADAR.

Figure 2.14: Experimental setup for language feature weighting task. *N-gram sizes with length setting 1–7, depending on a dataset; **15 weighting schemes; ***3 different classifiers: Multinomial Naïve Bayes, Logistic Regression and Linear SVM.

find optimal hyperparameters because that is out of the scope of this experiment. We particularly focus on the effects of different feature weighting schemes. The experimental setup is shown in Fig. 2.14.

2.8.3 Results and Discussion

In this section we present the results and discuss our findings on the 7 LID datasets described previously.

TweetLID'14 Table 2.14 shows the results on the TweetLID dataset using the F_1 measure designed for multilabel problems and used for system evaluation by the task organizers. This dataset was more difficult to work on due to the large class imbalance, unseen class problem and code-mixed tweets. Features that were used in the experiment include character n-grams ($n \in \{2..4\}$) and word unigrams. We show that, in the case of this dataset, the weighting scheme can have a significant impact on the results. Most of the weighting schemes did improve the final result compared to no-weighting, but not all of them were significant. Linear SVM showed the best (and significant) performance using mutual information weighting scheme (*mi*), followed by modified mutual information (*mi'*) and *cng*. Scheme *idf* did slightly improve the performance, but we did not find statistical significance on the given test set. Table A.3 in Appendix A shows the comparison between our best results and the workshop top three results. It seems that our method did perform better (best workshop result is 75.2%), but we were unable to verify whether our results are statistically significant, due to the lack of detailed results by the systems of interest. Additionally, we were left unclear if the gold test set is identical to the one that was used in the competition, because even with no weighting applied, we obtained slightly better results.

DSLCCv3.0'16 Tables 2.15 and 2.16 show the results on the DSLCCv3.0 dataset on the in-domain gold test set and two out-of-domain gold test sets, respectively, using macro-averaged F_1 measure. We performed 10-fold cross-validation and tested it separately on the gold test set. The challenge of this dataset, as mentioned, is that it had two additional out-of-domain test sets. We briefly described in the previous section that the training set comes from the news sites in respective languages, while B1 and B2 test sets are samples collected from Twitter. The feature set consists of character n-grams ($n \in \{1..7\}$) We can observe that the weighting at first glance didn't provide any gains in performance. However, McNemar's statistic with $p < 0.05$ indicates that *idf*, *bm25* and *cng* have statistically significant ratio of errors compared to the no-weighting setup. The top three results in the workshops on the in-domain test sets are 89.4%, 88.8% and 88.7%.

Although the results on B1 and B2 test sets seem very high, the task is simpler

Table 2.14: Results on the TweetLID dataset (global weighting scheme)×(classifier). F_1 measure was obtained using organizers' script for evaluation adjusted for multilabel problems. Due to the nature of the dataset we did not perform cross validation. $^{\downarrow}$significantly higher ratio of errors on gold test compared to no-weighting scheme (highlighted row). $^{\uparrow}$significantly lower ratio of errors on gold test compared to no-weighting scheme. McNemar's test, $p < 0.05$.

	Classifiers		
Wgt	**MNB**	**LR**	**LSVM**
none	68.47	69.30	76.53
idf	70.08$^{\uparrow}$	68.82	76.58
bm25	70.10$^{\uparrow}$	68.81	76.54
ig	68.47	69.31	76.54
gr	68.47	69.30	76.53
mi	**70.75**$^{\uparrow}$	**71.32**	**77.07**$^{\uparrow}$
mi'	69.68	70.04$^{\uparrow}$	76.80$^{\uparrow}$
χ^2	64.73$^{\downarrow}$	70.66$^{\uparrow}$	75.07$^{\downarrow}$
Δ_{sm}	68.47	69.30	76.53
Δ_{sm2}	68.47	69.30	76.53
Δ_{bm25}	68.47	69.30	76.53
rf	69.00$^{\uparrow}$	69.49	76.58
ne	68.62	69.32	76.47
$re_{b_0=0.2}$	68.60	69.31	76.47
$re_{b_0=0.5}$	68.58	69.34	76.50
$re_{b_0=0.7}$	68.51	69.35	76.54
euclid	68.47	69.30	76.53
cng	68.94$^{\uparrow}$	69.40	76.62$^{\uparrow}$

as the number of test classes is reduced to 5. The best results for both datasets (Table 2.16) were obtained using Linear SVM. Mutual information (*mi*) significantly improved the results over the no-weighting setup. Significant weighting schemes were obtained by *idf* and *cng*. Similar to the previous dataset, χ^2 had a detrimental effect on the performance. The top three results in the workshop on B1 test set are 91.9%, 91.3% and 89.7%. The top three results in the workshop on B2 test set are 87.7%, 85.7% and 83.8%.

DSLCCv4.0'17 Table 2.17 shows the results on the DSLCCv4.0 dataset on the gold test set with 10 fold cross validation on the training set. As features, we used character n-grams (3–6) and word unigrams. For MNB classifier the best result was obtained by applying the *bm25* weighting scheme which was significantly better that

Table 2.15: Results on the DSLCCv3.0 dataset (global weighting scheme)×(classifier). F_1 measure mean and standard deviation across 10 folds. $^{\downarrow}$significantly higher ratio of errors on gold test compared to no-weighting scheme (highlighted row). $^{\uparrow}$significantly lower ratio of errors on gold test compared to no-weighting scheme. McNemar's test, $p < 0.05$. $^{\Downarrow}$significantly lower 10 fold results compared to no-weighting scheme. $^{\Uparrow}$significantly higher 10 fold results compared to no-weighting scheme. Wilcoxon's test, $p < 0.05$.

	Classifiers		
Wgt	**MNB**	**LR**	**LSVM**
none	83.12 (81.71±0.44)	77.51 (78.33±0.47)	88.21 (86.94±0.25)
idf	**85.21**$^{\uparrow}$ **(83.12±0.35)**$^{\Uparrow}$	81.00$^{\uparrow}$ (81.34±0.38)$^{\Uparrow}$	88.50 (86.99±0.15)
bm25	85.21$^{\uparrow}$ (83.13±0.35)$^{\Uparrow}$	**81.05**$^{\uparrow}$ **(81.42±0.40)**$^{\Uparrow}$	**88.51**$^{\uparrow}$ **(86.99±0.14)**
ig	83.11 (81.69±0.43)	77.46 (78.31±0.46)	88.20 (86.94±0.25)
gr	83.12 (81.71±0.44)	77.48 (78.33±0.44)	88.21 (86.94±0.25)
mi	81.59)$^{\downarrow}$ (80.29±0.36)$^{\Downarrow}$	76.42 (77.13±0.51)$^{\Downarrow}$	87.86 (86.35±0.24)
mi'	80.72)$^{\downarrow}$ (80.18±0.25)$^{\Downarrow}$	77.67 (78.66±0.52)	87.67 (86.33±0.33)
χ^2	83.07 (81.66±0.45)	77.38 (78.26±0.50)	88.26 (86.94±0.22)
Δ_{sm}	83.12 (81.71±0.44)	77.45 (78.34±0.50)	88.21 (86.94±0.25)
Δ_{sm2}	83.12 (81.71±0.44)	77.49 (78.36±0.47)	88.21 (86.94±0.25)
Δ_{bm25}	83.12 (81.71±0.44)	77.57 (78.35±0.47)	88.21 (86.94±0.25)
rf	83.53 (82.12±0.39)	78.17 (78.90±0.51)	88.26 (87.09±0.22)
ne	82.95 (81.62±0.42)	77.48 (78.34±0.49)	88.16 (86.90±0.27)
$re_{b_0=0.2}$	82.95 (81.63±0.42)	77.54 (78.34±0.44)	88.16 (86.91±0.27)
$re_{b_0=0.5}$	83.00 (81.65±0.43)	77.52 (78.36±0.48)	88.17 (86.91±0.26)
$re_{b_0=0.7}$	83.04 (81.66±0.43)	77.50 (78.35±0.49)	88.17 (86.92±0.25)
euclid	83.12 (81.71±0.44)	77.55 (78.35±0.44)	88.21 (86.94±0.25)
cng	83.47 (81.89±0.41)	78.19 (78.90±0.40)	88.21 (86.90±0.24)

the no-weighting scenario for the training and gold data. A significant result was obtained with *idf*, while others, such as *cng* and *rf* did show improvement, but were not significantly better. Surprisingly, *mi* and *mi'* had a significant negative effect. For the LR classifier the result was similar, but *rf* and *cng* did have significant positive impact. The best overall performance was obtained by linear SVM with *bm25* and *idf* weighting. Other weighting schemes did not have a significant impact on the performance. For comparison, the three best performing systems at the workshop were evaluated at 92.7%, 92.5% and 91.6%, respectively. All of them implement two level classification — language group classifier, followed by a language specific classifier. Interestingly, the best performing system used the *bm25* weighting scheme with SVMs, which confirms our findings.

Table 2.16: Results on the DSLCCv3.0 dataset (global weighting scheme)×(classifier). F_1 measure on B1 & B2 out-of-domain datasets. $^{\downarrow}$significantly higher ratio of errors compared to no-weighting scheme (highlighted row). $^{\uparrow}$significantly lower ratio of errors compared to no-weighting scheme. McNemar's test, $p < 0.05$.

	B1			B2		
Wgt	**MNB**	**LR**	**LSVM**	**NB**	**LR**	**LSVM**
none	88.77	85.32	89.10	80.70	76.47	85.32
idf	89.05	84.47	88.91	82.03	77.33	85.93
bm25	89.05	83.54	88.91	82.03	77.12	85.93
ig	88.77	85.32	89.10	80.70	77.12	85.72
gr	88.77	85.32	89.10	80.70	76.47	85.72
mi	**89.26**	86.80$^{\uparrow}$	89.10	**82.39**	79.83$^{\uparrow}$	85.70
mi'	88.02	**87.02**$^{\uparrow}$	89.10	82.37	**80.45**$^{\uparrow}$	**86.62**
χ^2	79.10$^{\downarrow}$	77.40$^{\downarrow}$	85.96$^{\downarrow}$	73.94$^{\downarrow}$	69.94$^{\downarrow}$	81.63$^{\downarrow}$
Δ_{sm}	88.77	85.32	89.10	80.70	76.47	85.72
Δ_{sm2}	88.77	84.46	89.10	80.70	76.63	85.72
Δ_{bm25}	88.77	85.32	89.10	80.70	76.47	85.72
rf	88.82	86.51	89.12	81.98	79.33$^{\uparrow}$	85.94
ne	88.79	85.93	89.10	81.11	77.47	85.72
$re_{b_0=0.2}$	88.79	85.32	89.10	81.11	77.86$^{\uparrow}$	85.72
$re_{b_0=0.5}$	88.57	85.53	89.10	81.11	78.01$^{\uparrow}$	85.72
$re_{b_0=0.7}$	88.77	85.74	89.10	80.92	77.55	85.72
euclid	88.77	85.32	89.10	80.70	76.47	85.72
cng	88.37	85.53	**89.32**	81.18	78.21$^{\uparrow}$	85.94

ILI'18 Table 2.18 shows the results on the ILI dataset on the gold test set with 10 fold cross validation on the training set. For MNB classifier, we found that *mi* and *mi'* boosted the performance of the classifier, while *idf* and χ^2 had a negative impact. For the LR classifier the results are slightly different. While *mi* and *mi'* seem best, *idf*, *rf*, *ne*, *re* and *cng* had positive impact as well. Scheme *bm25* had a negative impact. The best performing classifier was linear SVM with *mi* as the best weighting scheme obtained 90.92% on the gold test. Only χ^2 had a negative impact. The three best performing systems at the workshop were evaluated at 95.8%, 90.2% and 89.8%, respectively. However, it is important to point out that the first result used an adaptation technique, where, through multiple runs on the gold test, they reused the best predictions for retraining and retested on the same gold test.

Table 2.17: Results on the DSLCCv4.0 dataset (global weighting scheme)×(classifier). F_1 measure mean and standard deviation across 10 folds. $^{\downarrow}$significantly higher ratio of errors on gold test compared to no-weighting scheme (highlighted row). $^{\uparrow}$significantly lower ratio of errors on gold test compared to no-weighting scheme. McNemar's test, $p < 0.05$. $^{\Downarrow}$significantly lower 10 fold results compared to no-weighting scheme. $^{\Uparrow}$significantly higher 10 fold results compared to no-weighting scheme. Wilcoxon's test, $p < 0.05$.

	Classifiers		
Wgt	**MNB**	**LR**	**LSVM**
none	85.08 (84.84±0.29)	80.36 (81.03±0.42)	90.28 (90.28±0.35)
idf	86.68$^{\uparrow}$ (86.29±0.33)$^{\Uparrow}$	83.12$^{\uparrow}$ (83.73±0.28)$^{\Uparrow}$	90.73$^{\uparrow}$ (90.47±0.32)$^{\Uparrow}$
bm25	**86.72$^{\uparrow}$ (86.34±0.33)$^{\Uparrow}$**	**83.28$^{\uparrow}$ (83.86±0.23)$^{\Uparrow}$**	**90.74$^{\uparrow}$ (90.47±0.33)$^{\Uparrow}$**
ig	85.01 (84.79±0.28)	80.50 (80.99±0.40)	90.28 (90.28±0.35)
gr	85.08 (84.84±0.29)	80.65 (81.05±0.42)	90.28 (90.28±0.35)
mi	83.36$^{\downarrow}$ (83.04±0.36)$^{\Downarrow}$	79.38$^{\downarrow}$ (79.72±0.41)$^{\Downarrow}$	89.85 (89.69±0.33)
mi'	83.96$^{\downarrow}$ (83.81±0.33)$^{\Downarrow}$	80.66 (81.21±0.37)	89.64 (89.68±0.37)
χ^2	85.12 (84.84±0.29)	80.32 (81.07±0.44)	90.24 (90.27±0.36)
Δ_{sm}	85.08 (84.84±0.29)	80.57 (81.08±0.41)	90.28 (90.28±0.35)
Δ_{sm2}	85.08 (84.84±0.29)	80.59 (81.03±0.36)	90.28 (90.28±0.35)
Δ_{bm25}	85.08 (84.84±0.29)	80.58 (81.06±0.48)	90.28 (90.28±0.35)
rf	85.55 (85.41±0.33)	81.05$^{\uparrow}$ (81.53±0.38)$^{\Uparrow}$	90.47 (90.40±0.35)
ne	85.01 (84.79±0.27)	80.63 (81.07±0.31)	90.26 (90.24±0.36)
$re_{b_0=0.2}$	84.99 (84.80±0.27)	80.57 (81.07±0.43)	90.27 (90.25±0.36)
$re_{b_0=0.5}$	85.03 (84.80±0.28)	80.49 (81.01±0.39)	90.26 (90.25±0.36)
$re_{b_0=0.7}$	85.03 (84.81±0.27)	80.46 (81.09±0.35)	90.27 (90.26±0.37)
euclid	85.08 (84.84±0.29)	80.50 (81.06±0.45)	90.28 (90.28±0.35)
cng	85.38 (85.08±0.26)	81.09$^{\uparrow}$ (81.55±0.35)$^{\Uparrow}$	90.28 (90.26±0.36)

DFS'18 Table 2.19 shows the results on the DFS dataset on the gold test set with 10 fold cross validation on the training set. For the MNB classifier, we found that *idf*, *rf*, *ne* and *cng* significantly boosted the performance compared to the no-weighting scheme, with *idf* giving the best boost. None of the weighting schemes negatively affected the performance. Schemes *re* and χ^2 schemes did provide significant boost, but only on the gold test set (McNemar's test), while the folds do not seem significantly different (Wilcoxon's test). For the LR classifier, the highest and only significant impact was *mi* scheme and this is the overall best result — F_1 on the gold test set is 63.95%. The linear SVM classifier does not seem significantly impacted by the weighting schemes. The three best performing systems at the workshop were evaluated at 66.0%, 64.6% and 63.6%, respectively. The first result is produced by a single

Table 2.18: Results on the ILI'18 dataset (global weighting scheme)×(classifier). F_1 measure mean and standard deviation across 10 folds. F_1 measure mean and standard deviation across 10 folds inside brackets, and F_1 measure on gold standard dataset. $^{\downarrow}$significantly higher ratio of errors on gold test compared to no-weighting scheme (highlighted row). $^{\uparrow}$significantly lower ratio of errors on gold test compared to no-weighting scheme. McNemar's test, $p < 0.05$. $^{\Downarrow}$significantly lower 10 fold results compared to no-weighting scheme. $^{\Uparrow}$significantly higher 10 fold results compared to no-weighting scheme. Wilcoxon's test, $p < 0.05$.

	Classifiers		
Wgt	**MNB**	**LR**	**LSVM**
none	86.42 (96.52±1.00)	85.09 (94.53±1.75)	90.66 (97.40±0.90)
idf	85.97$^{\downarrow}$ (96.78±1.09)$^{\Downarrow}$	85.64$^{\uparrow}$ (95.70±1.60)$^{\Uparrow}$	90.55 (97.52±0.91)$^{\Uparrow}$
bm25	86.60 (96.72±1.04)$^{\Uparrow}$	84.75$^{\downarrow}$ (94.44±1.82)$^{\Downarrow}$	90.62 (97.50±0.99)$^{\Uparrow}$
ig	86.42 (96.52±1.00)	85.06 (94.53±1.74)	90.66 (97.40±0.90)
gr	86.42 (96.52±1.00)	85.07 (94.53±1.76)	90.66 (97.40±0.90)
mi	**88.18**$^{\uparrow}$ **(96.84±0.90)**$^{\Uparrow}$	86.45$^{\uparrow}$ (95.06±1.44)$^{\Uparrow}$	90.86 (97.50±0.77)$^{\Uparrow}$
mi'	88.06$^{\uparrow}$ (96.69±0.83)$^{\Uparrow}$	**87.09**$^{\uparrow}$ **(95.30±1.24)**$^{\Uparrow}$	**90.92**$^{\uparrow}$ **(97.46±0.77)**
χ^2	77.46$^{\downarrow}$ (89.86±0.99)$^{\Downarrow}$	77.98$^{\downarrow}$ (89.38±2.86)$^{\Downarrow}$	87.47$^{\downarrow}$ (94.62±1.17)$^{\Downarrow}$
Δ_{sm}	86.42 (96.52±1.00)	85.08 (94.54±1.76)	90.66 (97.40±0.90)
Δ_{sm2}	86.42 (96.52±1.00)	85.11 (94.52±1.75)	90.66 (97.40±0.90)
Δ_{bm25}	86.42 (96.52±1.00)	85.10 (94.54±1.78)	90.66 (97.40±0.90)
rf	87.29$^{\uparrow}$ (96.69±0.90)$^{\Uparrow}$	86.32$^{\uparrow}$ (94.97±1.60)$^{\Uparrow}$	90.67 (97.45±0.82)
ne	86.48 (96.54±0.98)$^{\Uparrow}$	85.32$^{\uparrow}$ (94.61±1.75)$^{\Uparrow}$	90.72 (97.41±0.88)
$re_{b_0=0.2}$	86.48 (96.54±0.98)$^{\Uparrow}$	85.25$^{\uparrow}$ (94.60±1.76)$^{\Uparrow}$	90.70 (97.41±0.89)
$re_{b_0=0.5}$	86.47 (96.54±0.99)	85.22$^{\uparrow}$ (94.59±1.76)$^{\Uparrow}$	90.69 (97.41±0.91)
$re_{b_0=0.7}$	86.45 (96.53±0.98)	85.21$^{\uparrow}$ (94.58±1.76)$^{\Uparrow}$	90.66 (97.41±0.91)
euclid	86.42 (96.52±1.00)	85.11 (94.53±1.76)	90.66 (97.40±0.90)
cng	86.41 (96.54±0.98)	85.60$^{\uparrow}$ (94.90±1.65)$^{\Uparrow}$	90.66 (97.37±0.94)

linear SVM with 1–4 character n-grams, word unigrams and bigrams and hyperparameter tuning ($C = 0.4$). The second and third had similar setups, where they used an ensemble of linear SVMs, one trained on character n-grams and the other on PoS tags, n-grams, and syntactical information.

GDI'18 Table 2.20 shows the results on the GDI 2018 dataset on the gold test set with 10 fold cross validation on the training set. For all three classifiers the *mi'* weighting scheme provides the biggest boost. The MNB classifier results are significantly impacted by the *idf*, *mi*, *mi'*, χ^2, *rf* and *cng* weighting schemes. The LR classifier performs much better than MNB even without weighting, where *mi*, *mi'*

Table 2.19: Results on the DFS'18 dataset (global weighting scheme)×(classifier). F_1 measure mean and standard deviation across 10 folds. $^{\downarrow}$significantly higher ratio of errors on gold test compared to no-weighting scheme (highlighted row). $^{\uparrow}$significantly lower ratio of errors on gold test compared to no-weighting scheme. McNemar's test, $p < 0.05$. $^{\Downarrow}$significantly lower 10 fold results compared to no-weighting scheme. $^{\Uparrow}$significantly higher 10 fold results compared to no-weighting scheme. Wilcoxon's test, $p < 0.05$.

	Classifiers		
Wgt	**MNB**	**LR**	**LSVM**
none	59.71 (60.92±0.35)	63.36 (63.88±0.41)	63.14 (64.49±0.49)
idf	**61.53**$^{\uparrow}$ **(63.17±0.38)**$^{\Uparrow}$	63.48 (64.74±0.46)	62.17 (64.10±0.58)
bm25	59.71 (60.92±0.35)	63.40 (63.88±0.41)	63.14 (64.49±0.49)
ig	59.71 (60.92±0.35)	63.38 (63.88±0.40)	63.14 (64.49±0.49)
gr	59.71 (60.92±0.35)	63.37 (63.88±0.40)	63.14 (64.49±0.49)
mi	60.87$^{\uparrow}$ (63.40±0.32)$^{\Uparrow}$	**63.95**$^{\uparrow}$ **(65.59±0.41)**$^{\Uparrow}$	63.00 (66.31±0.53)
mi'	60.79$^{\uparrow}$ (63.03±0.36)$^{\Uparrow}$	63.83 (65.36±0.43)	63.06 (65.97±0.51)
χ^2	59.84$^{\uparrow}$ (60.80±0.28)	63.40 (63.85±0.43)	**63.31 (64.43±0.53)**
Δ_{sm}	59.71 (60.92±0.35)	63.37 (63.88±0.41)	63.14 (64.49±0.49)
Δ_{sm2}	59.71 (60.92±0.35)	63.36 (63.88±0.41)	63.14 (64.49±0.49)
Δ_{bm25}	59.71 (60.92±0.35)	63.39 (63.88±0.41)	63.14 (64.49±0.49)
rf	59.98$^{\uparrow}$ (61.32±0.41)$^{\Uparrow}$	63.63 (64.16±0.35)	63.14 (64.54±0.48)
ne	59.77$^{\uparrow}$ (60.98±0.35)	63.43 (63.92±0.39)	63.15 (64.53±0.49)
$re_{b_0=0.2}$	59.76$^{\uparrow}$ (60.98±0.34)	63.43 (63.91±0.40)	63.14 (64.52±0.48)
$re_{b_0=0.5}$	59.77$^{\uparrow}$ (60.96±0.35)	63.43 (63.90±0.41)	63.15 (64.53±0.49)
$re_{b_0=0.7}$	59.76$^{\uparrow}$ (60.95±0.34)	63.42 (63.89±0.40)	63.14 (64.52±0.48)
euclid	59.71 (60.92±0.35)	63.39 (63.88±0.40)	63.14 (64.49±0.49)
cng	60.36$^{\uparrow}$ (61.48±0.44)$^{\Uparrow}$	63.55 (64.13±0.40)	62.97 (64.52±0.53)

and *cng* had significantly positive impact. However, χ^2 had negative impact. The best classifier (linear SVM) showed significantly the best performance even without weighting. Scheme *mi'* provided the biggest boost. The only system that participated in the task with the "unknown" class setup was evaluated at 51.2%. This task, even without the additional class was shown to be difficult. Table A.4 in Appendix A shows the confusion between classes. Most problematic is XY.

GDI'19 Table 2.21 shows the results on the GDI dataset on the gold test set with 10 fold cross validation on the training set. All classifiers showed the biggest boost with the *mi* weighting scheme, MNB obtaining the best performance, 66.12%. For MNB, *idf*, *mi* and *mi'* had positive impact, while χ^2 had negative impact. For LR, *mi*

Table 2.20: Results on the GDI'18 dataset (global weighting scheme)×(classifier). F_1 macro-averaged shown. Due to the nature of the dataset we did not perform cross validation. $^{\downarrow}$significantly higher ratio of errors on gold test compared to no-weighting scheme (highlighted row). $^{\uparrow}$significantly lower ratio of errors on gold test compared to no-weighting scheme. McNemar's test, $p < 0.05$.

	Classifiers		
Wgt	**MNB**	**LR**	**LSVM**
none	28.74	46.34	51.15
idf	33.63$^{\uparrow}$	46.34	51.39
bm25	28.74	46.34	51.18
ig	28.74	46.34	51.15
gr	28.74	46.34	51.15
mi	39.20$^{\uparrow}$	48.41$^{\uparrow}$	51.82$^{\uparrow}$
mi'	**39.62**$^{\uparrow}$	**48.60**$^{\uparrow}$	**52.04**$^{\uparrow}$
χ^2	35.63$^{\uparrow}$	43.83$^{\downarrow}$	47.07$^{\downarrow}$
Δ_{sm}	28.74	46.35	51.15
Δ_{sm2}	28.74	46.34	51.15
Δ_{bm25}	28.74	46.34	51.15
rf	31.91$^{\uparrow}$	46.98	51.44
ne	29.47	46.47	51.23
$re_{b_0=0.2}$	29.41	46.43	51.23
$re_{b_0=0.5}$	29.17	46.42	51.24
$re_{b_0=0.7}$	29.09	46.40	51.21
euclid	28.74	46.34	51.15
cng	30.42$^{\uparrow}$	46.67$^{\uparrow}$	51.34

and *mi'* had positive impact, and χ^2 had negative impact. For linear SVM, only *mi* had positive impact, and χ^2 had negative impact. Table A.5 in Appendix A shows the confusion between classes. The three best performing systems at the workshop were evaluated at 75.93%, 75.41% and 74.55%, respectively. However, led by the previous year's task success of using an adaptation technique to enhance the final results, the top three systems used it and got significantly better results on the test set (almost 10% boost). The problem with the evaluation of this approach is that the dataset that the systems were tested on was also used in the adaptation. The fourth result used no adaptation and yielded 62.55%.

MADAR'19 Table 2.22 shows the results on the MADAR dataset on the gold test set with 10 fold cross validation on the training set. We used word unigrams and

Table 2.21: Results on the GDI'19 dataset (global weighting scheme)×(classifier). F_1 measure mean and standard deviation across 10 folds. $^{\downarrow}$significantly higher ratio of errors on gold test compared to no-weighting scheme (highlighted row). $^{\uparrow}$significantly lower ratio of errors on gold test compared to no-weighting scheme. McNemar's test, $p < 0.05$. $^{\Downarrow}$significantly lower 10 fold results compared to no-weighting scheme. $^{\Uparrow}$significantly higher 10 fold results compared to no-weighting scheme. Wilcoxon's test, $p < 0.05$.

	Classifiers		
Wgt	**MNB**	**LR**	**LSVM**
none	64.82 (79.35±1.53)	64.25 (78.74±1.48)	64.65 (80.76±1.79)
idf	65.56 (81.50±1.47)$^{\Uparrow}$	64.86 (80.56±1.46)	63.98$^{\downarrow}$ (80.64±2.13)
bm25	65.48 (81.52±1.39)$^{\Uparrow}$	64.90 (80.62±1.48)	63.96$^{\downarrow}$ (80.64±2.14)
ig	64.82 (79.35±1.53)	64.25 (78.74±1.47)	64.67 (80.76±1.79)
gr	64.82 (79.35±1.53)	64.25 (78.74±1.48)	64.65 (80.76±1.79)
mi	**66.12$^{\uparrow}$ (81.86±1.51)$^{\Uparrow}$**	**65.56$^{\uparrow}$ (81.03±1.36)$^{\Uparrow}$**	**64.97$^{\uparrow}$ (81.41±1.77)$^{\Uparrow}$**
mi'	65.76$^{\uparrow}$ (81.04±1.72)$^{\Uparrow}$	**65.56$^{\uparrow}$ (80.69±1.03)$^{\Uparrow}$**	64.72 (81.39±1.71)
χ^2	56.59$^{\downarrow}$ (69.94±2.81)$^{\Downarrow}$	61.05$^{\downarrow}$ (73.62±2.30)$^{\Downarrow}$	62.15$^{\downarrow}$ (75.54±2.32)$^{\Downarrow}$
Δ_{sm}	64.82 (79.35±1.53)	64.23 (78.74±1.48)	64.67 (80.76±1.79)
Δ_{sm2}	64.82 (79.35±1.53)	64.25 (78.74±1.46)	64.67 (80.76±1.79)
Δ_{bm25}	64.82 (79.35±1.53)	64.25 (78.75±1.50)	64.67 (80.76±1.79)
rf	65.12 (79.99±1.61)$^{\Uparrow}$	64.78 (79.22±1.37)	64.77 (80.81±1.78)
ne	64.92 (79.48±1.60)$^{\Uparrow}$	64.36 (78.86±1.56)	64.71 (80.75±1.81)
$re_{b_0=0.2}$	64.90 (79.47±1.59)$^{\Uparrow}$	64.33 (78.84±1.55)	64.67 (80.74±1.81)
$re_{b_0=0.5}$	64.84 (79.43±1.57)	64.33 (78.82±1.55)	64.63 (80.76±1.82)
$re_{b_0=0.7}$	64.82 (79.40±1.57)	64.29 (78.80±1.54)	64.63 (80.76±1.80)
euclid	64.82 (79.35±1.53)	64.23 (78.75±1.47)	64.65 (80.76±1.79)
cng	65.01 (79.89±1.44)$^{\Uparrow}$	64.50 (79.07±1.32)	64.68 (80.73±1.79)

1–3 character n-grams. The best classifier was MNB with the *idf* weighting scheme. Schemes *bm25* and *cng* had positive impact, while χ^2 had negative. However, *idf* and *bm25* both had lower levels of errors than *cng* in the gold test. LR was influenced by the same weighting schemes in a similar way. Linear SVM did not have any significant gains from weighting. Surprisingly, *idf* and *bm25* had negative impact. The three best performing systems at the workshop were evaluated at 67.32%, 67.31% and 67.20%, respectively. All three methods combined discriminative and generative algorithms (language models). A preliminary study by the organizers [241] showed that incorporating language models with a classifier can boost the performance and achieved 67.5%. Without a language model, they reported 63.60%. The classifier they used is MNB with word unigrams and 1-3 character n-grams.

Table 2.22: Results on MADAR dataset (global weighting scheme)×(classifier). F_1 measure mean and standard deviation across 10 folds. $^{\downarrow}$significantly higher ratio of errors on gold test compared to no-weighting scheme (highlighted row). $^{\uparrow}$significantly lower ratio of errors on gold test compared to no-weighting scheme. McNemar's test, $p < 0.05$. $^{\Downarrow}$significantly lower 10 fold results compared to no-weighting scheme. $^{\Uparrow}$significantly higher 10 fold results compared to no-weighting scheme. Wilcoxon's test, $p < 0.05$.

	Classifiers		
Wgt	**MNB**	**LR**	**LSVM**
none	61.77 (59.89±4.18)	61.61 (60.03±4.37)	62.03 (60.58±4.53)
idf	**64.31**$^{\uparrow}$ **(62.06±4.31)**$^{\Uparrow}$	**63.42**$^{\uparrow}$ **(61.69±4.50)**$^{\Uparrow}$	61.51$^{\downarrow}$ (60.29±4.58)$^{\Downarrow}$
bm25	64.29$^{\uparrow}$ (62.08±4.30)$^{\Uparrow}$	63.36$^{\uparrow}$ (61.72±4.50)$^{\Uparrow}$	61.52$^{\downarrow}$ (60.29±4.61)$^{\Downarrow}$
ig	61.63 (59.86±4.11)	61.30 (59.96±4.45)	61.87 (60.68±4.49)
gr	61.77 (59.89±4.18)	61.61 (60.02±4.38)	62.03 (60.58±4.53)
mi	62.00 (59.76±4.39)	62.24 (60.49±4.47)	**62.37 (60.62±4.67)**
mi'	60.52 (59.15±4.18)	62.17 (60.32±4.42)	62.29 (61.19±4.50)
χ^2	47.98$^{\downarrow}$ (47.96±3.09)$^{\Downarrow}$	52.78$^{\downarrow}$ (52.54±3.34)$^{\Downarrow}$	56.30$^{\downarrow}$ (55.69±3.40)$^{\Downarrow}$
Δ_{sm}	61.77 (59.89±4.18)	61.60 (60.02±4.37)	62.03 (60.58±4.53)
Δ_{sm2}	61.77 (59.89±4.18)	61.60 (60.03±4.37)	62.03 (60.58±4.53)
Δ_{bm25}	61.77 (59.89±4.18)	61.60 (60.03±4.37)	62.03 (60.58±4.53)
rf	61.77 (59.95±4.15)	61.66 (60.06±4.40)	62.03 (60.60±4.52)
ne	61.80 (59.88±4.17)	61.69 (60.06±4.32)	62.07 (60.62±4.56)
$re_{b_0=0.2}$	61.84 (59.88±4.15)	61.67 (60.06±4.31)	62.05 (60.62±4.56)
$re_{b_0=0.5}$	61.86 (59.88±4.12)	61.66 (60.05±4.32)	62.05 (60.62±4.54)
$re_{b_0=0.7}$	61.82 (59.87±4.13)	61.66 (60.04±4.34)	62.01 (60.61±4.50)
euclid	61.77 (59.89±4.18)	61.61 (60.02±4.37)	62.03 (60.58±4.53)
cng	62.02$^{\uparrow}$ (60.18±4.28)$^{\Uparrow}$	61.68 (60.28±4.33)	61.93 (60.50±4.54)

Table 2.23: The number of significantly improved results (↑) and significantly worsened results (↓) using standard schemes. The number indicates how many datasets had significant improvement per weighting scheme and classifier. Maximum 9.

	Clf.	*idf*	*bm25*	*ig*	*gr*	*mi*	*mi'*	χ^2	Δ_{all}	*rf*	*ne*	$re_{b_0=x}$
↑	**MNB**	5	3	-	-	5	4	2	-	4	1	1
	LR	3	3	-	-	6	6	1	-	3	1	2
	LSVM	1	2	-	-	3	3	-	-	-	-	-
↓	**MNB**	1	-	-	-	2	1	6	-	-	-	-
	LR	-	1	-	-	1	1	6	-	-	-	-
	LSVM	2	2	-	-	-	-	7	-	-	-	-

Overall results are summarized in Tables 2.23 and 2.24.

Table 2.24: The number of significantly improved results (↑) and significantly worsened results (↓) using proposed schemes. The number indicates how many datasets had significant improvement per weighting scheme and classifier. Maximum 9.

	Clf.	*euclid*	*cng*
↑	**MNB**	-	3
	LR	-	4
	LSVM	-	1
↓	**MNB**	-	-
	LR	-	-
	LSVM	-	-

2.9 Conclusion and Future Work

In this chapter we examined a couple of questions in the domain of automatic Language Identification. First, we explored the possibility of using the CNG algorithm for language relatedness analysis. Influenced by a study from Gamallo *et al.* [93] published in 2017, we applied our method and showed that CNG is appropriate for such a task. Nevertheless, "appropriateness" can be disputed by linguistic experts who claim that language similarity cannot and should not be represented as a single distance value. Our goal was not to disprove this, but to offer an automated way of measuring one aspect of language similarity. Second, we noticed the lack of a comprehensive study on the effect of feature weighting techniques in the domain of short texts. Additionally, we proposed two supervised weighting schemes, first based on Euclidean distance, and second based on CNG. So far, the usual weighting techniques used are *idf* and less frequently *bm25*. We considered 13 schemes found in the literature related to text classification tasks and 2 of our own. The general findings are:

- scheme *idf* is not always the best weighting method, it depends on the dataset;
- scheme χ^2 does not seem to work so well in the explored settings (short texts, very similar classes);
- mutual information (*mi*) and modified mutual information (*mi'*) can boost the results significantly compared to *no-weighting* and *idf*;
- the CNG-based weighting scheme shows promising results, as it was better than

no-weighting and *idf* on most of the explored datasets.

Because our goal was to explore the effect of various weighting schemes, we did not focus on the architecture of the classifier, nor hyperparameter tuning. One possible extension of this study is to work on these two aspects. Neural Networks, especially LSTMs, gained a lot of traction when in comes to language modelling. Although we did some preliminary experiments with NNs and LSTMs (not reported in this dissertation), our findings were that the datasets are likely too small to build neural models that are on par with probabilistic models. To support this claim, more thorough study should be conducted as part of the future work.

Chapter 3

Twitter Bot Detection using Digital Fingerprints and Diversity Measures

3.1 Introduction

Millions of Internet users collaborate and communicate through SNSs. One of the most influential platforms, at least in the western world, is Twitter. It provides free microblogging services, where users can share their ideas, news, advertise, promote figures, products and agendas. However, the openness and convenience of the Twitter platform has its drawbacks. It facilitates the creation and usage of automated malicious accounts whose aim is to manipulate other users for the sake of financial gain or to spread fake information. An automated user (bot) is a program that emulates a real person's behavior on social media. A bot can operate based on a simple set of behavioral instructions, such as tweeting, retweeting, "liking" posts, or following other users. In general, there are two types of bots based on their purpose: non-malicious and malicious [268]. The non-malicious bots are transparent, with no intent of mimicking real Twitter users [234]. Often, they share motivational quotes or images, tweet news headlines and other useful information [161], or help companies to respond to users [246]. On the other hand, malicious ones may generate spam, try to access private account information, trick users into following or subscribing to scams [1, 74, 91, 321], suppress or enhance political opinions [109, 228], create trending hashtags for financial gain, support political candidates during elections [22, 68], create fake online reviews [78, 177] or create offensive material to troll users. Additionally, some influencers may use bots to boost their audience size. Boshmaf *et al.* [35, 36] show that OSNs are vulnerable to infiltration of bot networks into legitimate social communities for the sake of collecting of sensitive personal data. Paradise *et al.* [209] extended their previous study to what kind of infiltration strategies work best, and they found that the most effective attack is randomly sprayed

friend requests. Similarly, Bokobza *et al.* [32] explored infiltration strategies on two social networks Flickr[1] and Twitter. They found that social communities whose social network topologies have low clustering coefficients are more vulnerable to infiltration. There has been some research related to the effectiveness of bot networks. A few authors designed covert bot networks by exploiting the social habits of individual users and using *steganography* techniques to conceal bot communication [192, 208].

At first, automated users sharing random bits of information across Twitter may not seem like a threat, but bots can potentially jeopardize online user security. Bots on social media platforms generate spam content and degrade overall user experience. With the growth of social networks and their influence in news and information sharing, bots have become a serious threat to democracies. The "foreign actors" use bots to share politically polarizing content in the form of fake news in order to increase its influence or intentionally promote certain people and their agendas. Counter-measures are needed to combat these coordinated influence campaigns. Bots are constantly evolving and adapting their behaviour to mimic real users. Nevertheless, many of these bots are coordinated [47], which means that they can show similar behaviour. This characteristic can be used to develop models for bot detection.

We experiment on bot detection techniques based on users' temporal behaviour. Additionally, we apply a set of statistical diversity measures to describe how diverse the user behaviour is over an extended period of time. Using datasets from two different researchers [55, 290] and a dataset from the PAN Author Profiling task [227] we examine if the automated accounts have less diverse behaviour than genuine user accounts and if these measures can help in detecting automated behaviour without diving into language-specific analyses. Second, we explore if the way the dataset is collected affects the ability of the measures to capture the difference between bot and human accounts.

3.2 Related Work

To battle the problem of Online Social Networks (OSNs) malicious content, a fairly large research community got involved. One of the most prominent tasks in recent

[1]Flickr is an image hosting service and video hosting service created in 2004.

social media analysis is detection of automated user accounts (bots). As said, research on this topic is very active [180, 313], because bots pose a big threat if they're intentionally steered to target important events across the globe, such as political elections [22, 102, 117, 265, 290]. There are a few thorough survey papers that try to cover the most important developments in the area of social media malicious content detection. These include Verma *et al.* [292] from 2014, Kabakus *et al.* [133] from 2017, and the most recent one at the time of writing by Wu *et al.* [309].

Social media has been used in political communication over the last couple of years. Debates over political topics and political campaigns are a common occurrence on Twitter [81]. The platform makes it easier to broadcast information quickly and to a large group of people, which are some of the reasons why it gained such importance. However, information manipulation in political sense can have moral, ethical and legal implications. A few of the recent political situations include the Brexit debate in 2016 [117], the US presidential election in 2016 [22, 102], German state elections in 2017 [38], the Catalan independence referendum in 2017 [265], the Brazilian presidential election in 2019 [275], the political scene in Venezuela [86]. Edwards *et al.* [72] examined how well automated accounts communicate versus human accounts. Their study suggests that a TwitterBot agent is perceived differently than a human agent on variables related to perceptions of communication quality. However, Goga *et al.* [99] found that that in the case of identity impersonation attacks on Twitter, it can impact any user and the attackers are creating real-looking accounts that are harder to detect by current systems. Similarly, Elyashar *et al.* [74] experimented on Facebook and Xing[2] (German-based competitor to LinkedIn) networks where they demonstrated how easy it is to create malicious social bots that mimic social network "friends", which makes users vulnerable to exposing too much information about themselves (personal and career info). Everett *et al.* [76] found that a typical Internet user is twice as likely to be deceived by automated content than a security researcher. Also, they found that entertainment and adult content is significantly easier target for deception. Flores *et al.* [85] developed a system supported by social media bots acting as recruiters to help non-profit organizations recruit volunteers with specialized knowledge. He *et al.* [108] identified several behavioral features of

[2] `https://www.xing.com/`

social bots and based on them proposed a bot detection system. Stieglitz *et al.* [267] analysed the data from the 2016 US presidential election and found that there are some differences in behaviour between humans and bots such as the number of followers, retweets and used links per day of an account. Interestingly, Luceri *et al.* [164] showed that social bots can be classified according to their political leaning. They also showed that conservative bots share most of the topics of discussion with their human counterparts, while liberal bots show less overlap and a more inflammatory attitude. Thieltges *et al.* [278] discussed related ethical questions in relation to the bot detection development process by examining the trade-off among three variables which characterize a machine learning method: accuracy, transparency and robustness. Accuracy represents a rate at which the ML method makes the right decision, transparency is the openness of the ML method (the features and the underlying learning algorithm are not a black box) and robustness represents the number of features being used. They argue that the more transparent the approach is, the more likely it is that the robustness and accuracy decrease over time (bot creators adjust their new bots to bypass detection).

There are a couple of ways to categorize the approaches to bot detection. Ferrara *et al.* [80] discussed the approaches in a more general way, dividing it to three groups which include: social graph-based, crowdsourcing-based and feature-based methods. The categorization laid out by Wu *et al.* [309] seemed more relevant to our work. Guided by their comprehensive taxonomy, we briefly describe representative papers for each category. The research is divided into the following groups:

1. Syntax analysis
    (a) Key segment (URLs, keywords and username patterns)
    (b) Tweet content (term frequency, bag of words and sparse learning)
2. Feature analysis
    (a) Statistical information (tweet & account, only tweet and campaign)
    (b) Social graph (graph based and neighbourhood based)
3. Blacklist

Detection based on syntax analysis These methods rely on using segments such as keywords, username patterns and URLs to represent the context of tweets and their authors. The motivation behind it is that malicious accounts tend to contain deceptive URLs, keywords or usernames to mimic a genuine account. It is shown that a malicious tweet is more likely to contain an appealing title followed by an external URL [50, 51].

Detection based on feature analysis Bot detection approach by Cresci *et al.* [55] is based on DNA-inspired fingerprinting of temporal user behaviour. They defined a vocabulary B^n, where n is the dimension. An element of the vocabulary represents a label for a tweet. User activity is represented as a sequence of tweet labels. They found that bots share Longer Common Substrings (LCSs) than regular users. The point where LCS has the biggest difference is used as a cut-off value to separate bots from genuine users. In the follow-up study [56] they benchmark several state-of-the-art techniques proposed by the related literature and compare results with their approach. They show that Twitter, humans, and cutting-edge applications are still in its infancy, and they are not accurate in detecting the new social spambots. Similarly, Chavoshi *et al.* [46, 47] were rather focused on temporal user behaviour on Twitter. They used Dynamic Time Warping (DTW) distance to capture bot networks. The intuition behind this is that bots inside bot networks follow the same tweeting pattern (repeated tweeting at approximately the same time over a certain period of time).

Statistical information A relatively early paper on Twitter bot detection [298] uses features such as follower and friend numbers, number of duplicate tweets in a 20 tweet sequence, number of URLs and number of mentions, and four different classifiers among which Naïve Bayes classifier performed best. Alarifi *et al.* [6] analyzed the meta-attributes of users and found that statistics of the tweets (character number and standard deviation, number of hashtags and mentions, links, number of retweets, favorites etc.) can help in identifying automated accounts. Dickerson *et al.* [68] showed that a collection of network-, linguistic-, and especially sentiment-based features can improve the identification of bots. The Twitterati [170] bot identification system is based on features including inter tweet delay (the frequency between consecutive tweets), whether the content contains spam keywords, near duplicate tweets,

Klout score[3] and the tweeting device. It uses simple C4.5 (decision tree) to distinguish humans and bots. Igawa *et al.* [119] proposed an interesting approach based on a Discrete Wavelet Transform (DWT) feature vector representation of each user account in conjunction with a new weighting scheme they call Lexicon Based Coefficient Attenuation (LBCA). Their system distinguishes three classes: humans, bots and cyborgs (bot assisted human accounts) and shows an accuracy over 94%. A follow-up study [132] applies the same method on different data where they distinguish different classes: humans, legitimate bots and fraudulent bots.

Graph-based Authors [253] explore methods for fake news detection on social media, which is closely related to the problem of automated accounts. They state that the performance of detecting fake news only from content in general doesn't show good results, and they suggest to use user social interactions as auxiliary information to improve the detection. OCTracker [23] is a framework developed for tracking overlapping community evolution in OSNs. It models a community structure and dynamic changes in social networks using a density-based approach. Their follow-up study [24] uses the framework to detect spammers in online social communities. SybilRank [41] is a proposed system that relies on social graph properties to rank users according to their perceived likelihood of being fake. Another system called Íntegro [34] proposes an amalgamation of graph-based features and account-based features. Their approach outperformed SybilRank by a large margin (Area Under the ROC Curve (AUC)=0.92, which was 30% greater than SybilRank; ROC - Receiver Operating Characteristic Curve). Yang *et al.* [313] found that SVM and a threshold classifier in conjunction with selected features that include friend request frequency, outgoing requests accepted, incoming requests accepted and clustering coefficient (mutual connectivity) obtain an accuracy between 98.68% and 99.50%. They conducted the experiments on the data from Beijing-based OSN Renren[4]. A framework by Ahmed *et al.* [4] for bot detection uses the Euclidean distance between feature vectors to build a similarity graph of the accounts. After the graph is built, they perform clustering and community detection algorithms to identify groups of similar accounts in the graph. Messias *et al.* [180] explored strategies of how a bot

[3]Klout was a Web service that used social media analytics to rank its users according to online social impact. The service is not available anymore, since the company got acquired.

[4]`http://renren.com/`

can interact with real users to increase their influence. They show that a simple strategy can trick influence scoring systems. BotOrNot [60] is an openly accessible solution available as API for the machine learning system for bot detection. They showed that the system is accurate in detecting social bots. Ferrara *et al.* [81] used an extensive set of features (tweet timing, tweet interaction network, content, language and sentiment) to detect online campaigning as early as possible.

Bot problem on social media platforms inspired many competitions and evaluation campaigns such as DARPA Twitter Bot Challenge [270] and PAN Author Profiling Task [58, 223, 227][5].

Gender Identification

Gender and age identification are common author profiling problems explored in the literature. Most research agrees that stylistic features are key to distinguishing users by gender and age. Although not perfect, it is an established basis for a successful identification system. The research on this topic is big and active, but we draw a limit by describing a few most relevant, recent and influential studies.

One of the earlier works [100] uses slang words and variation in sentence length in addition to standard vocabulary words and part of speech. It shows significant improvement (about 10%) compared to using vocabulary words only. The dataset consists of 20K blog posts collected in 2004 from Blogger[6]. Thelwall *et al.* [277] explored the correlation between sentiment and age and gender on the MySpace social network. They found that women are more likely to give and receive positive comments compared to men. They argue that women are more successful OSN users partly because of their greater ability to textually harness positive affect. Sarawgi *et al.* [245] conducted a thorough study on a cross-topic dataset and explored the effect of a topic on detection performance. They used a Probabilistic Context-Free Grammar (PCFG) model, two shallow word-level and character level language models and bag of words (BoW). Peersman *et al.* [212] studied the prediction of gender and age on a corpus of chat texts from the Belgian social networking site Netlog[7]. They used character and word n-grams and linear SVMs. An interesting observation was

[5] `https://pan.webis.de/publications.html`
[6] `https://www.blogger.com/`
[7] The service has been unavailable since 2015.

described by Dadvar *et al.* [57]. They found that men and women use different foul words and that gender-specific features can improve cyberbullying detection.

Schwartz *et al.* [249] extracted 700M words, phrases, and automatically generated topics from Facebook and correlated them with gender, age, and personality. They used a popular tool in psychology called Linguistic Inquiry and Word Count (LIWC)[8] to extract psychometric features from the raw text.

A language independent approach [10] was proposed which relies on five color-based features extracted from Twitter profiles (such as profile background color). Magno *et al.* [169] conducted an experiment on data from the Google+ social network to study gender differences in 73 countries. They found a strong correlation between online indicators of inequality and established offline indicators. Using social graph link analysis they showed that women in less developed countries with larger gender differences have a higher social status online as measured in terms of number of followers (in-degree links). Author profiling focusing on age and gender identification in Roman Urdu and English languages was also studied [77]. They used a set of stylometry, word, character and text richness-based features which are standard recommended features for this task. Stylometry and statistical readability measures have been successfully applied in spam email detection tasks [250, 251].

The best performing system at the PAN Author Profiling Workshop in 2019 was by Pizzaro *et al.* [220]. For both bot and gender tasks, their best model was linear SVM with character and word n-grams. They did not apply any novel methods, except a thorough hyperparameter tuning.

3.3 Digital fingerprint of user online behaviour

DNA sequences have been exploited in different areas such as forensics, anthropology, and biomedical science. Cresci *et al.* [55] used the idea of DNA coding to describe social media user behaviour in the temporal dimension. The same idea was used in this study, with a slightly modified way of coding. We define a set of codes A_n with

[8] `http://liwc.wpengine.com/`

the length $n = 6$. The meaning of each code is given in (3.1).

$$A_6 = \begin{cases} 0, & \text{plain} \\ 8, & \text{retweet} \\ 16, & \text{reply} \\ 1, & \text{has hastags} \\ 2, & \text{has mentions} \\ 4, & \text{has URLs} \end{cases} \tag{3.1}$$

Each character in the vocabulary A_n, which describes a tweet, is constructed by adding up codes for tweet features. The first three codes describe the type of the tweet (retweet, reply, or plain) and the rest describe the content of the tweet. For example, if a tweet is neither retweet nor reply, it is plain (with the $code = 0$). If the tweet contains hashtags, then $code = code + 1$, If the same tweet contains URLs, then $code = code + 4$. Final tweet code is 5. We transform it to a character label by using ASCII character table indexes: $ASCII\_tbl[65 + 5] = F$. Hence, the vocabulary, given the code set A_n, consists of $3 * 2^3 = 24$ unique characters. The number of tweets with attributes encoded with characters determines the length of the sequence. The sequence, in our case, is simply the length of a user timeline, that is, actions in chronological order with the appropriate character encoding.

An example of a user fingerprint generated from their timeline looks like:

$$fp_{user} = (ACBCASSCCAFFADADFAFASCB...)$$

3.3.1 Fingerprint segmentation using n-gram technique

To calculate data statistics, we extracted n-grams of different length (1−, 2− and 3−grams appeared to work best). An example of 3-gram extraction of a sample user fingerprint is shown in Fig. 3.1.

N-gram segments are used to calculate richness and diversity measures, which seem to unveil the difference between genuine user and bot online behaviour.

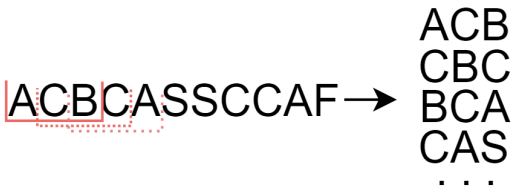

Figure 3.1: 3-gram extraction example from user fingerprint.

3.4 Statistical Measures for Text Richness and Diversity

Statistical measures for diversity have a long history and wide application [287]. A constancy measure for a natural language text is defined in this thesis as a computational measure that converges to a value for a certain amount of text and remains invariant for any larger size. Because such a measure exhibits the same value for any size of text larger than a certain amount, its value could be considered as a text characteristic. Common labels used are: N is the total number of words in a text, $V(N)$ is the number of distinct words, $V(m, N)$ is the number of words appearing m times in the text, and m_{max} is the greatest frequency over all words.

3.4.1 Yule's K Index

Yule's [314] original intention for the K Index use was for the author attribution task, assuming that it would differ for texts written by different authors.

$$K = C\frac{S_2 - S_1}{S_1^2} = C\Big[-\frac{1}{N} + \sum_{m=1}^{m_{max}} V(m,N)(\frac{m}{N})^2\Big]$$

To simplify, $S_1 = N = \sum_m mV(m,N)$, and $S_2 = \sum_m m^2V(m,N)$. C is a constant originally determined by Yule, and it is 10^4.

3.4.2 Shannon's H Index

Shannon's diversity index (H) is a measure that is commonly used to characterize species diversity in a community. Shannon's index accounts for both abundance and evenness of the species present. The proportion of species i relative to the total number of species (p_i) is calculated, and then multiplied by the natural logarithm of this proportion ($ln(p_i)$). The resulting product is summed across species, and

multiplied by -1.

$$H = -\sum_{i=1}^{V(N)} p_i ln(p_i)$$

$V(N)$ is the number of distinct species.

3.4.3 Simpson's D Index

Simpson's diversity index (D) is a mathematical measure that characterizes species diversity in a community. The proportion of species i relative to the total number of species (p_i) is calculated and squared. The squared proportions for all the species are summed, and the reciprocal is taken.

$$D = \frac{1}{\sum_{i=1}^{V(N)} p_i^2}$$

3.4.4 Honoré's R Statistic

Honoré (1979) proposed a measure which assumes that the ratio of hapax legomena $(1, N)$ is constant with respect to the logarithm of the text size:

$$R = 100 \frac{log(N)}{1 - \frac{V(1,N)}{V(N)}}$$

3.4.5 Sichel's S Statistic

Sichel [254] observed that the ratio of hapax dis legomena $V(2, N)$ to the vocabulary size is roughly constant across a wide range of sample sizes.

$$S = \frac{V(2, N)}{N}$$

We use this measure to express the constancy of n-gram *hapax dis legomena* (number of n-grams that occur two times) which we show to be distinct for genuine and bot accounts.

Fig. 3.4 shows the comparison of density plots of all measures of bot accounts versus genuine users.

3.5 Measures for Text Readability

For the gender identification task, we additionally use two readability measures designed for English and Spanish. In 1949, Flesch [84] designed a readability test score used to indicate how difficult a passage in English is to understand.

$$readability_{en} = 206.835 - 1.015 \times \frac{total\ words}{total\ sentences} - 84.6 \times \frac{total\ syllables}{total\ words}$$

The score scale is 0-100, where, for example, range 0-30 indicates low readability, but understood by university graduates, while 90-100 indicates very readable text, easily understood by an 11 year old student.

In 1959 Fernandez [79] introduced the equivalent score for Spanish.

$$readability_{es} = 206.84 - 0.6 \times (total\ syllables) - 1.02 \times (total\ words)$$

We chose to use the readability score as a feature based on interesting findings in the related literature [9, 110]. However, it is important to note that we did not conduct detailed analysis of the features for the gender identification subtask, as our focus was on the bot identification task. The dataset provided by the organizers of the PAN Author Profiling Workshop included bot and gender identification in the same task.

3.6 Methodology

We conducted the experiments with five different algorithms: Support Vector Machines, Logistic regression, K nearest neighbours and two ensemble methods — Random Forest and Gradient Boosting. The implementation was done using the *scikit-learn* machine learning package in python. For hyper-parameter tuning we used grid search cross validation method for every classifier. Extensive grid searches didn't show significant improvement for the classifiers from using the default parameters provided in the library. The only improvement was observed with the linear SVM classifier. We applied all classifiers on different number of n-grams (1-3), where combinations were: 1, $1 + 2$, and $1 + 2 + 3$. We ran three experiments on all classifiers. The first is 10-fold cross validation on the Cresci dataset, the second is 10-fold cross

Figure 3.2: Experimental setup for bot detection task. *N-gram sizes with length setting 1–4; **5 statistical measures; ***5 different classifiers.

validation on the Varol dataset, and the third is the experiment on classifiers with the entire Cresci dataset training and the entire Varol dataset for validation. With the first and second experiments the aim was to explore how important it is for a dataset to be collected in a shorter time frame versus extended period of time, which is the case with the observed datasets. The third experiment is designed to test if the dataset with better results can improve the performance of the second dataset. The architecture is shown in Fig. 3.2.

3.7 Evaluation on Cresci and Varol datasets

In the following subsections we briefly describe the datasets.

3.7.1 The Cresci (2017) Dataset

This dataset was obtained from Cresci *et al.* [56] in the form that was used in the original study. The Twitter dataset constitutes the real-world data used in our experiments. Table 3.1 reports the number of accounts and tweets they feature. According

	Users	**Tweets**
Genuine	3,474	8,377,522
Spambots #1	991	1,610,176
Spambots #2	3,457	428,542
Spambots #3	464	1,418,626
Total	8,386	11,834,866

Table 3.1: The Cresci 2017 dataset.

to Cresci *et al.* [56] the genuine accounts are a random sample of genuine (human-operated) accounts. The social spambots #1 dataset was crawled from Twitter during

the Mayoral election in Rome 2014. Spambots #2 dataset is a set of tweets from a group of bots who spent several months promoting a specific hashtag. Spambots #3 group advertised products on sale on Amazon.com. The deceitful activity was carried out by spamming URLs pointing to the advertised products.

3.7.2 The Varol (2017) Dataset

This dataset is made available by Varol *et al.* [290] on the website[9]. The dataset in the original study consisted of 3,000 user accounts manually annotated by four volunteers. At the time of download of the labeled user ids, the dataset consisted of 2,573 annotated samples. However, when we crawled the bot accounts, some of the users were banned or had a protected profile. The final dataset in this study consists of 2,115 accounts. Table 3.2 shows how many accounts were lost per class.

	Total	**Genuine**	**Bots**
Original	2,573	1,747	826
Used in study	2,115	1,421	694

Table 3.2: The Varol 2017 dataset.

The dataset was crawled on January 5th, 2019 and it contains 5,261,940 tweets. The number of tweets per user ranges from 20 to 3,250 (we filtered out accounts that have fewer than 20 tweets). Data imbalance is evident in the original annotated dataset, as well as the reduced one.

3.7.3 Experiments

A t-SNE visualization of both datasets is shown in Fig. 3.3. Features used for the visualization are same as for the classifiers (diversity measures of fingerprint n-grams). The Varol dataset (the figure on the left (a)) appears to have more confusion between the genuine and bot samples, but the separation is still visible. The right hand figure *(b)* shows the Cresci dataset where we coloured separately the three types of bots and the genuine accounts. It is interesting to notice that the three types of bots appear to be distinct groups in the feature space. This can be explained with how the datasets were collected. While the Cresci dataset is collected around targeted

[9] `https://botometer.iuni.iu.edu/bot-repository/datasets.html`

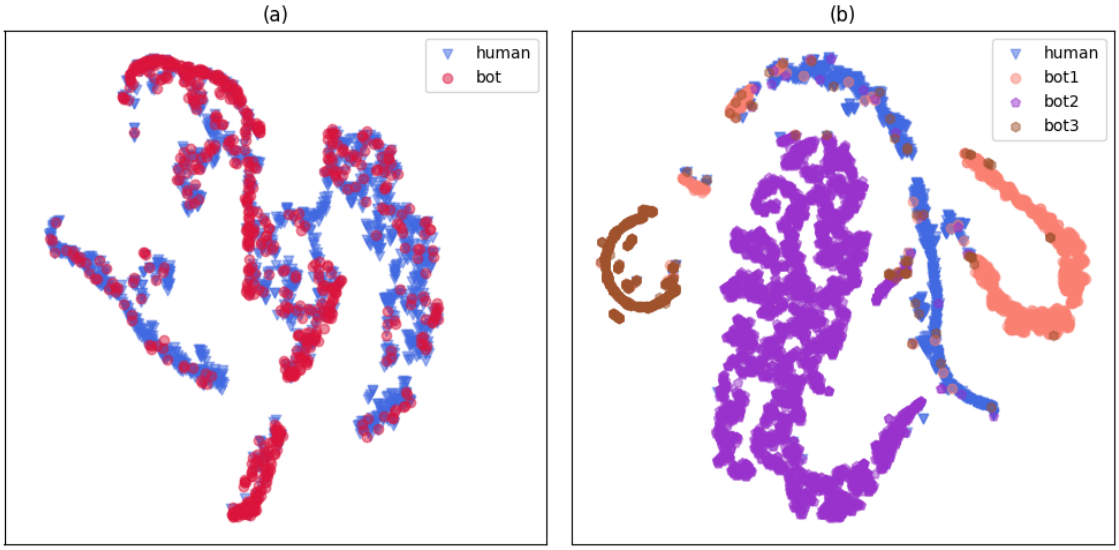

Figure 3.3: t-SNE representation: (a) Varol dataset and (b) Cresci dataset.

events in a certain time-frame, the Varol dataset is a collection of accounts that may or may not be connected by the same background topic.

Feature extraction consists of user behaviour fingerprint generation, n-gram segmentation (where n is 1, 2 and 3), and finally, diversity measures calculation on n-gram population per sample. Fig. 3.4 illustrates the density differences of each measure for all n-grams. The figure shows that the selected measures uncover the difference between automated and genuine users. Shannon's and Simpson's indices were able to capture the differences between bot networks in the Cresci dataset, besides the difference from genuine accounts. The last two measures mentioned in Section 3.4, Honoré's and Sichel's measures, are developed for natural language text. Both of them measure features that naturally occur in texts — *hapax legomena* (words that occur once in a sample) and *hapax dis legomena* (words that occur twice in a sample). However, Fig. 3.4 (bottom five diagrams) illustrates that the genuine and bot accounts show slight differences as well.

3.7.4 Results and Discussion

We report the results of the experiments using the F_1 measure (Table 3.3). The values represent the average of 10-fold validation scores. First, we analyze the use of statistical diversity of n-grams as features for the set of different classifiers and

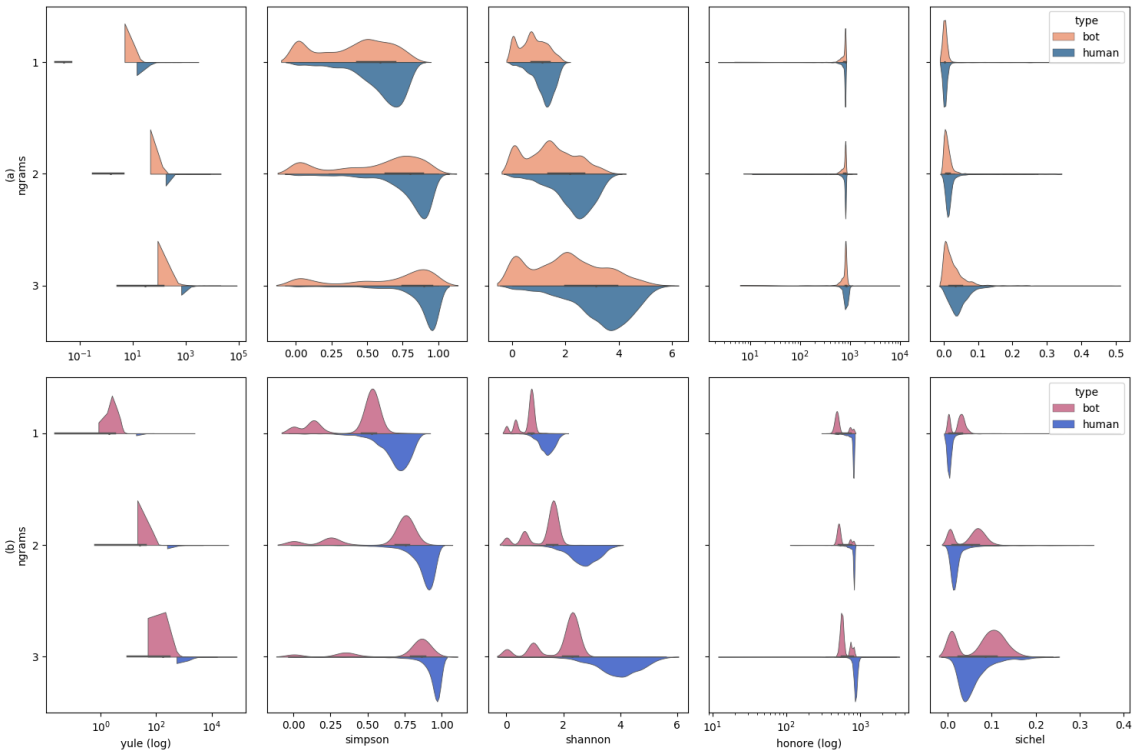

Figure 3.4: Diversity measure distributions for Varol (top) and Cresci (bottom) datasets.

the effect of increasing the n-gram order on the performance of the models. Training the Random Forest classifier on n-grams shows an increase in the performance for both datasets. However, the increase is slight with the increase in the number of n-grams from 1 to 3. The Random Forest classifier has the best performance with the F_1 average 96.67% for Experiment 1, and 73.06% for Experiment 2. Second, we can observe the dramatic difference in performance between the two datasets. In the data visualizations (Fig. 3.3 and Fig. 3.4) the data separation in the Varol dataset is somewhat worse than in the Cresci dataset, and this is reflected in the classifiers' performance. Our argument is that this is due to different data collection techniques. As mentioned earlier, the Cresci dataset was collected around specific events and using keywords, so the users, especially bots, have correlated behaviour. On the other hand, the Varol dataset was collected (directly from Twitter, given the provided labeled ids) two years after the first study performed by the original researcher [290]. The differences between human and bot accounts are less distinguished, but still show significant difference according to the diversity measures. In our third experiment, we

used the entire Cresci dataset to train the models (we used rhe best parameters from Experiment 1 for each model setup) and tested it on the entire Varol dataset. The results obtained were very similar to the ones in Experiment 2, and we did not gain much of an improvement. The best classifier performance was obtained with SVM, and a combination of 1-, 2- and 3-gram features reaching average $F_1 = 74.03\%$.

Table 3.3: 10-fold validation on datasets, F_1 measure shown. * - results are using the entire Varol dataset as test for Cresci trained classifiers. $^{\uparrow}$ - significantly lower ratio of errors than the rest of the unmarked runs in the column using McNemar's test with $p < 0.05$. $^{\Uparrow}$ - significantly different than the rest of the unmarked runs in the column using Wilcoxon's test with $p < 0.05$.

Features	Classifier	Cresci'17	Varol'17	Varol'17.test*
1-gram	**GB**	95.18±6.14	72.29±5.07$^{\Uparrow}$	68.52
	LSVM	95.77±5.66	66.31±8.58	71.79
	LR	95.80±5.29	67.87±8.89	71.65
	KNN	95.52±7.90	66.44±5.60	70.53
	RF	95.74±8.01	69.19±5.44	71.79
1+2-gram	**GB**	95.96±6.10	72.21±5.26$^{\Uparrow}$	71.12
	LSVM	96.08±5.15	68.24±8.72	72.69$^{\uparrow}$
	LR	96.17±5.19	70.41±8.41	72.78$^{\uparrow}$
	KNN	96.43±6.60$^{\Uparrow}$	69.89±7.22	72.64$^{\uparrow}$
	RF	96.43±6.48$^{\Uparrow}$	71.40±6.56	71.38
1+2+3-gram	**GB**	96.45±5.33$^{\Uparrow}$	72.33±4.55$^{\Uparrow}$	72.17
	LSVM	96.20±4.62	68.83±8.61	**74.03**$^{\uparrow}$
	LR	96.33±4.93	69.69±8.89	72.78$^{\uparrow}$
	KNN	96.33±6.97	70.57±6.53	72.32
	RF	**96.67±6.18**$^{\Uparrow}$	**73.06±7.58**$^{\Uparrow}$	73.11$^{\uparrow}$

In Fig. 3.5 we show a pruned estimator from the Random Forest classifier trained on the Cresci dataset with diversity measures on unigrams. The most influential feature for this classifier is Simpson's diversity measure (root). The separation between bot and human is on 2.79 value. The accounts which have less or equal the value are more likely to be bots. Other measures, such as Shannon on the second level, separate accounts further. To note, this is a pruned classifier with maximum depth of 3, while in Table 3.3 we did not have depth constraint (default setting in `scikit-learn`). This classifier has average F_1 measure of 95.48% (±5.08) using 10-fold validation.

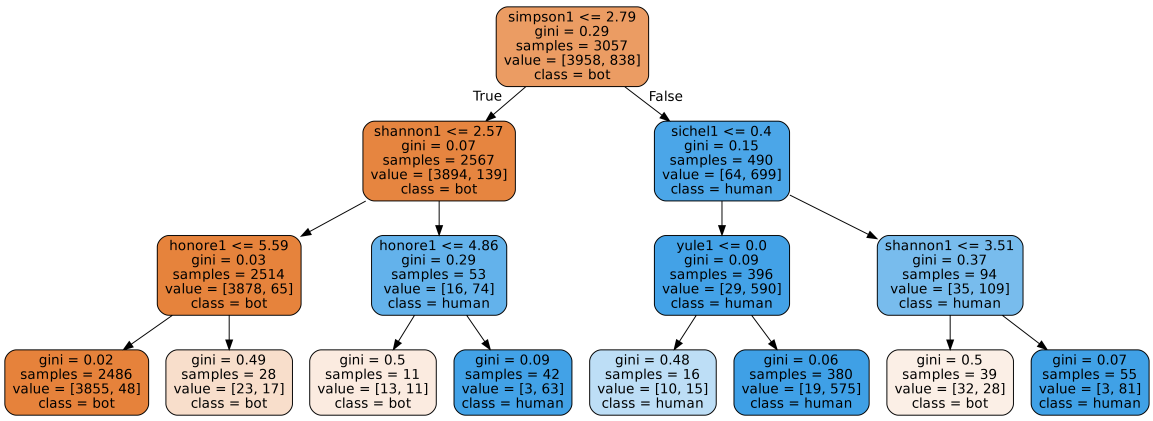

Figure 3.5: Example decision tree estimator from Random Forest classifier. Cresci dataset.

3.8 Experiments with the PAN Author Profiling Task

3.8.1 Spanish and English datasets

The dataset provided by the organizers of the PAN Author Profiling Task [227] is divided into two parts: English and Spanish. The English dataset consists of training and development subsets, with 2,880 and 1,240 samples, respectively. The Spanish dataset is slightly smaller and consists of training and development subsets, with 2,080 and 920 samples, respectively. Each sample is a user timeline in chronological order, with 100 messages per user. Fig. 3.6 and Fig. 3.7 show the datasets using t-SNE [167], an enhanced method based on stochastic neighbour embedding. The features used for both visualizations are the ones used for the classifiers in the final submitted run (Experiment 4 for bots, and Experiment 5 for gender).

3.8.2 Bot Identification

In Fig. 3.8 we show the comparison of density plots of all diversity measures of bot accounts versus genuine users. We can see that the diversity measures are different for bots and genuine users. We exploit this characteristic to build a good classifier with as few features as possible.

For the bot identification sub-task we conducted four experiments with five different classifiers (Gradient Boosting, Random Forest, SVM, Logistic Regression, K Nearest Neighbours). The differences between the experiments are more focused on

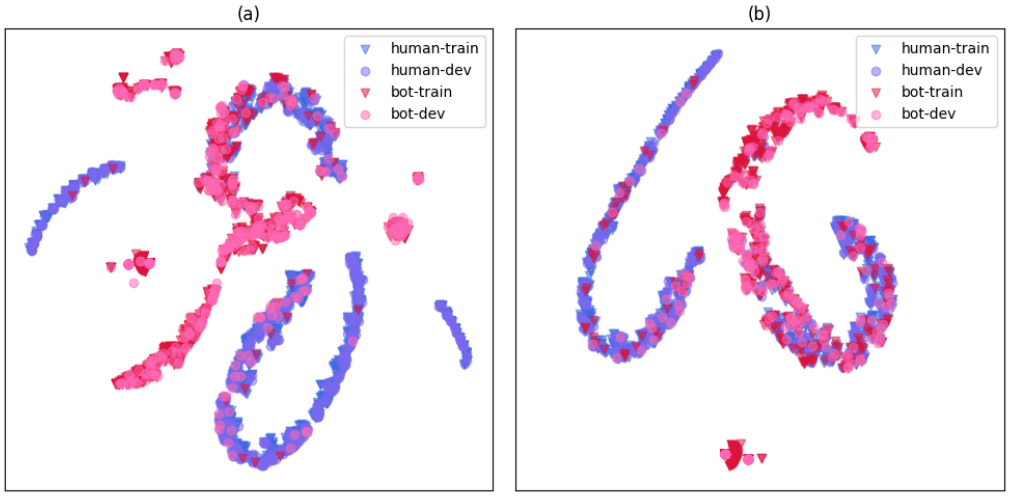

Figure 3.6: Bot t-SNE visualization. (a) English, (b) Spanish

testing the improvement with training data increase, as well as feature set generalization using raw fingerprint n-grams versus statistical diversity measures.

Experiment 1

In Experiment 1 we used character n-grams of user fingerprints described in Section 3.3. The n-gram lengths used are 2, 3 and 4. We can see that some classifiers have fairly similar results (Table 3.4, column E1). The best classifier is Random Forest for both languages. In this experiment we used the training subsets for English and Spanish separately.

Experiment 2

In Experiment 2 we used the diversity measures calculated on character n-grams of user fingerprint described in Section 3.4. The n-gram lengths used are 2, 3 and 4. The best classifier is Random Forest for both languages. In this experiment we used the training subsets for English and Spanish separately.

Experiment 3

In Experiment 3 (Table 3.5, column E3) we used the same features as in Experiment 1. The best classifier is the Gradient Boosting ensemble for both languages. In this

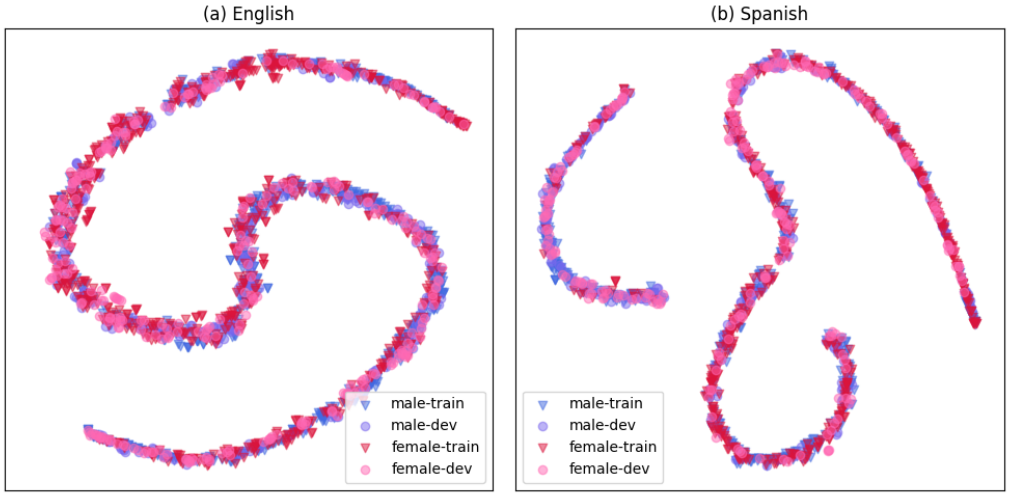

Figure 3.7: Gender t-SNE visualization. (a) English, (b) Spanish

Table 3.4: Bot classification. Results from testing on the development dataset. Per language training dataset. *not available due to memory restrictions.

		E1			E2		
Dataset	Classifier	Precision	Recall	F1	Precision	Recall	F1
English	**GB**	91.97	91.53	91.51	92.63	92.34	92.33
	SVM	91.74	91.61	91.61	92.53	92.42	92.41
	LR	88.40	87.50	87.43	92.61	92.42	92.41
	KNN	–*	–*	–*	92.84	92.58	92.57
	RF	92.84	92.18	**92.15**	92.93	92.66	**92.65**
Spanish	**GB**	86.66	86.63	86.63	84.29	83.91	83.87
	SVM	86.02	85.98	85.97	81.64	81.63	81.63
	LR	86.63	86.63	86.63	85.10	84.78	84.75
	KNN	–*	–*	–*	86.17	85.87	85.84
	RF	91.15	90.33	**90.28**	85.03	84.89	**84.88**

experiment we used the training subsets for English and Spanish combined. Because the features are language independent, we combined the training datasets into one, and tested it on both languages. The final model is the same for both languages.

Experiment 4

In Experiment 4 (Table 3.5, column E4) we used the same features as in Experiment 2. As in Experiment 3, we combined the training datasets into one, and tested it on

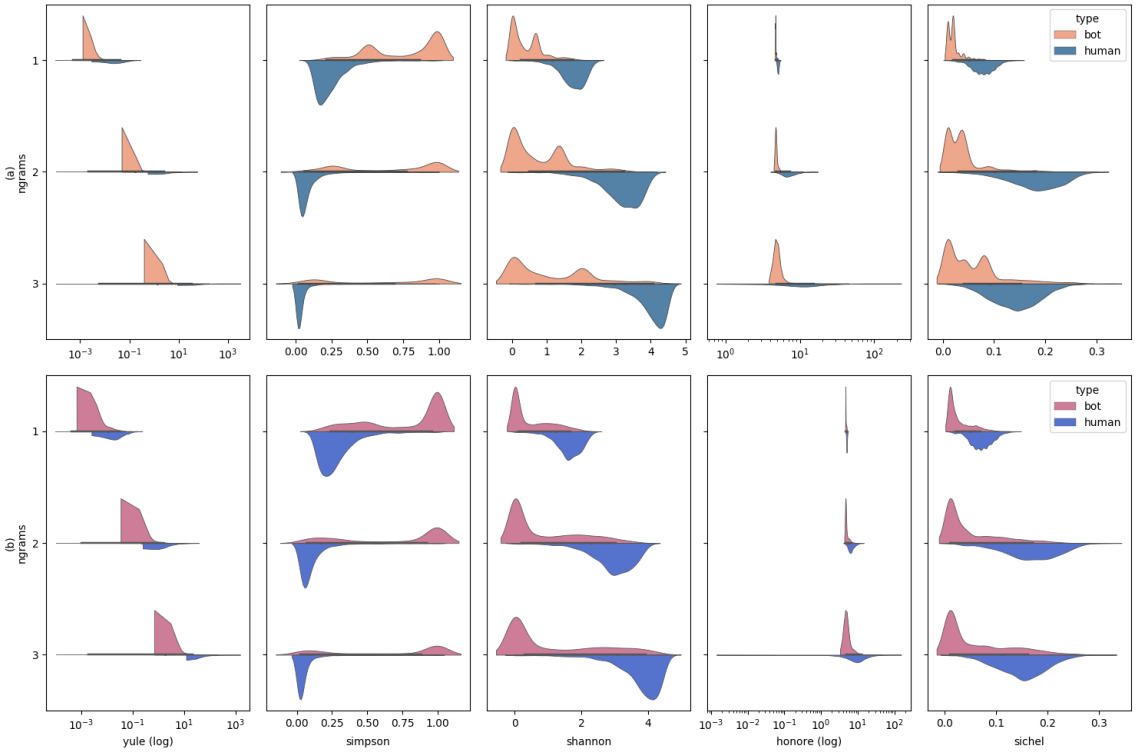

Figure 3.8: Diversity measures density per dataset, per user type. (a) English – top row, (b) Spanish – bottom row.

both languages. The best classifier for English is Gradient Boosting ensemble and K Nearest Neighbours for Spanish.

Although a better performance was obtained on separately trained models for the two languages (Random Forest, Table 3.4) with raw features, we opted for Gradient Boosting ensemble which was trained on combined dataset (the Spanish portion slightly dropped in performance). The best classifier from Experiment 4 was retrained on the combined training and development sets for the official ranking.

3.8.3 Gender Identification (Experiment 5)

For the gender identification sub-task we used the same set of classifiers as for bot detection. The results in Table 3.6 show that the Gradient Boosting classifier performed the best for both languages. This subtask was language dependent, that is, each language had its own model. This is a different case from the bot identification subtask, where we developed a language independent model.

Table 3.5: Bot classification. Results from testing on the development dataset. Combined training dataset. †used as final classifier (E4 for official ranking). *not available due to memory restrictions.

		E3			E4		
Dataset	Classifier	Precision	Recall	F1	Precision	Recall	F1
English	**GB**†	92.52	92.42	**92.41**	93.30	93.06	**93.05**
	SVM	90.94	90.81	90.80	91.99	91.77	91.76
	LR	91.21	91.13	91.12	92.14	92.02	92.01
	KNN	−*	−*	−*	92.56	92.42	92.41
	RF	91.89	91.53	91.51	92.56	92.42	92.41
Spanish	**GB**†	88.96	88.80	**88.79**	85.12	84.24	84.14
	SVM	85.88	85.87	85.87	84.90	84.35	84.29
	LR	84.78	84.78	84.78	84.73	84.46	84.43
	KNN	−*	−*	−*	85.86	85.43	**85.39**
	RF	87.64	86.96	86.90	84.98	84.35	84.28

Table 3.6: Gender classification. Results from testing on the development dataset. †,‡used as final classifiers.

Dataset	Classifier	Precision	Recall	F1
English	**GB**†	81.67	81.29	**81.23**
	SVM	77.82	77.74	77.73
	LR	76.30	76.29	76.29
	KNN	60.54	60.48	60.43
	RF	79.26	79.19	79.18
Spanish	**GB**‡	70.62	70.00	**69.77**
	SVM	65.92	65.87	65.84
	LR	64.18	64.13	64.10
	KNN	58.51	58.48	58.45
	RF	65.68	65.43	65.30

3.8.4 Results on Test Data

The official results are shown in Table 3.7. Bot detection for English performed with similar results as in our experiments with the development set, while for Spanish it performed better. Similar improvement was obtained with the Spanish dataset for gender identification. The models for the final evaluation are trained on both, training and development sets.

Table 3.7: Final results on test dataset. Averaged per language.

Dataset	Bot	Gender
English	92.16	79.28
Spanish	89.56	74.94
Average	90.86	77.11

3.9 Conclusion and Future Work

In this chapter we conducted a set of experiments to find a simple, yet effective bot detection method on the Twitter social media platform. We show that it is possible to detect automated users by using a fingerprint of user behaviour and a set of statistical measures that describe different aspects of that behaviour. The measures describe "constancy" or "diversity" of the pattern. The hypothesis was that the automated users show lower diversity, and tend to use a smaller set of types of messages over an extended period of time. Through visual analysis, discussion and classification results we showed that assumption did hold under our experimental setup. Additionally, we conducted the experiments on two different datasets used earlier in the research community to examine if the time-span of user behaviour has an impact on the ability to detect bots. We showed that the dataset which was collected focusing around specific topics and shorter time-spans generally performed better than the dataset where these user patterns diverge. The strength of this approach lies in the fact that it is language independent.

The main drawback of our approach is that a classifier needs at least 20 tweets per user to generate a fingerprint. The number 20 was empirically picked based on observations during the experiments (keeping the fingerprints shorter than 20 worsened the results of all classifiers). Another point is that social bots evolve over time, and they tend to be more difficult to identify with established machine learning methods. Bot creators can take advantage of the present ML knowledge and enhance their algorithms, so they stay undetected longer.

And last, to further verify our results and perform more thorough study, we plan to apply our approach to more datasets. Additionally, we plan to develop an unsupervised method for bot detection on the same set of features using clustering techniques.

Chapter 4

Topic Extraction Using the Centroid of Phrase Embeddings on Healthy Aging Survey Open-ended Answers

4.1 Introduction

Survey research is a very common approach when it comes to gaining insights into a research subject. For example, it is used in different domains, such as health and health services [39], marketing and consumer analysis [262, 276], but it originated in social sciences. Although the survey data is collected using a standardized form, Open-Ended (OE) questions can be part of it. Its primary role is to to clarify ambiguities and provide explanations and potentially identify opinions that researchers did not include in the standardized form [231, 217]. Another important point to mention is that OE questions expand the capability of the survey to capture spontaneous thoughts, sentiments and attitudes. This is useful in marketing research where companies can measure consumers' attitude towards their products.

Nonetheless, processing such questions requires great human effort. Because of the nature of OE questions, the standard approach in identifying the topics requires researchers to go through all the answers and label them manually. This may not be a challenge for smaller studies, but in the case of tens of thousands of samples the task can take a lot of resources to accomplish. If the data is labeled by multiple researchers, the process is prone to errors, which is usually measured with between-rater variance [279, 83]. An important challenge in automated processing of the OE answers is that the texts are relatively short. Extracting topics from short texts is difficult because most of the traditional methods rely on word co-occurrence, which assumes that the related words occur together relatively frequently, and this is not a reasonable assumption in the sparse data collections such as survey answers [114].

In this study our focus is on a survey from Canadian Longitudinal Study on Aging (CLSA) conducted on over 50,000 older adults living in the 10 Canadian provinces.

The survey responses were collected in two official languages of Canada: French and English. Demographic forecasts indicate that Canada's population is aging and the demographic structure will change dramatically over the next two decades. The numbers show that 25% of the population will be over 65 by 2036, almost double compared to 2009 [30]. The consequences of the demographic shift are among Canada's most pressing health and social policy issues. To put it into perspective, the total health and social care expenditures in Canada now exceed $300 billion with healthcare alone at approximately $211 billion, the largest expenditure item in provincial budgets [252]. Optimizing population health and wellness over the trajectory of aging — i.e. optimizing "healthy aging" — is therefore a major research and policy goal in Canada [53]. Therefore, we are analyzing the answers on the following OE question: "What do you think makes people live long and keep well?"

The aim of this study is to analyze open-ended survey responses by applying a combination of IR and unsupervised ML techniques to discover the potential differences among certain subgroups, including gender, age, and presence of health conditions. We describe an interesting solution in a form of framework for group profiling based on difference in opinions (that is, topics) and compare it with probabilistic topic modelling approaches. Our goal is to extract the topic-representative keyphrases that are more intuitive for topic labeling by the domain expert by introducing part-of-speech information, as well as semantic relatedness in a form of word embeddings.

4.2 Related Work

Automatic topic identification has a long history [16, 33], and it has been covered by a couple survey research papers over the course of the years [3, 8, 59, 127, 158]. In 1990, Deerwester *et al.* [62] identified deficiencies of term-matching retrieval. Their underlying assumption is that every document has some underlying latent semantic structure that is partially obscured by the randomness of word choice in queries for retrieval.

Domain of the topic identification refers to tasks of finding semantically meaningful topics from a document corpus. The base assumption says that there are hidden variables (topics) which describe the similarities between observable variables (that is, documents).

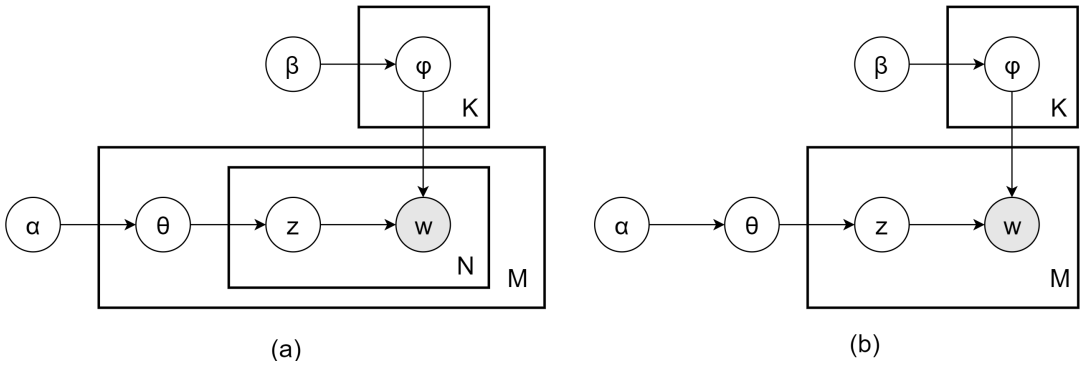

Figure 4.1: Traditional (a) LDA model and (b) DMM model in plate notation.

Some of the most influential representatives of topic modelling methods are probabilistic Latent Semantic Indexing (pLSA) [113] and its generalization – Latent Dirichlet Allocation (LDA) [26]. LDA has been around for awhile, and has been applied to different domains, such as short [114, 156] and long texts [27, 217], genetic data [218], and images [115]. The model is shown in Fig. 4.1 (a) and in a simple notation can be expressed as following:

1. Sample a topic proportion $\theta_i \sim \text{Dir}(\alpha)$, given $i \in \{1, \ldots, M\}$;
2. Sample a multinomial distribution over words $\varphi_k \sim \text{Dir}(\beta)$, given $k \in \{1, \ldots, K\}$;
3. For each of the word positions i, j ($i \in \{1, \ldots, M\}$, $j \in \{1, \ldots, N_i\}$):
   (a) Sample a topic $z_{i,j} \sim \text{Multinomial}(\theta_i)$;
   (b) Sample a word $w_{i,j} \sim \text{Multinomial}(\varphi_{z_{i,j}})$.

where $\text{Dir}(\alpha)$ and $\text{Dir}(\beta)$ are a *Dirichlet distribution* with a symmetric parameter α (typically is sparse and $\alpha < 1$) and β (typically is sparse), respectively.

However, an LDA model in its original setup has a few shortcomings, especially when the target documents are short, or there are too many topics. This paper focuses on the former. Dirichlet Mixture Model (DMM) in Fig. 4.1 (b) has been applied on short texts, but it comes with a disadvantage: one document can be assigned with one topic. This is not the case with our dataset, where a survey participant could talk about a mixture of topics.

Topic modelling has been used in various domains. Lately, significant attention is dedicated to modelling of short texts, due to OSN presence in all aspects of modern

society. The source of short texts are not limited to online microblogs, news feeds and forums. As already mentioned, OE questions in survey data, medical records are also characterized by being short and very often noisy. Pivovarov *et al.* [219] give an overview of the methods being used in the clinical domain. Some of the research [154, 163] is particularly focused on using topic modelling on healthcare data. In the next few sections we present some of the most influential and most recent work in topic modelling on short and noisy texts.

4.2.1 Topic Models in Short Texts

In the literature there are several studies on topic extraction from survey OE responses. The main characteristics of these texts are that they are short (usually between one and a few tens of words), not complete sentences, may or may not have punctuation, and prone to a degree of grammatical mistakes. Roberts *et al.* [231] proposed a Structural Topic Model (STM) for topic discovery in OE responses. The main difference between traditional LDA and STM is that they include covariates of interest into the prior distributions for document-topic proportions and topic-word distributions. With this setup the result is a model where each OE response is a mixture of topics with incorporated prior knowledge about topical variance. Thorough experiments on topic modelling on OE responses were performed by Pietsch *et al.* [217] using two then state-of-the-art algorithms (BTM, WNTM) [312, 326] and LDA as a baseline. They examine suitability of the automated algorithms to replace manual analysis and give some general recommendations for researchers and practitioners how to choose the right method for a given research task. They particularly chose the algorithms which are designed to address the issue of short documents. We conduct our comparative analysis with the same set of algorithms, hence the next few paragraphs are dedicated to description of the same.

Biterm Topic Model (BTM) [49, 312] is a model that does not use an external knowledge source to deal with the short documents or missing context as some other methods (LF-LDA). The main difference between BTM and LDA is that the input for it is not a set of documents D, but set of biterms B calculated on the corpus level. A biterm b represents a word pair that co-occurred in a specified short context window. Additionally, LDA uses the word co-occurrence pattern per document to

generate words while BTM generates biterms. One drawback of BTM model is that it uses the word co-occurrence frequency over the whole corpus, which gives advantage to the most frequent word pairs. Chen *et al.* [48] apply a simple modification by weighting word co-occurrence with PMI. Another drawback of BTM model is the fact that it generates one topic distribution for all documents which limits the model's expressiveness when the documents contain many topics. To address this, Zhu *et al.* [323] proposed GraphBTM which uses *Graph Convolutional Networks (GCNs)* to represent biterms as graphs. To overcome the data sparsity of LDA and the aforementioned drawback of BTM, they sample a fixed number of documents and merge them to a small corpus as a sample. Lu *et al.* [162] extended the BTM model by integrating Recurrent Neural Network (RNN) layer to model term semantic similarity, and used IDF to filter high frequency common words.

Word Network Topic Model (WNTM) [326] is a recent model that infers topic distributions for words instead of documents to avoid the disadvantage of LDA with short texts. The core of the algorithm is word co-occurrence network which is created by moving a sliding window of length S through each document. The network nodes are the vocabulary of the corpus and the edges represent the co-occurrences of each word pair weighted by the number of co-occurrences in the corpus. In another words, for each word w_v a pseudo-document d_p is created that consists of all words that co-occur with w_v, i.e. all words that are direct neighbours of w_v in the word network. The generated pseudo-documents are used as input in WNTM. However, WNTM can introduce unrelated word co-occurrence information, which hurts the performance and the coherence of the final topics. They [299] also proposed an improvement called Robust WNTM (RWNTM), which filters out unrelated word co-occurrences information.

Traditional LDA has a drawback of neglecting word order within documents, that is, documents are not treated as a sequence of words but rather as a BoW. There have been several attempts to address this problem, but most of the early attempts were computationally costly. Topic Keyword Model (TKM) [248] tries to address this in an efficient manner. They use the information of the topic assignments of the keyword neighbouring words, so they have an impact on the topic assignment to that particular keyword.

Some of the methods aim to improve traditional LDA topic modelling by transforming (concatenating) the original short documents into larger ones. This approach [179] was conducted in the experiments on three different Twitter datasets. A tweet pooling scheme based on hashtags was proposed and compared to a few earlier techniques [114, 191, 302] based on author, topic burstiness and temporal pooling. They showed that hashtag-based pooling outperforms other schemes in most of the cases and significantly outperforms LDA topic models on Twitter data. There are other variants of pseudo-document generation to improve short text topic modelling. Hajjem *et al.* [104] used IR technique to cluster similar tweets in larger pseudo-documents. The process consists of three steps. The first step is preliminary set generation (set length n), where they cluster tweets based on cosine similarity. The second step is aggregation of similar preliminary sets into pooled set representation (of length m, where $m < n$). This set is basically a set of pseudo-documents used for the next step. The final step is traditional LDA. They compared the results of different variants of the methodology with LDA trained on one document (whole dataset merged into one) and BTM. Another interesting approach [25] was proposed as a general framework for addressing the issues with short text topic modelling. They build the model using BTM, WNTM and LF-LDA by applying an expansion procedure on each document. Two variants were proposed: co-frequency expansion (CoFE) and distributed representation-based expansion (DREx).

To overcome the sparsity of short texts, Jin *et al.* [128] proposed Dual Latent Dirichlet Allocation (DLDA) model where they use transfer learning from auxiliary long text data to cluster the short texts. DLDA jointly learns two sets of topics on short and auxiliary texts and couples the topic parameters to deal with the possible inconsistencies between the datasets. Lin *et al.* [157] proposed a method they call Dual-sparse Topic Model that addresses both the sparsity in the topic mixtures and word usage. They use *"spike-and-slab"* prior to decouple the sparsity and smoothness of the document-topic and topic-word distributions. In this way, the model allows that an individual document can select a few focused topics and a topic can select focused terms. Lime *et al.* [156] conducted the experiments on a Twitter dataset and incorporated the additional information into their custom LDA model including authorship, hashtags and the user-follower network. They jointly model the text

and the social network and apply hierarchical Poisson-Dirichlet processes (PDP) for text modelling and a Gaussian process random function model for social network modelling.

Another common approach to address the problem of the sparsity of short texts is by incorporating external knowledge (similarity and relations between words). Xie *et al.* [311] proposed a Markov Random Field regularized Latent Dirichlet Allocation (MRF-LDA) model. Their model consists of a MRF on the latent topic layer of the standard LDA to enforce the words labeled as similar to fall into the same topic. Hence, the topic assignment of each word is not independent, but affected by the topic assignments of the correlated words. They compare it to two other methods which extend standard LDA: DF-LDA and Quad-LDA. DF-LDA [12] uses a Dirichlet Forest prior instead of Dirichlet prior over the topic-word multinomials to encode "Must-Links" and "Cannot-Links" between words. A "Must-Link" represents a primitive which means that two words have similar probability within any topic (either small or large, but similar). A "Cannot-Link" represents a primitive which means that two words cannot both have large probability within any topic. Quad-LDA [195] regularizes the topic-word distributions with a structured prior to incorporate word relations. MRF-LDA seemed to outperform both of the aforementioned methods.

Another extension of traditional LDA is Latent Feature LDA (LF-LDA) [201]. It addresses the sparsity of short texts by using pre-trained word vector representations (Word2Vec [183] and GloVe [214]). In LF-LDA, the generative process is similar to original LDA but differs in the way how words are generated from topics. In LDA, a word can only be drawn from the Dirichlet multinomial distribution ϕ that is trained on the target corpus, while LF-LDA additionally allows draw from the multinomial distribution based on word vector representation of words and topics. This means that LF-LDA incorporates semantic knowledge from external corpora. They also introduce additional hyperparameter λ which determines the probability of word sampling from external latent feature component. However, their implementation adds to the complexity of the model and the inference time grows with the number of documents, which makes it unsuitable for large datasets. Li *et al.* [155] proposed another similar extension to Dirichlet Multinomial Mixture model with *Generalized Pólya Urn* sampling method (GPU-DMM). They show that the inference is much

faster than in LF-LDA. DMM however, assumes that a document has only one topic, which inherently is a simpler model. Another similar approach with external knowledge was presented by Hu *et al.* [118], with the difference being that they employ a continuous vector space and can handle out-of-vocabulary words.

Model developed by Xie *et al.* [310] combines LDA and k-means clustering. They used variational inference in learning phase. Some of the earlier works [123, 226] proposed to guide topic modelling by setting a set of topic representative seed words to initialize the model. The main limitation of this approach is that the researcher has to know what topics to expect and which seed words are good representatives of the topics.

4.2.2 Multilingual Topic Models

A significant amount of literature has studied transferring the probabilistic topic modelling concept from monolingual to multilingual settings [296]. Bilingual LDA has been independently designed by several researchers [61, 122, 186, 203, 216, 221]. In one of the early works in the domain Zhang *et al.* [320] proposed an extension of standard pLSA to extract topics from cross-lingual datasets. They bridge the gap between different languages $(L_1, L_2, ...L_n)$ by introducing aligned dictionary. In this setting they define word distribution of a cross-lingual topic θ for language L_i as $p_i(w_i|\theta) = \frac{p(w_i|\theta)}{\sum_{w \in V_i} p(w|\theta)}$, where V_i is vocabulary of language L_i. These formulations are extension of the traditional maximum likelihood estimator to estimate parameters and discover cross-lingual topics. Another similar work from around the same time — JointLDA [122] addresses the issue in a similar way. Authors extended the standard LDA model with a bilingual dictionary to mine multilingual topics from an unaligned corpus (experiments conducted on English and Spanish). A more recent study by Vulic *et al.* [295] consider the topic modelling for multilingual datasets by training bilingual word embeddings. It is important to note that our approach is different in a sense that we are using pre-trained aligned word vectors due to the fact that the dataset presented in this study has a couple of limitations, such as size, length of documents and imbalance between the languages.

4.2.3 Neural Network Topic Models

ProdLDA [263] is a method based on *Autoencoded Variational Inference For Topic Model (AVITM)*. The model follows the traditional LDA algorithm in keeping the Dirichlet multinomial parameterisation and instead of a variational Bayes approximation of the posterior distribution for the inference phase it applies a Laplace approximation to allow gradient to backpropagate to the variational distribution. On the other hand, Miao *et al.* [181] employed models that directly parameterise the multinomial distribution with neural networks and jointly learn the model and variational parameters during the inference phase.

OnSeS [204] is a short text summarization method which is based on Word2Vec word representation and neural network model for summary generation. The proposed algorithm consists of three phases including clustering of texts using the k-means algorithm, ranking content of each cluster by building a graph-based ranking model using BM25 and generating main point of each cluster with the help of neural machine translation model on the top ranked sentence. Srivastava *et al.* [264] presented a topic modelling algorithm based on *Deep Boltzmann Machines (DBM)* which was at the time of publication state-of-the-art and outperformed LDA and other neural model called Neural Autoregressive Density Estimators (DocNADE) [151] on standard benchmark datasets (20 Newsgroups and Reuters RCV1-v2).

4.2.4 Vector Space Models

Vector Space Models (VSMs) are widely used in IR to represent documents in a matrix format, where each row is a term vector consisting of, in the simplest form, the frequency of all terms in the given document and each column is a document vector consisting of the frequency of documents the term appears in. However, the size of the vector depends on the vocabulary (all possible terms in the corpus) and the size of the corpus. Researchers tried to tackle this problem for many years [189, 258, 286] using different methods to "compress" vectors and retain the information they encode. In 2013, Mikolov *et al.* [182, 184] presented a new method (NN-based) for learning word vectors - Word2Vec and made a huge breakthrough. They demonstrated better performance on word similarity tasks and managed to train their model in a fraction

of the time required for pre-existing solutions. Soon after this publication Pennington *et al.* [214] gave a probabilistic solution with comparable performance, and they called it GloVe. A year later, Levy *et al.* [153] showed that neural network-based models are not superior to traditional probabilistic models when it comes to word representation. They presented a study on the effect of different hyperparameters in embedding algorithms which can be transferred to traditional methods. As a base, they used Positive Pointwise Mutual Information (PPMI) matrix and Shifted PPMI with Singular Value Decomposition (SVD) factorization. PPMI is constructed as $PPMI(w, c) = max(PMI(w, c), 0)$, while SPPMI is $SPPMI(w, c) = max(PMI(w, c) - logk, 0)$, where c is the context of word w and k is a hyperparameter representing a number of negative samples. Soleimani *et al.* [259] proposed using PMI with a fraction of negative samples, $PMI(w, c) = \{PMI(w, c), PMI(w, c) > \alpha; 0, otherwise$, where α is the fraction of negative samples. Their method demonstrated superior performance. Sajadi *et al.* [239] proposed an interesting method for inferring word embeddings based on Wikipedia concept graph and PageRank [206].

Advances in word embeddings paved the way to improvements in document vector space representations and this is a very active area of research. Kusner *et al.* [148] proposed a model for sentence representation based on Word Mover's Distance (WMD) which was inspired by *earth mover's distance* metric. Other recent methods include Word Mover's Embedding [308], Distance to Kernel-based similarity measure between documents [307], Universal Sentence Encoder [44] and Bidirectional Encoder Representations from Transformers (BERT) [67] by Google researchers.

4.2.5 "Off-the-Shelf" Topic Modelling Tools

The fact that most of the approaches to topic modelling are language independent and don't require any domain-specific knowledge, they are relatively easy to publish and apply on different datasets. "Off-the-shelf" topic modelling tools are often used as baselines for novel work. Tools that were available at the time of writing are listed in Table 4.1.

Table 4.1: List of available "off-the-shelf" tools for topic modelling.

TM tool	Description
`Blei LDA`[1]	Original author LDA implementation in C [27].
`LDA and HDP`[2]	Traditional LDA and Hierarchical Dirichlet Process (HDP) implementation in Java by Block [28].
`jLDADMM`[3]	Traditional LDA and DMM implementations in Java [200].
`Gensim LDA`[4]	LDA implementation in Gensim framework (Python).
`Mallet LDA`[5]	LDA implementation in Mallet framework (Java).
`STTM`[6]	Implementations for DMM, GPU-DMM, GPU-PDMM, LF-DMM, BTM, WNTM, PTM, SATM, ETM, LDA and LF-LDA [225].
`GraphBTM`[7]	Implementation for GraphBTM from the original paper [323].
`ProdLDA`[8]	Implementation for ProdLDA from the original paper [263].
`TKM`[9]	Implementation for TKM from the original paper [248].

4.3 Standardized Evaluation Metric

Evaluation of the unsupervised machine learning models such as topic modelling pose a challenge because usually there is no gold standard to compare to. Most topic modeling research show qualitative assessments of the inferred topics or simply assert that topics are semantically meaningful. Existing quantitative assessments use an external task, such as IR [301] or a classification problem. AlSumait *et al.* [11] explored the differences between topic-specific distributions over words and the corpus-wide distribution over words to identify overly-general topics. Aletras *et al.* [7] defined a topic coherence measure based on context vectors for every topic top word. A context vector of a word w represents a vector generated bu using word co-occurrence counts in context window of size of 5. Lau *et al.* [152] used two topic evaluation methods -

[1] `https://github.com/blei-lab/lda-c`
[2] `http://www.bradblock.com/tm-0.1.tar.gz`
[3] `https://github.com/datquocnguyen/jLDADMM`
[4] `https://radimrehurek.com/gensim/models/ldamodel.html`
[5] `http://mallet.cs.umass.edu/index.php`
[6] `https://github.com/qiang2100/STTM`
[7] `https://github.com/valdersoul/GraphBTM`
[8] `https://github.com/akashgit/autoencoding_vi_for_topic_models`
[9] `https://github.com/JohnTailor/tkm`

word intrusion and topic coherence. Word intrusion is calculated by identifying an intruder word among the top words of a topic. They structured the topic evaluation in two different tasks: word intrusion and observed coherence. For topic coherence they found that the UCI measure (which is defined below) performed better than NPMI (Normalized PMI). Some research [66, 233] used and assessed the developed metrics on a range of different corpora.

The state-of-the-art evaluation methods for topic coherence are the intrinsic measure *UMass* [187] and the extrinsic measure *UCI* [196, 197] which depends on external reference corpora. Umass is defined as in Eq. (4.1).

$$score_{UMass}(w_i, w_j) = \log \frac{D(w_i, w_j) + \epsilon}{D(w_i)} \tag{4.1}$$

where $D(w_i)$ is the count of documents containing the word w_i, $D(w_i, w_j)$ the count of documents containing both words w_i and w_j, and D the total number of documents in the corpus. This score measures how much, within the words used to describe a topic, a common word is on average a good predictor for a less common word. ϵ is added to avoid a logarithm of zero. Stevens *et al.* [266] found that UMass coherence performs better if parameter ϵ is chosen to be small instead of $\epsilon = 1$ as in the original publication. UCI is defined as in Eq. (4.2).

$$score_{UCI}(w_i, w_j) = \log \frac{p(w_i, w_j)}{p(w_i)p(w_j)} \tag{4.2}$$

where $p(w_i, w_j) = D_{ref}(w_i, w_j)/D_{ref}$ and $p(w_i) = D_{ref}(w_i)/D_{ref}$, D_{ref} is the total number of documents in the external reference corpus, $D_{ref}(w_i)$ is the count of documents in the reference corpus containing the word and $D_{ref}(w_i, w_j)$, the count of documents containing both words.

4.4 Dataset

CLSA is a study and national platform of adult development and aging individuals, each with unique experiences of their environments, communities, and health and social systems. The CLSA follows 50,000 Canadians between the ages of 45 and 85 years over a 20-year period. However, the data utilized in this thesis come from the study baseline, collected between 2010-2015. CLSA is designed as a research platform

with the aim to accelerate understanding of the complex interplay among the vast array of determinants of health, from gene-environment interactions, to lifestyles, social networks and transitions in retirement and wealth.

After applying the pre-processing tasks that are described below the number of responses in English is 41,496 and 9,296 in French. To get a better understanding of the data, we conduct a simple statistical analysis. In the English subset 24.60% of responses are in the length range 1-3, 33.41% in 4-6, 20.44% in 7-9, 9.95% in 10-12 and 11.60% longer than 12, relative to the total responses in English. In the French subset 39.37% of responses are in the length range 1-3, 38.02% in 4-6, 14.16% in 7-9 and 8.44% longer than 9 words. That means that more than a half of the responses are shorter than 7 words.

4.5 Methodology

Our approach consists of a number of steps towards building a set of phrase groups that represent meaningful topics. Unlike probabilistic topic modelling methods, the method relies on IR techniques and a ML unsupervised method – clustering. In our approach we use a spectral clustering algorithm. k-means and kernelized k-means (Gaussian kernel) gave similar results from the perspective of coherence scores (we used the top 20 words of each cluster for evaluation to make it comparable to Dirichlet-based topic models), but overall content of clusters seemed subtly better as a final result. The intuition behind our approach is that IR methods can facilitate and speed up researcher's learning about the data by introducing structure to the unstructured text documents. With the right data representation model, one can exploit the full power of other variables in the survey and get insights into possible correlations. We refer to this method as Graph-aided Topic Clustering (GTC). The experimental setup is illustrated in Fig. 4.2.

Pre-processing

We conducted a couple of pre-processing steps to decrease the noise in the dataset and to transform the data in such a way that it complies with the requirements of the methods for topic modelling. First, standard pre-processing techniques are performed, such as conversion to lowercase and the removal of numbers and punctuation [173].

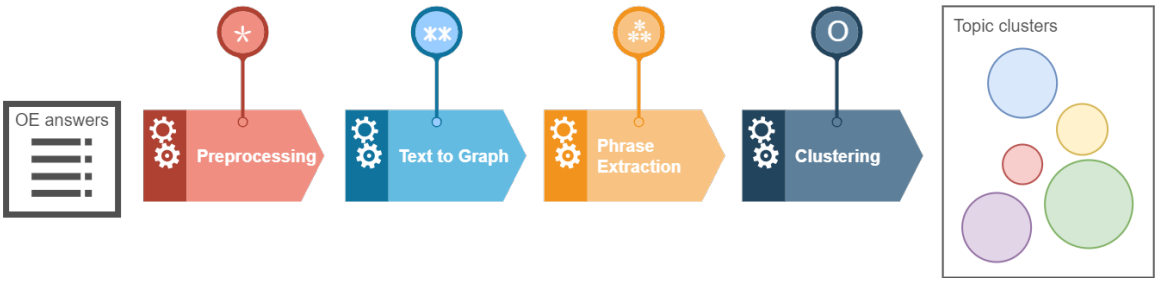

Figure 4.2: Experimental setup for phrase clustering task. *tokenization, lemmatization, PoS tagging, grammar correction; **neighbouring words are nodes connected with edges; ***Phrases extracted with PoS patterns; ^{O}spectral clustering with k parameter (number of clusters).

Using Stanford Log-linear Part-Of-Speech Tagger [281] for French and English we tokenized and tagged the entire corpus. For unsupervised spelling correction (unsupervised in a sense that we did not know which words are misspelled) contextual grammar correction [120] was used which relies on the external Google 1T N-grams corpus. To identify the candidates for spell correction, we scanned through words that have frequency less than 5 and checked if they exist in FastText aligned word vectors [130] used later in the process. If the words did not exist in the FastText word vectors, they are flagged for spell correction. In general, the dataset did not contain many misspellings, and the number of flagged words is less than 200. Lemmatization of the English language was performed using `Spacy`[10], and for the French language we used dictionary-based lemmatizer [238]. Although the dataset consists of English and French responses, we did not perform translation.

4.5.1 Graph Representation of Text

The dataset is represented by a directed graph $G = (V, E, C)$, where $V = w_1, w_2, ..., w_N$ is the set of nodes (i.e. vertices), each representing a word token. $E \subset \{(w_i, w_j) \mid w_i, w_j \in V\}$ is the set of edges between the vertices and it represents a direct neighbour connection between two word tokens. Each edge $e \in E$ is an ordered pair $e = (w_i, w_j)$ and is associated with a weight $we_{w_i,w_j} > 0$, which indicates the strength of the relation (frequency of the relation between two tokens in the dataset). Fig. 4.3 illustrates an example of the graph representation of two sentences.

[10] `https://spacy.io/`

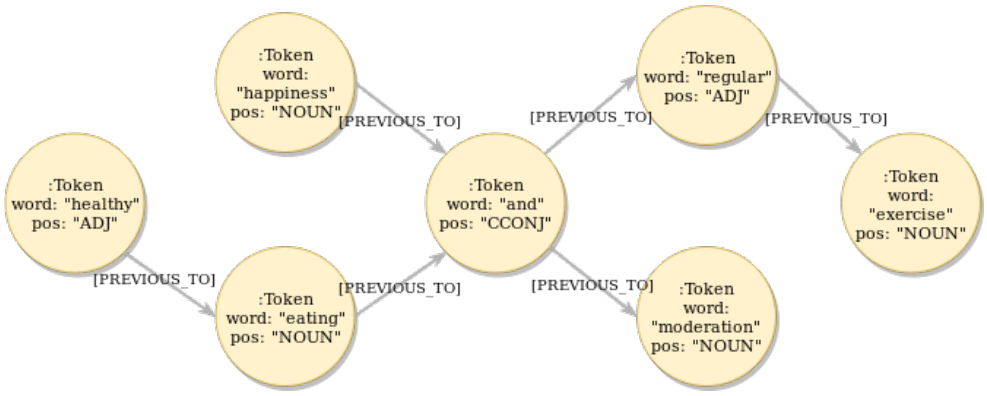

Figure 4.3: Example — graph representation of two answers "healthy eating and regular exercise" and "happiness and moderation."

Ganesan *et al.* [94] used similar concept to represent a set of unstructured and short texts and perform summarization. Our work is different in a few aspects. First, our goal is to extract characteristic keyphrases of 1-3 words in length, while they try to capture longer common sequences of words. Second, each token is enriched with additional information such as lemma form and part-of-speech tag which are used in keyword extraction process. Third, the whole word graph is extended with other fields from the survey, such as participant id and other variables of interest. This makes it possible to reconstruct each participant's response to its original form. We used the Neo4j graph database [194] because of its powerful SQL-like declarative graph query language called Cypher and its accompanying graph-specific features. Fig. 4.4 shows the conceptual model of a part of CLSA survey (the survey itself is far more complex including over 300 variables) that is relevant to this study.

4.5.2 Centroid of Phrase Word Embeddings

To extract the word phrases consisting of one, two or three words we used the tag information. The only words considered are verbs, nouns, adjectives and adverbs. The meaningful phrases are constructed by considering neighbouring words with the PoS tag rules that describe common phrase constructions in English and French:

- `^(DET)?(-?ADJ)*-?NOUN(-ADJ)?$`
- `^VERB-NOUN$`
- `^ADV-ADJ$`

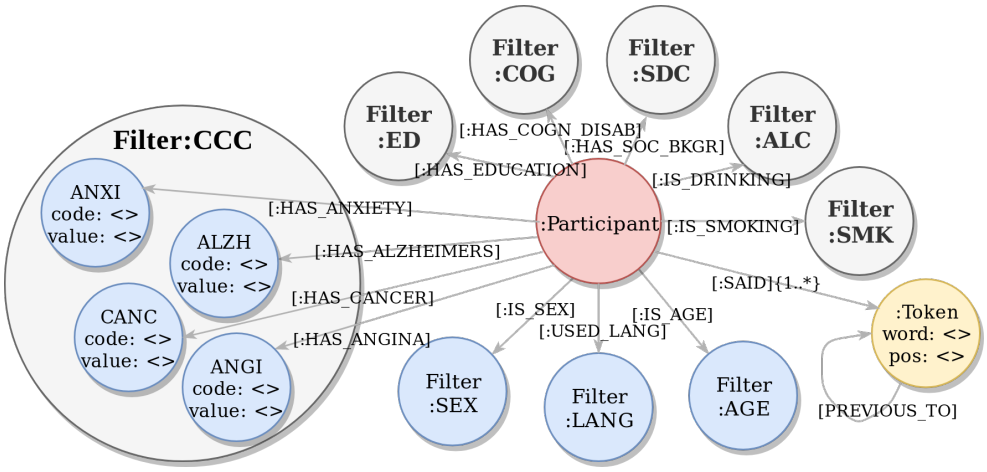

Figure 4.4: Conceptual graph model of the survey dataset. Filters: SDC — socio-demographic characteristics; ED — education; COG — cognitive disabilities; CCC — health conditions; ALC — alcohol consumption; SMK — smoking.

- `^VERB-ADV$`
- `^ADJ$`

Word2Vec [183], as mentioned earlier, is known as a computationally efficient predictive vector space model (VSM) for learning word embeddings from raw text. The FastText implementation is considered to be state-of-the-art (at the time of writing) for a couple of reasons. First, the models are trained using subword information, meaning that words are represented as a sequence of character n-grams. Second, the models for different languages can be aligned in the same vector space so the words from different languages with high semantic similarity are close to each other [130]. We opted for using FastText pre-trained aligned word vectors for English and French.

To represent a multi-word phrase, we calculate a centroid of word vectors. The centroid of a finite set of m ($m = 3$ in our case) word vectors $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_m \in \mathbb{R}^d$ (d is the vector dimension and in our case $d = 300$) is given as follows:

$$\mathbf{p_C} = \frac{\sum_{i=1..m} \mathbf{w}_i}{m} \tag{4.3}$$

Note that this is a very simple representation and there is significant work done in document and sentence vector representations [13]. Our motivation to use centroids stems from the fact that the phrases are very short and the neighbouring words

are likely to be semantically close. However, different phrase representations will be investigated in follow-up work.

4.5.3 Spectral Clustering

The extracted phrases represented as the phrase vectors are clustered using a spectral clustering algorithm. The spectral clustering algorithm is essentially a modification to k-means clustering algorithm with a few extra pre-steps. Given a set of n vectors (phrases) $\mathbf{P} = \{\mathbf{p}_{C1}, \mathbf{p}_{C2}, \ldots, \mathbf{p}_{Cn} \in \mathbb{R}^d\}$, the objective of spectral clustering is to divide these vectors into k clusters. The steps of the algorithm for spectral clustering are:

- Construct an affinity matrix A, consisting of pairwise similarities a_{ij}. The similarity measure method used to calculate a_{ij} in this paper is Gaussian kernel function for constructing the similarity $a_{ij} = exp(-\gamma\|\mathbf{p}_{Ci} - \mathbf{p}_{Cj}\|^2)$, where $\gamma(= \sigma^2)$ is a specified scaling parameter used for determining the size of neighbourhoods.

- Compute the normalized Laplacian matrix L based on the affinity matrix A as $L = D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$, where D is an $n \times n$ diagonal matrix with $d_i = \sum_{j=1}^{n} a_{ij}$ on the diagonal.

- Compute the k largest eigenvectors of the normalized Laplacian matrix L, and form the matrix $V = (v_{ij})_{n\times k}$ using these eigenvectors as its columns.

- Form the matrix $U = (u_{ij})_{n\times k}$ by normalizing the rows of V, such that $u_{ij} = v_{ij}/\sqrt{\sum_j v_{ij}^2}$.

- Each row of U represents a new vector for a phrase in $\mathbb{R}^k$ space. Then cluster the vectors using the k-means method.

- Assign each phrase $\mathbf{p}_{Ci}$ to a given cluster c if the corresponding row i in U is assigned to this cluster.

4.5.4 Hyperparameter settings

For the experiments on BTM, WNTM and LDA we used the implementations by Qiang *et al.* [225]. The reason for choosing hyperparameters values α, β and γ as

Table 4.2: Hyperparameters for the models used.

Method	# of topics	# of models	hyperparameters
LDA	$k \in \{2, 4, .., 50\}$	20	$\alpha = 0.05, \beta = 0.01$
BTM	$k \in \{2, 4, .., 50\}$	20	$\alpha = 0.05, \beta = 0.01$
WNTM	$k \in \{2, 4, .., 50\}$	20	$\alpha = 0.05, \beta = 0.01$
GTC	$k \in \{2, 4, .., 50\}$	2	$\gamma = \{0.1, 1.0\}, \texttt{kernel} = rbf$

shown in Table 4.2 is simply because they are recommended settings for short texts in the studies [201, 217, 312, 326] that proposed the algorithms.

4.6 Results

4.6.1 Quantitative evaluation

Fig. 4.5 gives an overview of the coherence scores (UMass top and UCI bottom row) produced for the different methods. The topic is considered more coherent if the score is higher. The UMass coherence score, as mentioned earlier, is calculated on the corpus itself. It indicates that coherence slowly decreases with the number of topics. It also shows a significantly lower value for the GTC approach. The reason for this is that the responses (documents) are very short and the number of topically related terms within a response is low (1-3 related terms). Hence, the point-wise mutual information statistic is unable to pick up semantically related terms from different documents because they rarely occur in the same context. The coherence measure performed on the reference external corpus (Wikipedia with longer documents and more samples) demonstrates almost opposite results. GTC shows better coherence scores for French and English ($k > 20$). The entire French Wikipedia (around 2.2 million documents) and the entire simple English Wikipedia (around 200 thousand documents) were used as reference corpora. Due to the volume of the standard edition of English Wikipedia (5.9 million articles at the time of writing) we were unable to use it as a reference, which may have been reflected in the results.

4.6.2 Qualitative Evaluation

We explore the quality of the topics based on the opinions of three domain experts. We used topics generated for the setting where $k = 20$. The reason for choosing this

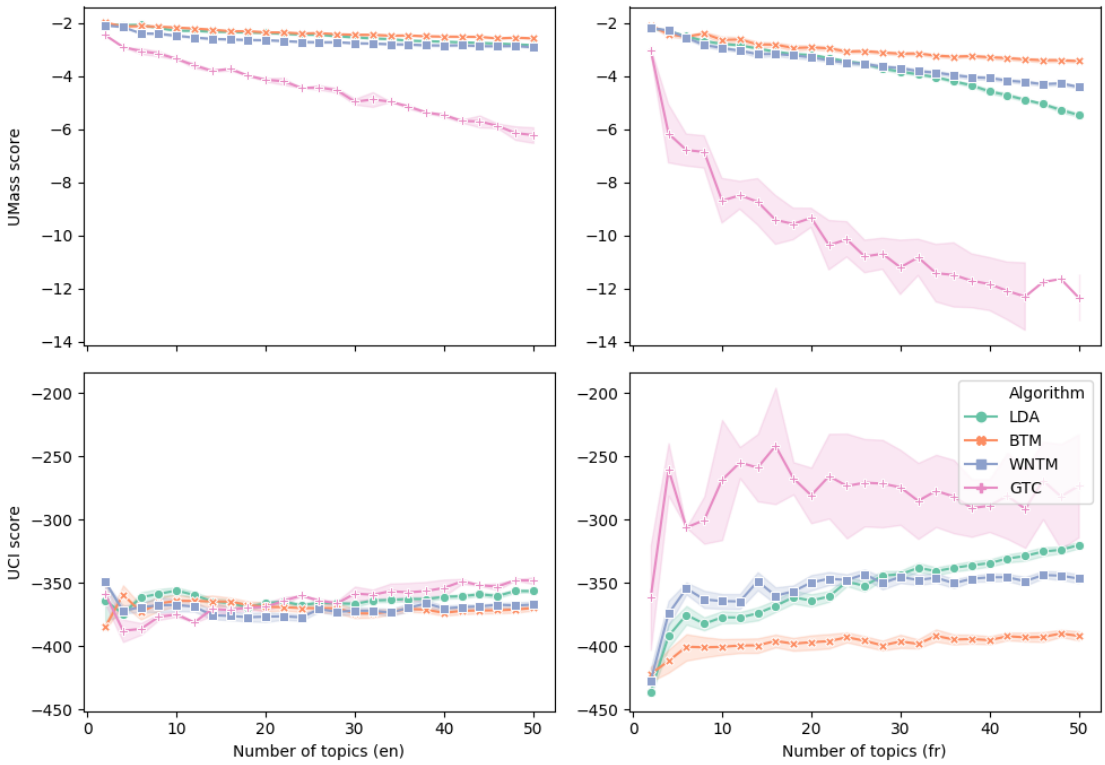

Figure 4.5: *UMass* (top row) and *UCI* (bottom row) coherence measures calculated and averaged over different models for top 10 representative terms for English (left column) and French (right column) subsets.

number is based on an empirical assumption about the number of topics derived from a prior simple analysis of the data. The coherence scores did not provide a definitive choice in terms of the number of topics, but 20 seemed like a good choice where the coherence score for English becomes better than the rest of the methods.

Age Groups

Pennebaker *et al.* [213] explored the correlation of use of words and age. They found that the older the individuals are, they use more positive and fewer negative affect words, among the other findings. Inspired by their experiments, we examine the differences between age groups in our dataset. The experiment is set up as a set of binary classification problems. The classes are: 1 (45-54 age range), 2 (55-64 age range), 3 (65-74 age range) and 4 (75+ age). The classification is applied pairwise with all possible age group combinations. Fig. 4.6 shows logistic regression results

Table 4.3: Top 10 terms and coherence scores for two example topics per method for English subset, where $k = 20$. (a) best topic according to UMass score, (b) best topic according to UCI score.

Method	UMass	UCI	Terms
LDA (a)	-1.8966	-326.8925	exercise good social family diet friend healthy relationship life activity
LDA (b)	-2.5279	-171.6791	eat exercise properly right healthy active n't eating food drink
BTM (a)	-1.9055	-276.5447	exercise activity social active diet physical mental mind healthy good
BTM (b)	-2.5599	-198.8898	exercise eat food good diet vegetable healthy not n't fruit
WNTM (a)	-2.2245	-182.6595	positive attitude life outlook mental good n't people happy not
WNTM (b)	-2.5611	-234.1146	good healthy active prop regular positive social balanced attitude activity
GTC (a)	-2.8286	-357.9075	active activity important interest physical physically mind mentally interested mental
GTC (b)	-3.9109	-254.3686	positive attitude moderation outlook good humour fun laugh humor mental

with 10-fold validation on each pair. An interesting observation is that with the bigger age gap the classification accuracy tends to increase and the trends are similar in both languages. Please note that the features for the classification consist of lemmas which are filtered based on the following PoS tags: nouns, adjectives, adverbs and verbs. The classification results and differences would be likely higher if we included the filtered words which is out of the scope of this paper.

The most notable trend in Fig. 4.7 is that most of the topics show ordered gradual increase/decrease in a topic involvement per group. The most notable difference is for the first age group which use phrases from topic 0 cluster and topic 19 cluster more than other groups. Topic 0 cluster contains words about exercise and topic 19 is about healthy eating and diet.

Gender

To examine the differences between genders in the dataset the experiment is set up as a binary classification problem. The classes are: F (women) and M (men). Using

Table 4.4: Top 10 terms and coherence scores for two example topics per method for French subset, where $k = 20$. (a) best topic according to UMass score, (b) best topic according to UCI score.

Method	UMass	UCI	Terms
LDA (a)	-2.0289	-375.1233	physique vie social activite alimentation exercice bien bon mental travail
LDA (b)	-2.5971	-93.5049	physique activite alimentation bon exercice nutrition mental stress sain activites
BTM (a)	-2.0253	-280.7812	actif pas physiquement bien bon alimentation exercice vie sante stress
BTM (b)	-3.2007	-182.1551	pas problemes regulier vis mental trop difference physique alimentation vie
WNTM (a)	-2.1108	-320.6495	plus possible vie bon alimentation pas medecin moins stress exercice
WNTM (b)	-2.9369	-104.4695	soin sante gens plus bien mental pas exercice personne c'est
GTC (a)	-2.7957	-378.3365	alimentation bon nourriture nutrition sain physique exercice gestion genetique activite
GTC (b)	-8.1423	47.5306	activite physique actif genetique activites activities gene excess genes hygiene

logistic regression and the same set of features as for the age groups we show that there is a difference between men's and women's responses. Fig. 4.8 illustrates the results on 10-fold cross validation.

Fig. 4.9 shows the differences in topics between genders. The most notable differences are topics 0, 2, 3, 4 and 5. Topic 0 cluster has terms mostly about exercise. Male participants use words from this cluster more than females. Clusters 2, 3 and 4 contain word related to family, children and relationships. Female participants tend to talk about these topics slightly more than males. The other clusters seem more or less balanced.

Pre-existing Conditions

In this section we examine the topical differences in participants that reported health conditions. On the conceptual graph (Fig. 4.4) the filter is referred as "CCC". Similar classification experiments were conducted on subsets of participants who reported

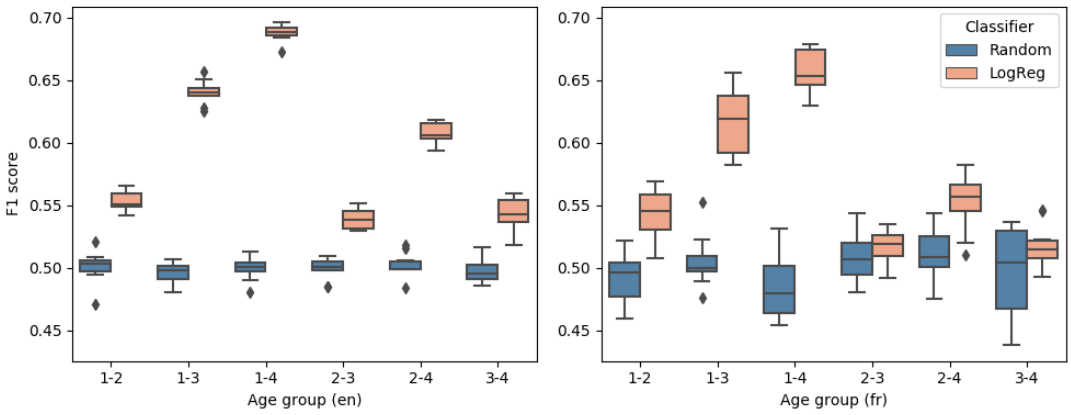

Figure 4.6: Pairwise classification with 10-fold validation between age groups for English (left) and French (right) subsets.

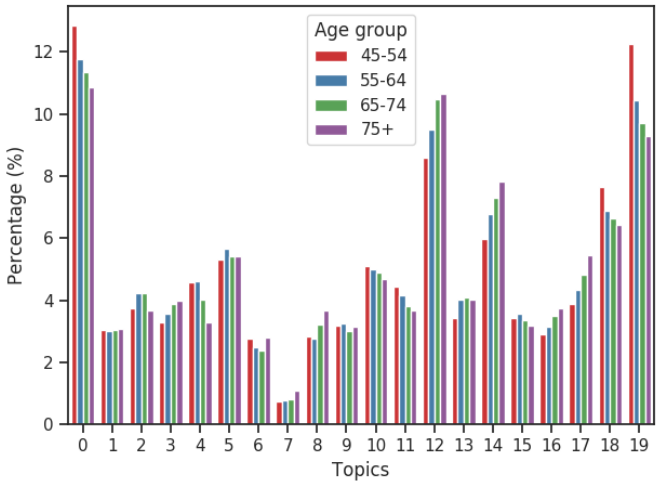

Figure 4.7: Difference in topics among age groups.

anxiety versus who did not, cancer versus who did not and Alzheimer's disease versus participants who did not. However, there was no significant difference between the groups and logistic regression classifier did not perform better than random. Although the difference was not detected in the classification experiments, the topic modelling methodology can help in discovering the differences on a semantic level. Fig. 4.10 and Fig. 4.11 show the topical distribution for three setups: anxiety-no anxiety, cancer-no cancer and Alzheimer's-no Alzheimer's.

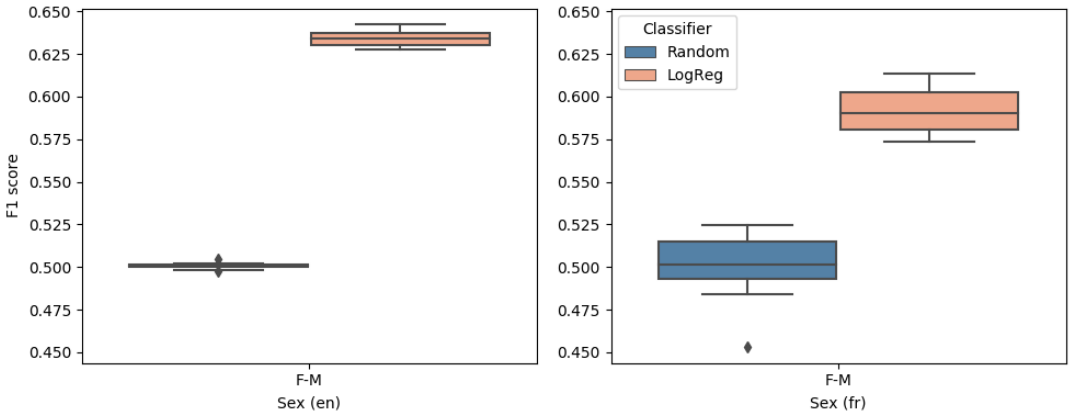

Figure 4.8: Classification with 10-fold validation between genders for English (left) and French (right) subsets.

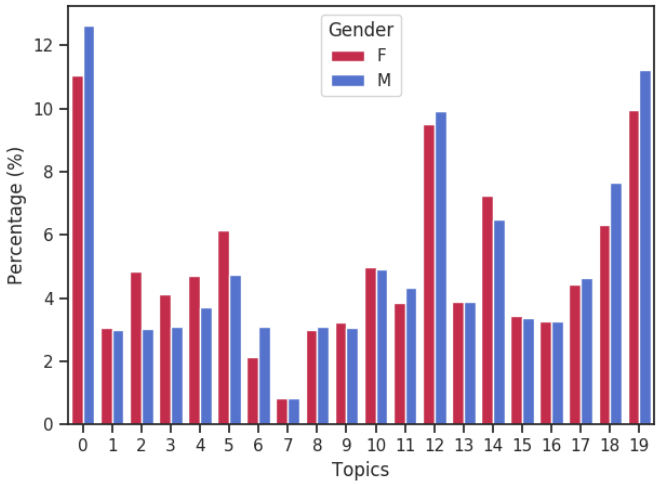

Figure 4.9: Difference in topics between genders.

4.7 Conclusion and Future Work

In summary, the current work has demonstrated an alternative method for topic extraction from OE responses. We compared the method with probabilistic approaches for short texts: BTM and WNTM, and LDA as a baseline. The results are compared based on *Umass* and *UCI* coherence measures which are two common unsupervised evaluation approaches. The observation is that these two measures, although based on the same idea (point-wise mutual information), show different results on the dataset. The main difference is that the former is intrinsic (based on the statistics of the

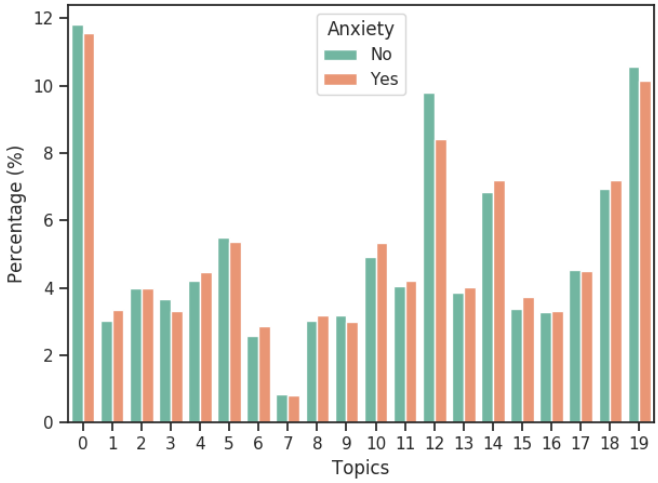

Figure 4.10: Difference in topics in setup anxiety-no anxiety.

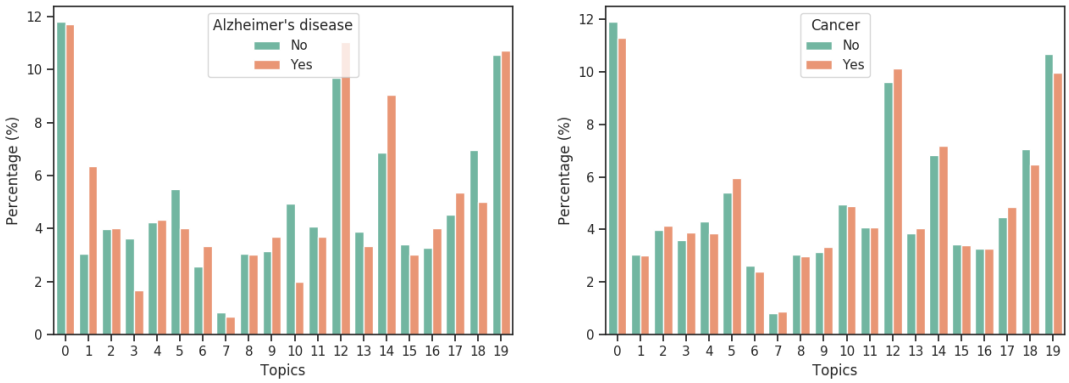

Figure 4.11: Difference in topics in setups: Alzheimer's-no Alzheimer's (left), cancer-no cancer (right).

dataset) and the latter is extrinsic (based on the statistics of the larger external corpus). We show and discuss why, in this case study, the extrinsic measure is more suitable to measure topic coherence. Additionally, we explore topical distributions with different grouping setups and discover some interesting insights about the data.

In this chapter we found that:

- IR and clustering-based method can be used for topic modelling in short open-ended survey answers;
- although not perfect, it is a more suitable method for creating user to topic mappings;

- intrinsic and extrinsic measures for short texts give the opposite results, which can be used to interpret to which degree we can use semantic information from short text corpora, and if possible;

- the extrinsic coherence measure is more suitable for short texts where it is hard to draw semantic information due to the sparseness.

Nevertheless, there are a couple of drawbacks of this approach that are important to mention. First, it is not suitable for online topic modelling as it depends on clustering and PoS tagging which are too slow for real-time settings, at least with the tools that we used in this study. However, the surveys are closed sets that are primarily focused on exploratory analyses and the prompt performance time is not a requirement. Second, the quality of the results largely depends on the quality of the pre-trained word vectors. To put it into perspective, for domain-specific datasets this can pose a challenge in a sense that the word vectors may not have good coverage for domain-specific terms.

For future work, we plan to conduct manual topic labeling and evaluate our methods in the usual way found in papers on topic models. Researchers, along with coherence scores, report classification accuracy on topic classification tasks. Next, BERT [67] can be used for phrase representation instead of simple phrase centroid distance, given that BERT has demonstrated better performance than GloVe and FastText on benchmark datasets and particularly works better for contextual word representation. This can be useful in cases when word meaning highly depends on surrounding words. For example, in French word "*hygiéne*" in general, means "*hygiene*", but in the context "*hygiéne* et sécurité" means "*health* and safety", or in "*hygiéne de vie*" is a compound phrase that means "*lifestyle*". And last, we plan to apply *Bernoulli Mixture Model* (BMM) to the generated binary matrix $(participants) \times (topics)$ and analyze "soft" clusters in relation to other survey variables.

Chapter 5

Conclusion

In this thesis, we tackled some problems related to author profiling based on short and noisy textual data. The problems are deconstructed and expressed in three projects where our aim was to explore a set of questions specific to the explored domains.

In the first project we looked into a few special cases of language identification (LID) on textual data. LID for long texts is considered to be a solved problem. However, traditional methods seem to fall apart when the texts of interest are short, noisy, or the languages are very similar. In the first part, we use CNG approach to quantify language differences. Using the dataset on 44 (40) European languages from Gamallo *et al.* we compare our results and show that CNG is suitable for modelling language distances. In the second part, we used seven different datasets gathered from evaluation labs at prominent conferences over the course of the years (2014–2019). Communal characteristics of these datasets are noise, brevity and sparsity. Most of the datasets are balanced. One of them is unbalanced (TweetLID), two datasets are dealing with transcribed utterances (GDI 2018 and 2019), four are focused on fine-grained language variant identification (both GDI, DFS and MADAR), and two are of a challenging size (both DSLCC). We test different global feature weighting methods, and propose a new one based on CNG distance. To our knowledge, there is no comprehensive study on impact of weighting techniques on language identification tasks. We show that *idf*, *BM25*, *mutual info* and *cng* can significantly improve the performance.

In the second project we explored automated account detection on social media. Bot and fake news detection are very active areas of research. Processing large amounts of data on social media and detecting anomalous behaviours is computationally challenging. With that consideration in mind, we explore different statistical diversity measures to characterize online user behaviour. The behaviour in this context means what types of messages an author uses on a social platform in a certain

period of time (mentions — interactions with other users, hashtags — interactions with topics, and similar). Our hypothesis is that automated accounts have less diverse interactions than genuine users. On two out of three datasets, we show that it is possible to distinguish automated bots with over 90% of accuracy. One of the datasets comes from PAN Author Profiling shared task, where we were placed 13th out of 50 publicly visible submissions. It is important to stress out that our method was computationally the fastest, due to using only 6 features. Our method is also language independent. As a part of future possible extensions, we can use our approach as a preliminary step for collecting suspicious accounts, and then use fine-tuned classifiers to decide if the bot is truly malicious.

Finally, the third project focuses on an unsupervised machine learning problem. Topic modelling on short and noisy texts has been in the focus of domain research for many years, mainly due to the popularity of OSNs. However, this kind of texts is not only common for microblogs. The use case study in this thesis focuses on mining open-ended survey answers from Canadian Longitudinal Study on Aging (CLSA) available in English and French language. The samples (answers) are brief and noisy which makes standard LDA model not so useful in discovering latent topics. We show that clustering-based methods with transfer learning from external knowledge are a better alternative to LDA-based topic models. In this case we also show that, between the two standard topic coherence measures used frequently in literature to report on the quality of topics, give opposite results. Our conclusion is that the intrinsic measure, which relies only on word pair distribution of the dataset is not capable of capturing semantically similar words, because they usually don't appear in the same sample, given that most of the answers are laconic. On the other hand, the extrinsic measure is driven by semantics of external knowledge base, and hence it is capable of assigning high score to semantically related words.

5.1 Future Work

In the LID task, we considered only one aspect of building a powerful classification model. We focused on identifying the best global weighting schemes, and did not experiment on the architecture improvements, nor hyperparameter tuning. Nowadays, it is very common to use complex ensemble methods. In the language distance

experiments, we have shown that some languages are more similar then others. This characteristic has a big impact on a model performance. Based on the thorough data analysis we can identify the classes that are harder to distinguish and build a model that is tuned particularly for those classes. We have seen that the most of datasets in the LID feature weighting task have classes that are hard to separate. One approach is to build a hierarchical model of classifiers: the root classifier distinguishes languages groups, and leaf classifiers distinguish specific language varieties/dialects [141]. Some languages have very limited digital resources, and the training sets are sparse, small and unbalanced. The approach for underresourced languages can involve transfer learning. In recent years, NLP domain had a number of breakthroughs (Google's BERT language model). Harnessing the power of pre-trained language models can introduce knowledge that is otherwise not available in our limited training set. Neural Networks, especially LSTMs, gained a lot of traction when in comes to language modelling. Although we did some preliminary experiments with NNs and LSTMs (not reported in this thesis), our findings were that the datasets are likely too small to build neural models that are on par with the probabilistic models. To support this claim, more thorough study should be conducted as a part of the future work.

In the bot identification task, we considered a simple method based on statistical diversity measures. To further verify our results and perform more thorough study, we plan to apply our approach to more datasets. Additionally, we plan to develop an unsupervised method for bot detection on the same set of features using clustering techniques. As the social bots are constantly evolving and exhibiting human-like behaviour, single simple classifier is likely to drop in performance, as time passes. There is no ideal proposition to address this issue, as bot creators can constantly use the knowledge of the research and improve on their bot models. Nevertheless, we plan to work on a complex ensemble method, that incorporates different aspects of user online behaviour, including topic interest, temporal behaviour, social network, likelihood to interact with fake news posts, etc.

In the topic modelling task on health data, we proposed a set of steps to extract topics and create mapping participant-to-topic. There are a couple of aspect where we can extend and improve our approach. First,we plan to conduct manual topic

labeling and evaluate our methods in the usual way found in studies on topic models. Researchers, along with coherence scores, report classification accuracy on topic classification tasks. Although the labelling can be an expensive and tedious task, the benefits of having a gold standard generated by domain experts are great. The evaluations are more descriptive and explainable. Next, language models, such as BERT [67] can be used for phrase representation instead of a simple phrase centroid distance, given that BERT has demonstrated better performance than GloVe and FastText on benchmark datasets and particularly works better for contextual word representation. This can be useful in cases when word meaning highly depends on the context. For example, in French word "*hygiéne*" in general, means "*hygiene*", but in the context "*hygiéne* et sécurité" means "*health* and safety", or in "*hygiéne de vie*" is a compound phrase that means "*lifestyle*". And last, we plan to apply *Bernoulli Mixture Model* (BMM) to the generated binary matrix (*participants*)×(*topics*) and analyze "soft" clusters in relation to other survey variables. Last, some of the drawbacks of our approach is that in consists of a few computationally expensive steps, which makes it unfit for any kind of on-demand analysis (as a Web service). To address this, we plan to experiment to alternative approaches which do not involve PoS tagging and spectral clustering.

References

[1] Norah Abokhodair, Daisy Yoo, and David W McDonald. Dissecting a social botnet: Growth, content and influence in Twitter. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, CSCW '15, pages 839–851, New York, NY, USA, 2015. ACM.

[2] Tony Abou-Assaleh, Nick Cercone, Vlado Kešelj, and Ray Sweidan. Detection of new malicious code using n-grams signatures. In *2nd Annual Conference on Privacy, Security and Trust (PST)*, pages 13–15, 2004.

[3] Charu C Aggarwal and Cheng Xiang Zhai. A survey of text clustering algorithms. In *Mining text data*, pages 77–128. Springer, 2012.

[4] Faraz Ahmed and Muhammad Abulaish. A generic statistical approach for spam detection in online social networks. *Computer Communications*, 36(10):1120–1129, 2013.

[5] Mohamed Al-Badrashiny, Heba Elfardy, and Mona Diab. AIDA2: A hybrid approach for token and sentence level dialect identification in Arabic. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 42–51, Beijing, China, July 2015. Association for Computational Linguistics (ACL).

[6] Abdulrahman Alarifi, Mansour Alsaleh, and AbdulMalik Al-Salman. Twitter Turing test: Identifying social machines. *Information Sciences*, 372(C):332–346, December 2016.

[7] Nikolaos Aletras and Mark Stevenson. Evaluating topic coherence using distributional semantics. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013) – Long Papers*, pages 13–22, Potsdam, Germany, March 2013. Association for Computational Linguistics (ACL).

[8] Rubayyi Alghamdi and Khalid Alfalqi. A survey of topic modeling in text mining. *A Survey of Topic Modeling in Text Mining (IJACSA)*, 6(1), 2015.

[9] Omar Ali, Ilias Flaounas, Tijl De Bie, Nick Mosdell, Justin Lewis, and Nello Cristianini. Automating news content analysis: An application to gender bias and readability. In *Proceedings of the First Workshop on Applications of Pattern Analysis*, pages 36–43, 2010.

[10] Jalal S. Alowibdi, Ugo A. Buy, and Philip Yu. Language independent gender classification on Twitter. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '13, pages 739–743, New York, NY, USA, 2013. ACM.

[11] Loulwah AlSumait, Daniel Barbará, James Gentle, and Carlotta Domeniconi. Topic significance ranking of LDA generative models. In *Machine Learning and Knowledge Discovery in Databases*, pages 67–82, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[12] David Andrzejewski, Xiaojin Zhu, and Mark Craven. Incorporating domain knowledge into topic modeling via Dirichlet forest priors. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 25–32, New York, NY, USA, 2009. ACM.

[13] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. A simple but tough-to-beat baseline for sentence embeddings. In *ICLR*, 2017.

[14] Ehsaneddin Asgari and Mohammad R. K. Mofrad. Comparing fifty natural languages and twelve genetic languages using word embedding language divergence (WELD) as a quantitative measure of language distance, 2016.

[15] AiTi Aw, Min Zhang, Juan Xiao, and Jian Su. A phrase-based statistical model for SMS text normalization. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 33–40. Association for Computational Linguistics, 2006.

[16] Frank B Baker. Information retrieval based upon latent class analysis. *Journal of the ACM (JACM)*, 9(4):512–521, 1962.

[17] Timothy Baldwin and Marco Lui. Language identification: The long and the short of the matter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 229–237, Los Angeles, CA, USA, June 2010. Association for Computational Linguistics (ACL).

[18] Utsab Barman, Amitava Das, Joachim Wagner, and Jennifer Foster. Code mixing: A challenge for language identification in the language of social media. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 13–23, Doha, Qatar, October 2014. Association for Computational Linguistics (ACL).

[19] Utsab Barman, Joachim Wagner, Grzegorz Chrupała, and Jennifer Foster. DCU-UVT: Word-level language classification with code-mixed data. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 127–132, Doha, Qatar, October 2014. Association for Computational Linguistics (ACL).

[20] Kenneth R Beesley. Language identifier: A computer program for automatic natural-language identification of on-line text. In *Proceedings of the 29th annual conference of the American Translators Association*, volume 47, page 54, 1988.

[21] Alessio Benavoli, Giorgio Corani, Janez Demšar, and Marco Zaffalon. Time for a change: A tutorial for comparing multiple classifiers through Bayesian analysis. *The Journal of Machine Learning Research*, 18(1):2653–2688, 2017.

[22] Alessandro Bessi and Emilio Ferrara. Social bots distort the 2016 U.S. presidential election online discussion. *First Monday*, 21(11), 2016.

[23] Sajid Yousuf Bhat and Muhammad Abulaish. OCTracker: A density-based framework for tracking the evolution of overlapping communities in OSNs. In *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*, ASONAM '12, pages 501–505, Washington, DC, USA, 2012. IEEE Computer Society.

[24] Sajid Yousuf Bhat and Muhammad Abulaish. Community-based features for identifying spammers in online social networks. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '13, pages 100–107, New York, NY, USA, 2013. ACM.

[25] Paulo Bicalho, Marcelo Pita, Gabriel Pedrosa, Anisio Lacerda, and Gisele L. Pappa. A general framework to expand short text for topic modeling. *Information Sciences*, 393(C):66–81, July 2017.

[26] David M. Blei. Probabilistic topic models. *Communications ACM*, 55(4):77–84, April 2012.

[27] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, March 2003.

[28] Brad Block. Collapsed variational HDP, 2011.

[29] Leticia Bode. Political news in the news feed: Learning politics from social media. *Mass Communication and Society*, 19(1):24–48, 2016.

[30] Nora Bohnert, Jonathan Chagnon, and Patrice Dion. Population projections for Canada (2013 to 2063), Provinces and Territories (2013 to 2038). Technical report, Statistics Canada, 2015.

[31] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *CoRR*, abs/1607.04606, 2016.

[32] Yasmin Bokobza, Abigail Paradise, Guy Rapaport, Rami Puzis, Bracha Shapira, and Asaf Shabtai. Leak sinks: The threat of targeted social eavesdropping. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, ASONAM '15, pages 375–382, New York, NY, USA, 2015. ACM.

[33] Harold Borko and Myrna Bernick. Automatic document classification. *Journal of the ACM (JACM)*, 10(2):151–162, April 1963.

[34] Yazan Boshmaf, Dionysios Logothetis, Georgos Siganos, Jorge Lera, Jose Lorenzo, Matei Ripeanu, Konstantin Beznosov, and Hassan Halawa. Íntegro: Leveraging victim prediction for robust fake account detection in large scale OSNs. *Computers & Security*, 61:142–168, 2016.

[35] Yazan Boshmaf, Ildar Muslukhov, Konstantin Beznosov, and Matei Ripeanu. The socialbot network: When bots socialize for fame and money. In *Proceedings of the 27th Annual Computer Security Applications Conference*, ACSAC '11, pages 93–102, New York, NY, USA, 2011. ACM.

[36] Yazan Boshmaf, Ildar Muslukhov, Konstantin Beznosov, and Matei Ripeanu. Design and analysis of a social botnet. *Computer Networks*, 57(2):556–578, February 2013.

[37] Houda Bouamor, Sabit Hassan, and Nizar Habash. The MADAR shared task on Arabic fine-grained dialect identification. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 199–207, Florence, Italy, August 2019. Association for Computational Linguistics (ACL).

[38] Florian Brachten, Stefan Stieglitz, Lennart Hofeditz, Katharina Kloppenborg, and Annette Reimann. Strategies and influence of social bots in a 2017 German state election - A case study on Twitter. *CoRR*, abs/1710.07562, 2017.

[39] Georgios-Ioannis Brokos, Prodromos Malakasiotis, and Ion Androutsopoulos. Using centroids of word embeddings and word mover's distance for biomedical document retrieval in question answering. *CoRR*, abs/1608.03905, 2016.

[40] Weicheng Cai, Danwei Cai, Shen Huang, and Ming Li. Utterance-level end-to-end language identification using attention-based CNN-BLSTM. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5991–5995, May 2019.

[41] Qiang Cao, Michael Sirivianos, Xiaowei Yang, and Tiago Pregueiro. Aiding the detection of fake accounts in large scale social online services. In *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation*, NSDI'12, pages 15–15, Berkeley, CA, USA, 2012. USENIX Association.

[42] Simon Carter, Wouter Weerkamp, and Manos Tsagkias. Microblog language identification: Overcoming the limitations of short, unedited and idiomatic text. *Language Resources and Evaluation*, 47(1):195–215, March 2013.

[43] William B. Cavnar and John M. Trenkle. N-gram-based text categorization. In *In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, 1994.

[44] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. Universal sentence encoder. *CoRR*, abs/1803.11175, 2018.

[45] Hakan Ceylan and Yookyung Kim. Language identification of search engine queries. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 1066–1074, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics (ACL).

[46] Nikan Chavoshi, Hossein Hamooni, and Abdullah Mueen. DeBot: Twitter bot detection via warped correlation. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pages 817–822, December 2016.

[47] Nikan Chavoshi, Hossein Hamooni, and Abdullah Mueen. Identifying correlated bots in Twitter. In *Social Informatics*, pages 14–21, Cham, 2016. Springer International Publishing.

[48] Guan-Bin Chen and Hung-Yu Kao. Word co-occurrence augmented topic model in short text. In *International Journal of Computational Linguistics & Chinese Language Processing, Volume 20, Number 2, December 2015 - Special Issue on Selected Papers from ROCLING XXVII*, December 2015.

[49] Xueqi Cheng, Xiaohui Yan, Yanyan Lan, and Jiafeng Guo. BTM: Topic modeling over short texts. *IEEE Transactions on Knowledge and Data Engineering*, 26(12):2928–2941, December 2014.

[50] Zi Chu, Steven Gianvecchio, Haining Wang, and Sushil Jajodia. Who is tweeting on Twitter: Human, bot, or cyborg? In *Proceedings of the 26th Annual Computer Security Applications Conference*, ACSAC '10, pages 21–30, New York, NY, USA, 2010. ACM.

[51] Zi Chu, Steven Gianvecchio, Haining Wang, and Sushil Jajodia. Detecting automation of Twitter accounts: Are you a human, bot, or cyborg? *IEEE Transactions on Dependable and Secure Computing*, 9(6):811–824, November 2012.

[52] Kenneth Church. Stress assignment in letter to sound rules for speech synthesis. In *Proceedings of the 23rd annual meeting on Association for Computational Linguistics*, pages 246–253, Chicago, IL, USA, July 1985. Association for Computational Linguistics.

[53] CIHR Institute of Aging. IA strategic plan 2013-2018: Living longer, living better. http://www.cihr-irsc.gc.ca/e/47179.html, 2013. Accessed: 2019-05-30.

[54] Çağrı Çöltekin and Taraka Rama. Discriminating similar languages with linear SVMs and neural networks. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 15–24, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.

[55] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. DNA-inspired online behavioral modeling and its application to spambot detection. *IEEE Intelligent Systems*, 31(5):58–64, 2016.

[56] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race. *CoRR*, abs/1701.03017, 2017.

[57] Maral Dadvar, FMG de Jong, Roeland Ordelman, and Dolf Trieschnigg. Improved cyberbullying detection using gender information. In *Proceedings of the Twelfth Dutch-Belgian Information Retrieval Workshop (DIR 2012)*. University of Ghent, 2012.

[58] Walter Daelemans, Mike Kestemont, Enrique Manjavancas, Martin Potthast, Francisco Rangel, Paolo Rosso, Günther Specht, Efstathios Stamatatos, Benno Stein, Michael Tschuggnall, Matti Wiegmann, and Eva Zangerle. Overview of PAN 2019: Author profiling, celebrity profiling, cross-domain authorship attribution and style change detection. In *Proceedings of the Tenth International Conference of the CLEF Association (CLEF 2019)*. Springer, September 2019.

[59] Ali Daud, Juanzi Li, Lizhu Zhou, and Faqir Muhammad. Knowledge discovery through directed probabilistic topic models: A survey. *Frontiers of computer science in China*, 4(2):280–301, 2010.

[60] Clayton Allen Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. BotOrNot: A system to evaluate social bots. In *Proceedings of the 25th International Conference Companion on World Wide Web*, WWW '16 Companion, pages 273–274, Republic and Canton of Geneva, Switzerland, 2016. International World Wide Web Conferences Steering Committee.

[61] Wim De Smet and Marie-Francine Moens. Cross-language linking of news stories on the Web using interlingual topic modelling. In *Proceedings of the 2Nd ACM Workshop on Social Web Search and Mining*, SWSM '09, pages 57–64, New York, NY, USA, 2009. ACM.

[62] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[63] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet. Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):788–798, 2010.

[64] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30, December 2006.

[65] Leon Derczynski, Alan Ritter, Sam Clark, and Kalina Bontcheva. Twitter part-of-speech tagging for all: Overcoming sparse and noisy data. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 198–206, Hissar, Bulgaria, September 2013. INCOMA Ltd. Shoumen, BULGARIA.

[66] Romain Deveaud, Eric San Juan, and Patrice Bellot. Are semantically coherent topic models useful for ad hoc information retrieval? In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 148–152, Sofia, Bulgaria, August 2013. Association for Computational Linguistics (ACL).

[67] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[68] John P. Dickerson, Vadim Kagan, and V. S. Subrahmanian. Using sentiment to detect bots on Twitter: Are humans more opinionated than bots? In *Proceedings of the 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '14, pages 620–627, Piscataway, NJ, USA, 2014. IEEE Press.

[69] Thomas G. Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923, October 1998.

[70] Nina Dongen. *Analysis and prediction of Dutch-English code-switching in Dutch social media messages*. Master's thesis, Universiteit van Amsterdam, Amsterdam, Netherlands, 2017.

[71] Gary F Simons Eberhard, David M and Charles D Fennig, editors. *Ethnologue: Languages of the World. Twenty-second edition*. SIL International, Dallas, TX, USA, 2019. Online version: http://www.ethnologue.com/.

[72] Chad Edwards, Autumn Edwards, Patric R. Spence, and Ashleigh K. Shelton. Is that a bot running the social media feed? Testing the differences in perceptions of communication quality for a human agent and a bot agent on Twitter. *Computers in Human Behavior*, 33:372–376, April 2014.

[73] Heba Elfardy and Mona Diab. Sentence level dialect identification in Arabic. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 456–461, Sofia, Bulgaria, August 2013. Association for Computational Linguistics (ACL).

[74] Aviad Elyashar, Michael Fire, Dima Kagan, and Yuval Elovici. Guided socialbots: Infiltrating the social networks of specific organizations' employees. *AI Communications*, 29:87–106, 2014.

[75] Ramy Eskander, Mohamed Al-Badrashiny, Nizar Habash, and Owen Rambow. Foreign words and the automatic processing of Arabic social media text written in Roman script. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 1–12, Doha, Qatar, October 2014. Association for Computational Linguistics (ACL).

[76] Richard M. Everett, Jason R. C. Nurse, and Arnau Erola. The anatomy of online deception: What makes automated text convincing? In *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, SAC '16, pages 1115–1120, New York, NY, USA, 2016. ACM.

[77] Mehwish Fatima, Komal Hasan, Saba Anwar, and Rao Muhammad Adeel Nawab. Multilingual author profiling on Facebook. *Information Processing & Management*, 53(4):886–904, 2017.

[78] Amir Fayazi, Kyumin Lee, James Caverlee, and Anna Squicciarini. Uncovering crowdsourced manipulation of online reviews. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, pages 233–242, New York, NY, USA, 2015. ACM.

[79] José Fernández Huerta. Medidas sencillas de lecturabilidad. *Consigna*, 214:29–32, 1959.

[80] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini. The rise of social bots. *Communications of the ACM*, 59(7):96–104, 2016.

[81] Emilio Ferrara, Onur Varol, Filippo Menczer, and Alessandro Flammini. Detection of promoted social media campaigns. In *tenth international AAAI conference on Web and social media*, 2016.

[82] Ramon Ferrer i Cancho, Ricard V. Solé, and Reinhard Köhler. Patterns in syntactic dependency networks. *Physical Review E*, 69:51915, May 2004.

[83] John W. Fleenor, Julie B. Fleenor, and William F. Grossnickle. Interrater reliability and agreement of performance ratings: A methodological comparison. *Journal of Business and Psychology*, 10(3):367–380, March 1996.

[84] Rudolf Flesch and Alan J Gould. *The Art of Readable Writing*, volume 8. Harper New York, 1949.

[85] Claudia Flores-Saviaga, Saiph Savage, and Dario Taraborelli. LeadWise: Using online bots to recruit and guide expert volunteers. In *Proceedings of the 19th*

ACM Conference on Computer Supported Cooperative Work and Social Computing Companion, CSCW '16 Companion, pages 257–260, New York, NY, USA, 2016. ACM.

[86] Michelle Forelle, Philip N. Howard, Andrés Monroy-Hernández, and Saiph Savage. Political bots and the manipulation of public opinion in Venezuela. *CoRR*, abs/1507.07109, 2015.

[87] Marc Franco-Salvador, Greg Kondrak, and Paolo Rosso. Bridging the native language and language variety identification tasks. *Procedia Computer Science*, 112:1554–1561, 2017. Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 21st International Conference, KES-20176-8 September 2017, Marseille, France.

[88] Marc Franco-Salvador, Nataliia Plotnikova, Neha Pawar, and Yassine Benajiba. Subword-based deep averaging networks for author profiling in social media — Notebook for PAN at CLEF 2017. In *Working Notes Papers of CLEF 2017 Evaluation Labs and Workshop*, Dublin, Ireland, September 2017.

[89] Marc Franco-Salvador, Paolo Rosso, and Francisco Rangel. Distributed representations of words and documents for discriminating similar languages. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*, pages 11–16, Hissar, Bulgaria, September 2015. Association for Computational Linguistics (ACL).

[90] Georgia Frantzeskou, Efstathios Stamatatos, Stefanos Gritzalis, and Sokratis Katsikas. Effective identification of source code authors using byte-level information. In *In Proceedings of the 28th International Conference on Software Engineering*, pages 893–896, 2006.

[91] Carlos Freitas, Fabricio Benevenuto, Saptarshi Ghosh, and Adriano Veloso. Reverse engineering socialbot infiltration strategies in Twitter. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, ASONAM '15, pages 25–32, New York, NY, USA, 2015. ACM.

[92] Pablo Gamallo, Marcos Garcia, and Susana Sotelo. Comparing ranking-based and Naïve bayes approaches to language detection on tweets. In *Proceedings of the Tweet Language Identification Workshop 2014 co-located with 30th Conference of the Spanish Society for Natural Language Processing (SEPLN 2014)*, pages 12–16, Girona, Spain, 2014.

[93] Pablo Gamallo, Jos Ramom Pichel, and Iaki Alegria. From language identification to language distance. *Physica A: Statistical Mechanics and its Applications*, 484:152–162, 2017.

[94] Kavita Ganesan, Cheng Xiang Zhai, and Jiawei Han. Opinosis: A graph based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 340–348, Beijing, China, August 2010. COLING 2010 Organizing Committee.

[95] Yuyang Gao, Wei Liang, Yuming Shi, and Qiuling Huang. Comparison of directed and weighted co-occurrence networks of six languages. *Physica A: Statistical Mechanics and its Applications*, 393:579–589, 2014.

[96] Archana Garg, Vishal Gupta, and Manish Jindal. A survey of language identification techniques and applications. *Journal of Emerging Technologies in Web Intelligence*, 6(4):388–400, 2014.

[97] Wang Geng, Wenfu Wang, Yuanyuan Zhao, Xinyuan Cai, and Bo Xu. End-to-end language identification using attention-based recurrent neural networks. In *Interspeech 2016*, pages 2944–2948, 2016.

[98] Fausto Giunchiglia, Khuyagbaatar Batsuren, and Gabor Bella. Understanding and exploiting language diversity. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 4009–4017, 2017.

[99] Oana Goga, Giridhari Venkatadri, and Krishna P. Gummadi. The doppelgänger bot attack: Exploring identity impersonation in online social networks. In *Proceedings of the 2015 Internet Measurement Conference*, IMC '15, pages 141–153, New York, NY, USA, 2015. ACM.

[100] Sumit Goswami, Sudeshna Sarkar, and Mayur Rustagi. Stylometric analysis of bloggers' age and gender. In *Proceedings of the Third International Conference on Weblogs and Social Media*, ICWSM '09, San Jose, CA, USA, 5 2009. The AAAI Press.

[101] Norbert Gövert and Gabriella Kazai. Overview of the initiative for the evaluation of XML retrieval (INEX) 2002. In *Proceedings of the First Workshop of the INitiative for the Evaluation of XML Retrieval (INEX), Schloss Dagstuhl, Germany, December 9-11, 2002*, pages 1–17, 2002.

[102] Andrew Guess, Jonathan Nagler, and Joshua Tucker. Less than you think: Prevalence and predictors of fake news dissemination on Facebook. *Science advances*, 5(1):eaau4586, 2019.

[103] Weiwei Guo, Hao Li, Heng Ji, and Mona Diab. Linking tweets to news: A framework to enrich short text data in social media. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 239–249, Sofia, Bulgaria, August 2013. Association for Computational Linguistics (ACL).

[104] Malek Hajjem and Chiraz Latiri. Combining IR and LDA topic modeling for filtering microblogs. *Procedia Computer Science*, 112:761–770, 2017. Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 21st International Conference, KES-20176-8 September 2017, Marseille, France.

[105] Bo Han and Timothy Baldwin. Lexical normalisation of short text messages: Makn sens a# twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 368–378. Association for Computational Linguistics, 2011.

[106] Donna Harman. Overview of the first TREC conference. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '93, pages 36–47, New York, NY, USA, 1993. ACM.

[107] Junqing He, Zhen Zhang, Xuemin Zhao, Peijia Li, and Yonghong Yan. Similar language identification for Uyghur and Kazakh on short spoken texts. In *2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, volume 2, pages 496–499, August 2016.

[108] Yukun He, Qiang Li, Jian Cao, Yuede Ji, and Dong Guo. Understanding socialbot behavior on end hosts. *International Journal of Distributed Sensor Networks*, 13(2):1550147717694170, 2017.

[109] Simon Hegelich and Dietmar Janetzko. Are social bots on Twitter political actors? Empirical evidence from a Ukrainian social botnet. In *Tenth International AAAI Conference on Web and Social Media*, 2016.

[110] Erin Hengel. Publishing while female. are women held to higher standards? evidence from peer review. Cambridge working papers in economics, Faculty of Economics, University of Cambridge, 2017.

[111] Peter Henrich. Language identification for the automatic grapheme-to-phoneme conversion of foreign words in a German text-to-speech system. In *EUROSPEECH*, 1989.

[112] Geoffrey E Hinton and Terrence J Sejnowski. Optimal perceptual inference. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 448–453. Citeseer, 1983.

[113] Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, UAI'99, pages 289–296, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[114] Liangjie Hong and Brian D. Davison. Empirical study of topic modeling in Twitter. In *Proceedings of the First Workshop on Social Media Analytics*, SOMA '10, pages 80–88, New York, NY, USA, 2010. ACM.

[115] Eva Hörster, Rainer Lienhart, and Malcolm Slaney. Image retrieval on large-scale image databases. In *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*, CIVR '07, pages 17–24, New York, NY, USA, 2007. ACM.

[116] Arthur S House and Edward P Neuburg. Toward automatic identification of the language of an utterance. I. Preliminary methodological considerations. *The Journal of the Acoustical Society of America*, 62(3):708–713, 1977.

[117] Philip N Howard, Samuel Woolley, and Ryan Calo. Algorithms, bots, and political communication in the US 2016 election: The challenge of automated political communication for election law and administration. *Journal of Information Technology & Politics*, 15(2):81–93, 2018.

[118] Weihua Hu and Jun'ichi Tsujii. A latent concept topic model for robust topic inference using word embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 380–386, Berlin, Germany, August 2016. Association for Computational Linguistics (ACL).

[119] Rodrigo Augusto Igawa, Sylvio Barbon Jr, Ktia Cristina Silva Paulo, Guilherme Sakaji Kido, Rodrigo Capobianco Guido, Mario Lemes Proena Jnior, and Ivan Nunes da Silva. Account classification in online social networks with LBCA and wavelets. *Information Sciences*, 332:72–83, 2016.

[120] Aminul Islam and Diana Inkpen. Real-word spelling correction using Google Web 1T 3-grams. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3 - Volume 3*, EMNLP '09, pages 1241–1249, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics (ACL).

[121] Aaron Jaech, George Mulcaire, Shobhit Hathi, Mari Ostendorf, and Noah A. Smith. Hierarchical character-word models for language identification. In *Proceedings of The Fourth International Workshop on Natural Language Processing for Social Media*, pages 84–93, Austin, TX, USA, November 2016. Association for Computational Linguistics (ACL).

[122] Jagadeesh Jagarlamudi and Hal Daumé. Extracting multilingual topics from unaligned comparable corpora. In *Proceedings of the 32Nd European Conference on Advances in Information Retrieval*, ECIR'2010, pages 444–456, Berlin, Heidelberg, 2010. Springer-Verlag.

[123] Jagadeesh Jagarlamudi, Hal Daumé, III, and Raghavendra Udupa. Incorporating lexical priors into topic models. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '12, pages 204–213, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[124] Magdalena Jankowska, Vlado Kešelj, and Evangelos Milios. Relative N-gram signatures: Document visualization at the level of character N-grams. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, pages 103–112, October 2012.

[125] Magdalena Jankowska, Evangelos Milios, and Vlado Kešelj. Author verification using common n-gram profiles of text documents. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 387–397, Dublin, Ireland, August 2014. Dublin City University and Association for Computational Linguistics.

[126] Tommi Jauhiainen, Marco Lui, Marcos Zampieri, Timothy Baldwin, and Krister Lindén. Automatic language identification in texts: A survey. *arXiv preprint arXiv:1804.08186*, 2018.

[127] Hamed Jelodar, Yongli Wang, Chi Yuan, Xia Feng, Xiahui Jiang, Yanchao Li, and Liang Zhao. Latent Dirichlet allocation (LDA) and topic modeling: Models, applications, a survey. *Multimedia Tools and Applications*, 78(11):15169–15211, 2019.

[128] Ou Jin, Nathan N Liu, Kai Zhao, Yong Yu, and Qiang Yang. Transferring topical knowledge from auxiliary long texts for short text clustering. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, pages 775–784, New York, NY, USA, 2011. ACM.

[129] K. Sparck Jones, S. Walker, and S.E. Robertson. A probabilistic model of information retrieval: Development and comparative experiments: Part 1. *Information Processing & Management*, 36(6):779–808, 2000.

[130] Armand Joulin, Piotr Bojanowski, Tomas Mikolov, Hervé Jégou, and Edouard Grave. Loss in translation: Learning bilingual word mapping with a retrieval criterion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2979–2984, Brussels, Belgium, October-November 2018. Association for Computational Linguistics (ACL).

[131] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, 2016.

[132] Sylvio Barbon Jr, Gabriel F. C. Campos, Gabriel M. Tavares, Rodrigo A. Igawa, Mario L. Proença Jr, and Rodrigo Capobianco Guido. Detection of human, legitimate bot, and malicious bot in online social networks based on wavelets. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 14(1s):26:1–26:17, March 2018.

[133] Abdullah Talha Kabakus and Resul Kara. A survey of spam detection methods on Twitter. *International Journal of Advanced Computer Science and Applications*, 8(3), 2017.

[134] Noriko Kando, Kazuko Kuriyama, Toshihiko Nozue, Koji Eguchi, Hiroyuki Kato, and Souichiro Hidaka. Overview of IR tasks at the first NTCIR workshop. *Proceedings of the first NTCIR workshop on research in Japanese text retrieval and term recognition*, pages 11–44, September 1999.

[135] Vlado Kešelj, Fuchun Peng, Nick Cercone, and Calvin Thomas. N-gram-based author profiles for authorship attribution. In *Proceedings of the conference pacific association for computational linguistics, PACLING*, volume 3, pages 255–264. sn, 2003.

[136] Gen-itiro Kikui. Identifying, the coding system and language, of on-line documents on the Internet. In *Proceedings of the 16th Conference on Computational Linguistics - Volume 2*, COLING '96, pages 652–657, Stroudsburg, PA, USA, 1996. Association for Computational Linguistics.

[137] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *CoRR*, abs/1609.02907, 2016.

[138] Tom Kocmi and Ondřej Bojar. LanideNN: Multilingual language identification on text stream. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 927–936, Valencia, Spain, April 2017. Association for Computational Linguistics (ACL).

[139] Andrey Kolmogorov. Sulla determinazione empirica di una legge di distribuzione. *Giornale dell'Istituto Italiano degli Attuari*, 4:83–91, 1933.

[140] Dijana Kosmajac and Vlado Kešelj. Language identification in multilingual, short and noisy texts using common n-grams. In *2017 IEEE International Conference on Big Data (Big Data)*, pages 2752–2759, December 2017.

[141] Dijana Kosmajac and Vlado Kešelj. Slavic language identification using cascade classifier approach. In *2018 17th International Symposium INFOTEH-JAHORINA (INFOTEH)*, pages 1–6, March 2018.

[142] Dijana Kosmajac and Vlado Kešelj. Automatic text summarization of news articles in Serbian language. *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*, pages 1–6, 2019.

[143] Dijana Kosmajac and Vlado Kešelj. Twitter bot detection using diversity measures. In *Proceedings of the 3rd International Conference on Natural Language and Speech Processing*, pages 1–8, Trento, Italy, 12–13 September 2019. Association for Computational Linguistics.

[144] Dijana Kosmajac and Vlado Kešelj. Twitter user profiling: Bot and gender identification. In *CLEF 2019 Labs and Workshops, Notebook Papers*. CEUR-WS.org, September 2019.

[145] Dijana Kosmajac, Vlado Kešelj, and Evangelos E. Milios. EulerianGrapher: Text visualisation at the level of character n-grams based on Eulerian graphs. In *ESIDA@IUI*, 2017.

[146] Canasai Kruengkrai, Prapass Srichaivattana, Virach Sornlertlamvanich, and Hitoshi Isahara. Language identification based on string kernels. In *IEEE International Symposium on Communications and Information Technology, 2005. ISCIT 2005.*, volume 2, pages 926–929. IEEE, 2005.

[147] Rahul Venkatesh RM Kumar, Anand M Kumar, and KP Soman. Amrita-CEN_NLP FIRE 2015 language identification for Indian languages in social media text. In *Working Notes of the Forum for Information Retrieval Evaluation (FIRE 2015)*, pages 28–30, Gandhinagar, India, December 2015.

[148] Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 957–966, Lille, France, 07–09 Jul 2015. PMLR.

[149] Man Lan, Chew Lim Tan, Jian Su, and Yue Lu. Supervised and traditional term weighting methods for automatic text categorization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(4):721–735, April 2009.

[150] Stefan Langer and Elexir Gmbh. Natural languages and the World Wide Web. *Bulletin de linguistique appliquée et générale*, 26:89–100, 2001.

[151] Hugo Larochelle and Stanislas Lauly. A neural autoregressive topic model. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 2708–2716. Curran Associates, Inc., 2012.

[152] Jey Han Lau, David Newman, and Timothy Baldwin. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 530–539, Gothenburg, Sweden, April 2014. Association for Computational Linguistics (ACL).

[153] Omer Levy, Yoav Goldberg, and Ido Dagan. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225, 2015.

[154] Cheng Li, Santu Rana, Dinh Phung, and Svetha Venkatesh. Hierarchical Bayesian nonparametric models for knowledge discovery from electronic medical records. *Knowledge-Based Systems*, 99:168–182, 2016.

[155] Chenliang Li, Haoran Wang, Zhiqian Zhang, Aixin Sun, and Zongyang Ma. Topic modeling for short texts with auxiliary word embeddings. In *Proceedings*

of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '16, pages 165–174, New York, NY, USA, 2016. ACM.

[156] Kar Wai Lim, Changyou Chen, and Wray L Buntine. Twitter-network topic model: A full Bayesian treatment for social network and text modeling. *CoRR*, abs/1609.06791, 2016.

[157] Tianyi Lin, Wentao Tian, Qiaozhu Mei, and Hong Cheng. The dual-sparse topic model: Mining focused topics and focused terms in short text. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14, pages 539–550, New York, NY, USA, 2014. ACM.

[158] Bing Liu and Lei Zhang. A survey of opinion mining and sentiment analysis. In *Mining Text Data*, pages 415–463. Springer US, Boston, MA, 2012.

[159] Haitao Liu and Wenwen Li. Language clusters based on linguistic complex networks. *Chinese Science Bulletin*, 55:3458–3465, 2010.

[160] Nikola Ljubešić, Nives Mikelić, and Damir Boras. Language indentification: How to distinguish similar languages? In *2007 29th International Conference on Information Technology Interfaces*, pages 541–546, June 2007.

[161] Tetyana Lokot and Nicholas Diakopoulos. News bots: Automating news and information dissemination on Twitter. *Digital Journalism*, 4(6):682–699, 8 2016.

[162] Heng-Yang Lu, Lu-Yao Xie, Ning Kang, Chong-Jun Wang, and Jun-Yuan Xie. Don't forget the quantifiable relationship between words: Using recurrent neural network for short text topic discovery. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[163] Hsin-Min Lu, Chih-Ping Wei, and Fei-Yuan Hsiao. Modeling healthcare data using multiple-channel latent Dirichlet allocation. *Journal of Biomedical Informatics*, 60:210–223, 2016.

[164] Luca Luceri, Ashok Deb, Adam Badawy, and Emilio Ferrara. Red bots do it better: Comparative analysis of social bot partisan behavior. In *Companion Proceedings of The 2019 World Wide Web Conference*, WWW '19, pages 1007–1012, New York, NY, USA, 2019. ACM.

[165] Yevgeny Ludovik, Ron Zacharski, and James R Cowie. Language recognition for mono-and multi-lingual documents. In *Proceedings of the VexTal Conference*, pages 209–214, Venice, Italy, November 1999.

[166] Marco Lui and Timothy Baldwin. langid.py: An off-the-shelf language identification tool. In *Proceedings of the ACL 2012 System Demonstrations*, pages 25–30, Jeju Island, Korea, July 2012. Association for Computational Linguistics (ACL).

[167] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[168] Shane MacNamara, Pádraig Cunningham, and John Byrne. Neural networks for language identification: A comparative study. *Information Processing & Management*, 34(4):395–403, 1998.

[169] Gabriel Magno and Ingmar Weber. International gender differences and gaps in online social networks. In *Social Informatics: 6th International Conference, SocInfo 2014, Barcelona, Spain, November 11-13, 2014. Proceedings*, pages 121–138, Cham, 2014. Springer International Publishing.

[170] Winnie Main and Narendra Shekokhar. Twitterati identification system. *Procedia Computer Science*, 45:32–41, 2015. International Conference on Advanced Computing Technologies and Applications (ICACTA).

[171] Prasenjit Majumder, Mandar Mitra, Dipasree Pal, Ayan Bandyopadhyay, Samaresh Maiti, Sukomal Pal, Deboshree Modak, and Sucharita Sanyal. The FIRE 2008 evaluation exercise. *ACM Transactions on Asian Language Information Processing (TALIP)*, 9(3):10, 2010.

[172] Shervin Malmasi, Marcos Zampieri, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, and Jörg Tiedemann. Discriminating between similar languages and Arabic dialect identification: A report on the third DSL shared task. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 1–14, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.

[173] Christopher Manning, Prabhakar Raghavan, and Hinrich Schütze. Introduction to information retrieval. *Natural Language Engineering*, 16(1):100–103, 2010.

[174] Robin Mansell. *The information society*. Routledge, 2009.

[175] Justin Martineau and Tim Finin. Delta TFIDF: An improved feature space for sentiment analysis. In *ICWSM*, 2009.

[176] Jane E Mason, Michael Shepherd, Jack Duffy, Vlado Kešelj, and Carolyn Watters. An n-gram based approach to multi-labeled Web page genre classification. In *2010 43rd Hawaii International Conference on System Sciences*, pages 1–10. IEEE, 2010.

[177] Dina Mayzlin, Yaniv Dover, and Judith Chevalier. Promotional reviews: An empirical investigation of online review manipulation. *American Economic Review*, 104(8):2421–55, August 2014.

[178] Walaa Medhat, Ahmed Hassan, and Hoda Korashy. Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5(4):1093–1113, 2014.

[179] Rishabh Mehrotra, Scott Sanner, Wray Buntine, and Lexing Xie. Improving LDA topic models for microblogs via tweet pooling and automatic labeling. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 889–892, New York, NY, USA, 2013. ACM.

[180] Johnnatan Messias, Lucas Schmidt, Ricardo Oliveira, and Fabrício Benevenuto. You followed my bot! Transforming robots into influential users in Twitter. *First Monday*, 18(7), 2013.

[181] Yishu Miao, Edward Grefenstette, and Phil Blunsom. Discovering discrete latent topics with neural variational inference. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2410–2419. JMLR. org, 2017.

[182] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[183] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc., 2013.

[184] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia, June 2013. Association for Computational Linguistics (ACL).

[185] George A. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41, November 1995.

[186] David Mimno, Hanna M Wallach, Jason Naradowsky, David A Smith, and Andrew McCallum. Polylingual topic models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 880–889, Singapore, August 2009. Association for Computational Linguistics (ACL).

[187] David Mimno, Hanna M Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. Optimizing semantic coherence in topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 262–272, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics (ACL).

[188] Toby J. Mitchell and John J. Beauchamp. Bayesian variable selection in linear regression. *Journal of the American Statistical Association*, 83(404):1023–1032, 1988.

[189] Andriy Mnih and Geoffrey E Hinton. A scalable hierarchical distributed language model. In *Advances in Neural Information Processing Systems 21*, pages 1081–1088. Curran Associates, Inc., 2009.

[190] Seppo Mustonen. Multiple discriminant analysis in linguistic problems. *Statistical Methods in Linguistics*, 4:37–44, 1965.

[191] Mor Naaman, Hila Becker, and Luis Gravano. Hip and trendy: Characterizing emerging trends on Twitter. *Journal of the American Society for Information Science and Technology*, 62(5):902–918, May 2011.

[192] Shishir Nagaraja, Amir Houmansadr, Pratch Piyawongwisal, Vijit Singh, Pragya Agarwal, and Nikita Borisov. Stegobot: A covert social network botnet. In *Information Hiding*, pages 299–313, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[193] Roberto Navigli and Simone Paolo Ponzetto. BabelNet: Building a very large multilingual semantic network. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 216–225, Uppsala, Sweden, July 2010. Association for Computational Linguistics (ACL).

[194] Neo4j graph platform. https://neo4j.com/, 2019. Accessed: 2019-05-30.

[195] David Newman, Edwin V Bonilla, and Wray Buntine. Improving topic coherence with regularized topic models. In *Advances in Neural Information Processing Systems 24*, pages 496–504. Curran Associates, Inc., 2011.

[196] David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. Automatic evaluation of topic coherence. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 100–108, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics (ACL).

[197] David Newman, Youn Noh, Edmund Talley, Sarvnaz Karimi, and Timothy Baldwin. Evaluating topic models for digital libraries. In *Proceedings of the 10th Annual Joint Conference on Digital Libraries*, JCDL '10, pages 215–224, New York, NY, USA, 2010. ACM.

[198] Matthew L. Newman, Carla J. Groom, Lori D. Handelman, and James W. Pennebaker. Gender differences in language use: An analysis of 14,000 text samples. *Discourse Processes*, 45(3):211–236, 2008.

[199] Choon-Ching Ng and Ali Selamat. Improved letter weighting feature selection on Arabic script language identification. In *Proceedings of the 2009 First Asian Conference on Intelligent Information and Database Systems*, ACIIDS '09, pages 150–154, Washington, DC, USA, 2009. IEEE Computer Society.

[200] Dat Quoc Nguyen. jLDADMM: A Java package for the LDA and DMM topic models. *CoRR*, abs/1808.03835, 2018.

[201] Dat Quoc Nguyen, Richard Billingsley, Lan Du, and Mark Johnson. Improving topic models with latent feature word representations. *Transactions of the Association for Computational Linguistics*, 3:299–313, 2015.

[202] Dong-Phuong Nguyen, Rilana Gravel, Rudolf Berend Trieschnigg, and Theo Meder. "How old do you think I am?": A study of language and age in Twitter. In *Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media, ICWSM 2013*, pages 439–448. AAAI Press, 7 2013. eemcs-eprint-23604.

[203] Xiaochuan Ni, Jian-Tao Sun, Jian Hu, and Zheng Chen. Mining multilingual topics from Wikipedia. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pages 1155–1156, New York, NY, USA, 2009. ACM.

[204] Jianwei Niu, Qingjuan Zhao, Lei Wang, Huan Chen, Mohammed Atiquzzaman, and Fei Peng. OnSeS: A novel online short text summarization based on BM25 and neural network. In *2016 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, December 2016.

[205] Daisuke Okanohara and Jun'ichi Tsujii. Text categorization with all substring features. In *Proceedings of the 2009 SIAM International Conference on Data Mining*, pages 838–846, 2009.

[206] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the Web. Technical Report 1999-66, Stanford InfoLab, November 1999. Previous number = SIDL-WP-1999-0120.

[207] Georgios Paltoglou and Mike Thelwall. A study of information retrieval weighting schemes for sentiment analysis. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1386–1395, Uppsala, Sweden, July 2010. Association for Computational Linguistics (ACL).

[208] Nick Pantic and Mohammad I. Husain. Covert botnet command and control using Twitter. In *Proceedings of the 31st Annual Computer Security Applications Conference*, ACSAC 2015, pages 171–180, New York, NY, USA, 2015. ACM.

[209] Abigail Paradise, Rami Puzis, and Asaf Shabtai. Anti-reconnaissance tools: Detecting targeted socialbots. *IEEE Internet Computing*, 18(5):11–19, September 2014.

[210] Timo Pawelka and Elmar Jürgens. Is this code written in English? A study of the natural language of comments and identifiers in practice. *2015 IEEE*

International Conference on Software Maintenance and Evolution (ICSME), pages 401–410, 2015.

[211] Claudia Peersman, Walter Daelemans, and Leona Van Vaerenbergh. Predicting age and gender in online social networks. In *Proceedings of the 3rd International Workshop on Search and Mining User-generated Contents*, SMUC '11, pages 37–44, New York, NY, USA, 2011. ACM.

[212] Claudia Peersman, Walter Daelemans, and Leona Van Vaerenbergh. Predicting age and gender in online social networks. In *Proceedings of the 3rd international workshop on Search and mining user-generated contents*, pages 37–44. ACM, 2011.

[213] James Pennebaker and Lori Stone. Words of wisdom: Language use over the life span. *Journal of Personality and Social Psychology*, 85:291–301, 9 2003.

[214] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics (ACL).

[215] Filippo Petroni and Maurizio Serva. Measures of lexical distance between languages. *Physica A: Statistical Mechanics and its Applications*, 389(11):2280–2283, 2010.

[216] James Petterson, Wray Buntine, Shravan M Narayanamurthy, Tibério S Caetano, and Alex J Smola. Word features for latent Dirichlet allocation. In *Advances in Neural Information Processing Systems 23*, pages 1921–1929. Curran Associates, Inc., 2010.

[217] Andra-Selina Pietsch and Stefan Lessmann. Topic modeling for analyzing open-ended survey responses. *Journal of Business Analytics*, 1(2):93–116, 2018.

[218] Pietro Pinoli, Davide Chicco, and Marco Masseroli. Latent Dirichlet allocation based on Gibbs sampling for gene function prediction. In *2014 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology*, pages 1–8, May 2014.

[219] Rimma Pivovarov and Nomie Elhadad. Automated methods for the summarization of electronic health records. *Journal of the American Medical Informatics Association*, 22(5):938–947, 4 2015.

[220] Juan Pizarro. Using n-grams to detect bots on Twitter. In *CLEF 2019 Labs and Workshops, Notebook Papers*. CEUR-WS.org, September 2019.

[221] John C Platt, Kristina Toutanova, and Wen-tau Yih. Translingual document representations from discriminative projections. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10,

pages 251–261, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics (ACL).

[222] Jordi Porta. Twitter language identification using rational kernels and its potential application to sociolinguistics. In *Proceedings of the Tweet Language Identification Workshop 2014 co-located with 30th Conference of the Spanish Society for Natural Language Processing (SEPLN 2014)*, volume 1228, Girona, Spain, 9 2014.

[223] Martin Potthast, Tim Gollub, Matti Wiegmann, and Benno Stein. TIRA integrated research architecture. In *Information Retrieval Evaluation in a Changing World - Lessons Learned from 20 Years of CLEF*. Springer, 2019.

[224] Nejla Qafmolla. Automatic language identification. *European Journal of Language and Literature*, 3(1):140–150, 2017.

[225] Jipeng Qiang, Yun Li, Yunhao Yuan, Wei Liu, and Xindong Wu. STTM: A tool for short text topic modeling. *arXiv preprint arXiv:1808.02215*, 2018.

[226] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 248–256, Singapore, August 2009. Association for Computational Linguistics (ACL).

[227] Francisco Rangel and Paolo Rosso. Overview of the 7th author profiling task at PAN 2019: Bots and gender profiling. In *CLEF 2019 Labs and Workshops, Notebook Papers*. CEUR-WS.org, September 2019.

[228] Jacob Ratkiewicz, Michael Conover, Mark Meiss, Bruno Gonçalves, Alessandro Flammini, and Filippo Menczer. Detecting and tracking political abuse in social media. In *In Proceedings of the 5th AAAI International Conference on Weblogs and Social Media (ICWSM11)*, 2011.

[229] Morton D Rau. Language identification by statistical analysis. Technical report, Naval Postgraduate School, Monterey, CA, USA, 1974.

[230] Radim Rehůřek and Milan Kolkus. Language identification on the Web: Extending the dictionary method. In *Proceedings of the 10th International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing '09, pages 357–368, Berlin, Heidelberg, 2009. Springer-Verlag.

[231] Margaret E Roberts, Brandon M Stewart, Dustin Tingley, Christopher Lucas, Jetson Leder-Luis, Shana Kushner Gadarian, Bethany Albertson, and David G Rand. Structural topic models for open-ended survey responses. *American Journal of Political Science*, 58(4):1064–1082, 2014.

[232] Stephen E Robertson and K Spärck Jones. Relevance weighting of search terms. *Journal of the American Society for Information science*, 27(3):129–146, 1976.

[233] Michael Röder, Andreas Both, and Alexander Hinneburg. Exploring the space of topic coherence measures. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, WSDM '15, pages 399–408, New York, NY, USA, 2015. ACM.

[234] Salto Martinez Rodrigo and Jacques Garcia Fausto Abraham. Development and implementation of a chat bot in a social network. In *Proceedings of the 2012 Ninth International Conference on Information Technology - New Generations*, ITNG '12, pages 751–755, Washington, DC, USA, 2012. IEEE Computer Society.

[235] Paolo Rosso, Francisco Rangel, Irazu Hernández Farías, Leticia Cagnina, Wajdi Zaghouani, and Anis Charfi. A survey on author profiling, deception, and irony detection for the Arabic language. *Language and Linguistics Compass*, 12(4):e12275, 2018. e12275 LNCO-0720.R1.

[236] François Rousseau and Michalis Vazirgiannis. Composition of TF normalizations: New insights on scoring functions for ad hoc IR. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 917–920, New York, NY, USA, 2013. ACM.

[237] Stephanie Rude, Eva-Maria Gortner, and James Pennebaker. Language use of depressed and depression-vulnerable college students. *Cognition and Emotion*, 18(8):1121–1133, 2004.

[238] Benoît Sagot. The Lefff, a freely available and large-coverage morphological and syntactic lexicon for French. In *7th international conference on Language Resources and Evaluation (LREC 2010)*, Valletta, Malta, May 2010.

[239] Armin Sajadi, Evangelos E. Milios, and Vlado Keselj. Vector space representation of concepts using Wikipedia graph structure. In *NLDB*, volume 10260 of *Lecture Notes in Computer Science*, pages 393–405. Springer, 2017.

[240] Ruslan Salakhutdinov and Geoffrey Hinton. Deep Boltzmann machines. In *Artificial intelligence and statistics*, pages 448–455, 2009.

[241] Mohammad Salameh, Houda Bouamor, and Nizar Habash. Fine-grained Arabic dialect identification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1332–1344, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics (ACL).

[242] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523, August 1988.

[243] Tanja Samardžić, Yves Scherrer, and Elvira Glaser. ArchiMob - A corpus of spoken Swiss German. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4061–4066, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA).

[244] Younes Samih and Laura Kallmeyer. *Dialectal Arabic Processing Using Deep Learning*. PhD thesis, The Faculty of Arts and Humanities at Heinrich Heine University, 2017.

[245] Ruchita Sarawgi, Kailash Gajulapalli, and Yejin Choi. Gender attribution: Tracing stylometric evidence beyond topic and genre. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, pages 78–86. Association for Computational Linguistics (ACL), 2011.

[246] Saiph Savage, Andres Monroy-Hernandez, and Tobias Höllerer. Botivist: Calling volunteers to action using online bots. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, CSCW '16, pages 813–822, New York, NY, USA, 2016. ACM.

[247] Jonathan Schler, Moshe Koppel, Shlomo Argamon, and James W Pennebaker. Effects of age and gender on blogging. In *Proceedings of AAAI Symposium on Computational Approaches for Analyzing Weblogs, 2006*, volume 6, pages 199–205, 2006.

[248] Johannes Schneider and Michail Vlachos. Topic modeling based on keywords and context. In *Proceedings of the 2018 SIAM International Conference on Data Mining*, pages 369–377, 2018.

[249] H. Andrew Schwartz, Johannes C. Eichstaedt, Margaret L. Kern, Lukasz Dziurzynski, Stephanie M. Ramones, Megha Agrawal, Achal Shah, Michal Kosinski, David Stillwell, Martin E. P. Seligman, and Lyle H. Ungar. Personality, gender, and age in the language of social media: The open-vocabulary approach. *PLOS ONE*, 8(9):1–16, 9 2013.

[250] Rushdi Shams and Robert E. Mercer. Classifying spam emails using text and readability features. In *2013 IEEE 13th International Conference on Data Mining*, pages 657–666, December 2013.

[251] Rushdi Shams and Robert E. Mercer. Supervised classification of spam emails with natural language stylometry. *Neural Computing and Applications*, 27(8):2315–2331, November 2016.

[252] Debra J. Sheets and Elaine M. Gallagher. Aging in Canada: State of the Art and Science. *The Gerontologist*, 53(1):1–8, 11 2012.

[253] Kai Shu, Suhang Wang, and Huan Liu. Understanding user profiles on social media for fake news detection. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 430–435. IEEE, 2018.

[254] H. S. Sichel. On a distribution law for word frequencies. *Journal of the American Statistical Association*, 70(351a):542–547, 1975.

[255] Utpal Kumar Sikdar and Björn Gambäck. Language identification in code-switched text using conditional random fields and BabelNet. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 127–131, Austin, Texas, November 2016. Association for Computational Linguistics (ACL).

[256] Vasiliki Simaki, Panagiotis Simakis, Carita Paradis, and Andreas Kerren. Identifying the authors' national variety of English in social media text. In *Proceedings of RANLP 2017 - Recent Advances in Natural Language Processing*, pages 671–678. Association for Computational Linguistics (ACL), 2017.

[257] Anil Kumar Singh and Jagadeesh Gorla. Identification of languages and encodings in a multilingual document. In *Building and Exploring Web Corpora (WAC3-2007): Proceedings of the 3rd Web as Corpus Workshop, Incorporating Cleaneval*, volume 4, page 95. Presses univ. de Louvain, 2007.

[258] Richard Socher, Christopher D Manning, and Andrew Y Ng. Learning continuous phrase representations and syntactic parsing with recursive neural networks. In *Proceedings of the NIPS-2010 deep learning and unsupervised feature learning workshop*, volume 2010, pages 1–9, 2010.

[259] Behrouz Haji Soleimani and Stan Matwin. Spectral word embedding with negative sampling. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[260] Ge Song, Yunming Ye, Xiaolin Du, Xiaohui Huang, and Shifu Bie. Short text classification: A survey. *Journal of Multimedia*, 9(5), 2014.

[261] Karen Spärck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.

[262] Sara Spinelli, Caterina Dinnella, Camilla Masi, Gian Paolo Zoboli, John Prescott, and Erminio Monteleone. Investigating preferred coffee consumption contexts using open-ended questions. *Food Quality and Preference*, 61:63–73, 2017.

[263] Akash Srivastava and Charles Sutton. Autoencoding variational inference for topic models. *arXiv preprint arXiv:1703.01488*, 2017.

[264] Nitish Srivastava, Ruslan Salakhutdinov, and Geoffrey Hinton. Modeling documents with a deep Boltzmann machine. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, UAI'13, pages 616–624, Arlington, Virginia, United States, 2013. AUAI Press.

[265] Massimo Stella, Emilio Ferrara, and Manlio De Domenico. Bots increase exposure to negative and inflammatory content in online social systems. *Proceedings of the National Academy of Sciences*, 115(49):12435–12440, 2018.

[266] Keith Stevens, Philip Kegelmeyer, David Andrzejewski, and David Buttler. Exploring topic coherence over many models and many topics. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 952–961, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics (ACL).

[267] Stefan Stieglitz, Florian Brachten, Davina Berthelé, Mira Schlaus, Chrissoula Venetopoulou, and Daniel Veutgen. Do social bots (still) act different to humans? – Comparing metrics of social bots with those of humans. In *Social Computing and Social Media. Human Behavior*, pages 379–395, Cham, 2017. Springer International Publishing.

[268] Stefan Stieglitz, Florian Brachten, Björn Ross, and Anna-Katharina Jung. Do social bots dream of electric sheep? A categorisation of social media bot accounts. *CoRR*, abs/1710.04044, 2017.

[269] Marija Stupar, Tereza Jurić, and Nikola Ljubešić. Language identification of Web data for building linguistic corpora. In *Proceedings of the 3rd International Conference on The Future of Information Sciences (INFuture 2011)*, pages 365–372, 2011.

[270] V. S. Subrahmanian, Amos Azaria, Skylar Durst, Vadim Kagan, Aram Galstyan, Kristina Lerman, Linhong Zhu, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. The DARPA Twitter bot challenge. *Computer*, 49(6):38–46, June 2016.

[271] Izumi Suzuki, Yoshiki Mikami, Ario Ohsato, and Yoshihide Chubachi. A language and character set determination method based on n-gram statistics. *ACM Transactions on Asian Language Information Processing (TALIP)*, 1(3):269–278, September 2002.

[272] Hidayet Takçi and Ekin Ekinci. Minimal feature set in language identification and finding suitable classification method with it. *Procedia Technology*, 1:444–448, 2012. First World Conference on Innovation and Computer Sciences (INSODE 2011).

[273] Hidayet Takçi and Tunga Güngör. A high performance centroid-based classification approach for language identification. *Pattern Recognition Letters*, 33(16):2077–2084, 2012.

[274] Toshiyuki Takezawa, Genichiro Kikui, Masahide Mizushima, and Eiichiro Sumita. Multilingual spoken language corpus development for communication

research. In *International Journal of Computational Linguistics & Chinese Language Processing, Volume 12, Number 3, September 2007: Special Issue on Invited Papers from ISCSLP 2006*, pages 303–324, September 2007.

[275] Carlos Teixeira, Gabriela Kurtz, Lorenzo Leuck, Pedro Sanvido, Joana Scherer, Roberto Tietzmann, Isabel Manssour, and Milene Silveira. Polls, plans and tweets: An analysis of the candidates' discourses during the 2018 Brazilian presidential election. In *Proceedings of the 20th Annual International Conference on Digital Government Research*, dg.o 2019, pages 439–444, New York, NY, USA, 2019. ACM.

[276] Frederieke ten Kleij and Pieter A.D Musters. Text analysis of open-ended survey responses: A complementary method to preference mapping. *Food Quality and Preference*, 14(1):43–52, 2003.

[277] Mike Thelwall, David Wilkinson, and Sukhvinder Uppal. Data mining emotion in social network communication: Gender differences in MySpace. *Journal of the American Society for Information Science and Technology*, 61(1):190–199, 2010.

[278] Andree Thieltges, Florian Schmidt, and Simon Hegelich. The devil's triangle: Ethical considerations on developing bot detection methods. In *2016 AAAI Spring Symposium Series*, 2016.

[279] Howard E Tinsley and David J Weiss. Interrater reliability and agreement of subjective judgments. *Journal of Counseling Psychology*, 22(4):358, 1975.

[280] Andrija Tomović, Predrag Janičić, and Vlado Kešelj. N-gram-based classification and unsupervised hierarchical clustering of genome sequences. *Computer Methods and Programs in Biomedicine*, 81(2):137–153, 2006.

[281] Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 252–259, 2003.

[282] Dat Tran and Dharmendra Sharma. Markov models for written language identification. In *Proceedings of the 12th international conference on neural information processing*, pages 67–70, 2005.

[283] Gaëtan Tremblay. The information society: From fordism to gatesism: The 1995 Southam lecture. *Canadian Journal of Communication*, 20(4), 1995.

[284] Erik Tromp and Mykola Pechenizkiy. Graph-based n-gram language identication on short texts. In *Proceedings of the 20th Annual Belgian Dutch Conference on Machine Learning (Benelearn 2011)*, Hague, Netherlands, 2011.

[285] Andrew Trotman, Antti Puurula, and Blake Burgess. Improvements to BM25 and language models examined. In *Proceedings of the 2014 Australasian Document Computing Symposium*, ADCS '14, pages 58:58–58:65, New York, NY, USA, 2014. ACM.

[286] Peter D Turney and Patrick Pantel. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37(1):141–188, January 2010.

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

[288] Chris van der Lee and Antal van den Bosch. Exploring lexical and syntactic features for language variety identification. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 190–199, Valencia, Spain, April 2017. Association for Computational Linguistics (ACL).

[289] Laurent Vannini and Hervé Le Crosnier. *Net. Lang: Towards the Multilingual Cyberspace*. C & F Éditions, 2012.

[290] Onur Varol, Emilio Ferrara, Clayton A Davis, Filippo Menczer, and Alessandro Flammini. Online human-bot interactions: Detection, estimation, and characterization. In *Eleventh international AAAI conference on Web and social media*, 2017.

[291] Tommi Vatanen, Jaakko J. Väyrynen, and Sami Virpioja. Language identification of short text segments with n-gram models. In *roceedings of the 7th International Conference on Language Resources and Evaluation (LREC 2010)*, pages 3423–3430, Valletta, Malta, 2010.

[292] Monika Verma and Sanjeev Sofat. Techniques to detect spammers in Twitter - A survey. *International Journal of Computer Applications*, 85(10), 2014.

[293] Tony Vitale. An algorithm for high accuracy name pronunciation by parametric speech synthesizer. *Computational Linguistics*, 17(3):257–276, 1991.

[294] John Vogel and David Tresner-Kirsch. Robust language identification in short, noisy texts: Improvements to LIGA. In *The Third International Workshop on Mining Ubiquitous and Social Environments (MUSE)*, pages 43–50, Bristol, UK, 2012.

[295] Ivan Vulić and Marie-Francine Moens. Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, pages 363–372, New York, NY, USA, 2015. ACM.

[296] Ivan Vulić, Wim De Smet, Jie Tang, and Marie-Francine Moens. Probabilistic topic modeling in multilingual settings: An overview of its methodology and applications. *Information Processing & Management*, 51(1):111–147, 2015.

[297] Ada Wan. Leveraging data-driven methods in word-level language identification for a multilingual Alpine heritage corpus. In *Proceedings of the Workshop on Multilingual and Cross-lingual Methods in NLP*, pages 45–54, San Diego, California, June 2016. Association for Computational Linguistics (ACL).

[298] Alex Hai Wang. Detecting spam bots in online social networking sites: A machine learning approach. In *Data and Applications Security and Privacy XXIV*, pages 335–342, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[299] Fei Wang, Rui Liu, Yuan Zuo, Hui Zhang, He Zhang, and Junjie Wu. Robust word-network topic model for short texts. In *2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 852–856. IEEE, 2016.

[300] Martin Wechsler, Páraic Sheridan, and Peter Schäuble. Multi-language text indexing for Internet retrieval. In *Computer-Assisted Information Searching on Internet*, RIAO '97, pages 217–232, Paris, France, France, 1997.

[301] Xing Wei and W. Bruce Croft. LDA-based document models for ad-hoc retrieval. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, pages 178–185, New York, NY, USA, 2006. ACM.

[302] Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He. TwitterRank: Finding topic-sensitive influential Twitterers. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, WSDM '10, pages 261–270, New York, NY, USA, 2010. ACM.

[303] Gergely Windisch and László Csink. Language identification using global statistics of natural languages. In *Proceedings of the 2nd Romanian-Hungarian Joint Symposium on Applied Computational Intelligence (SACI)*, pages 243–255, 2005.

[304] Jacek Wolkowicz, Zbigniew Kolka, and Vlado Kešelj. N-gram-based approach to composer recognition. *Archives of Acoustics*, 33(1), 2014.

[305] Haibing Wu, Xiaodong Gu, and Yiwei Gu. Balancing between over-weighting and under-weighting in supervised term weighting. *Information Processing & Management*, 53(2):547–557, 2017.

[306] Harry Wu and Gerard Salton. A comparison of search term weighting: Term relevance vs. inverse document frequency. In *Proceedings of the 4th Annual International ACM SIGIR Conference on Information Storage and Retrieval:*

Theoretical Issues in Information Retrieval, SIGIR '81, pages 30–39, New York, NY, USA, 1981. ACM.

[307] Lingfei Wu, Ian En-Hsu Yen, Fangli Xu, Pradeep Ravikumar, and Michael J. Witbrock. D2KE: From distance to kernel and embedding. *ArXiv*, abs/1802.04956, 2018.

[308] Lingfei Wu, Ian En-Hsu Yen, Kun Xu, Fangli Xu, Avinash Balakrishnan, Pin-Yu Chen, Pradeep Ravikumar, and Michael J. Witbrock. Word mover's embedding: From Word2Vec to document embedding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4524–4534, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[309] Tingmin Wu, Sheng Wen, Yang Xiang, and Wanlei Zhou. Twitter spam detection: Survey of new approaches and comparative study. *Computers & Security*, 76:265–284, 2018.

[310] Pengtao Xie and Eric P. Xing. Integrating document clustering and topic modeling. *CoRR*, abs/1309.6874, 2013.

[311] Pengtao Xie, Diyi Yang, and Eric Xing. Incorporating word correlation knowledge into topic modeling. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 725–734, Denver, Colorado, May 2015. Association for Computational Linguistics (ACL).

[312] Xiaohui Yan, Jiafeng Guo, Yanyan Lan, and Xueqi Cheng. A biterm topic model for short texts. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 1445–1456, New York, NY, USA, 2013. ACM.

[313] Zhi Yang, Christo Wilson, Xiao Wang, Tingting Gao, Ben Y Zhao, and Yafei Dai. Uncovering social network sybils in the wild. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 8(1):2, 2014.

[314] George U Yule. *The Statistical Study of Literary Vocabulary*. Cambridge University Press, 1944.

[315] Juglar Díaz Zamora and Adrian Fonseca Bruzón. Tweets language identification using feature weighting — Identificación de idioma en tweets mediante pesado de términos. In *Proceedings of the Tweet Language Identification Workshop 2014 co-located with 30th Conference of the Spanish Society for Natural Language Processing (SEPLN 2014)*, pages 30–34, Girona, Spain, 2014.

[316] Marcos Zampieri. Using bag-of-words to distinguish similar languages: How efficient are they? In *2013 IEEE 14th International Symposium on Computational Intelligence and Informatics (CINTI)*, pages 37–41, November 2013.

[317] Marcos Zampieri, Binyam Gebrekidan Gebre, and Sascha Diwersy. N-gram language models and POS distribution for the identification of Spanish varieties (Ngrammes et traits morphosyntaxiques pour la identification de variétés de l'Espagnol) [in French]. In *Proceedings of TALN 2013 (Volume 2: Short Papers)*, pages 580–587, Les Sables d'Olonne, France, June 2013. ATALA.

[318] Marcos Zampieri, Shervin Malmasi, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, Jörg Tiedemann, Yves Scherrer, and Noëmi Aepli. Findings of the VarDial evaluation campaign 2017. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 1–15, Valencia, Spain, April 2017. Association for Computational Linguistics (ACL).

[319] Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Ahmed Ali, Suwon Shon, James Glass, Yves Scherrer, Tanja Samardžić, Nikola Ljubešić, Jörg Tiedemann, Chris van der Lee, Stefan Grondelaers, Nelleke Oostdijk, Antal van den Bosch, Ritesh Kumar, Bornini Lahiri, and Mayank Jain. Language identification and morphosyntactic tagging: The second VarDial evaluation campaign. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, Santa Fe, USA, 2018.

[320] Duo Zhang, Qiaozhu Mei, and Cheng Xiang Zhai. Cross-lingual latent topic extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1128–1137, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics (ACL).

[321] Jinxue Zhang, Rui Zhang, Yanchao Zhang, and Guanhua Yan. On the impact of social botnets for spam distribution and digital-influence manipulation. In *2013 IEEE Conference on Communications and Network Security (CNS)*, pages 46–54, October 2013.

[322] Yuan Zhang, Jason Riesa, Daniel Gillick, Anton Bakalov, Jason Baldridge, and David Weiss. A fast, compact, accurate model for language identification of codemixed text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 328–337, Brussels, Belgium, October-November 2018. Association for Computational Linguistics (ACL).

[323] Qile Zhu, Zheng Feng, and Xiaolin Li. GraphBTM: Graph enhanced autoencoded variational inference for biterm topic model. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4663–4672, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[324] George Kingsley Zipf. *Human behavior and the principle of least effort*. Addison-Wesley Press, Oxford, England, 1949.

[325] Arkaitz Zubiaga, Inaki San Vicente, Pablo Gamallo, Nora Aranberri, and Aitzol Ezeiza. Overview of TweetLID: Tweet language identification at SEPLN 2014.

In *Proceedings of the Tweet Language Identification Workshop 2014 co-located with 30th Conference of the Spanish Society for Natural Language Processing (SEPLN 2014)*, pages 1–11, Girona, Spain, 2014.

[326] Yuan Zuo, Jichang Zhao, and Ke Xu. Word network topic model: A simple but general solution for short and imbalanced texts. *Knowledge and Information Systems*, 48(2):379–398, August 2016.

Appendix A

Additional results for LID experiments

We present supplementary information for the Chapter 2.

Table A.1: 44 languages corpora token information for large language groups.

		Web Corpus		Bible Corpus	
Language/Variety	**Code**	**Train**	**Test**	**Train**	**Test**
Bulgarian	bg	125,882	42,520	122,607	39,871
Macedonian	mk	81,074	16,966	123,289	35,827
Bosnian	bs	119,717	39,596	119,717	39,596
Croatian	hr	126,263	43,120	54,552	22,608
Serbian	sr	126,888	43,012	117,372	33,324
Slovenian	sl	126,575	42,670	122,568	37,595
Russian	ru	126,480	42,990	246,105	41,311
Ukrainian	uk	111,732	22,255	122,558	41,907
Slovak	sk	127,325	42,839	122,482	40,093
Czech	cs	126,630	42,703	118,041	39,884
Polish	pl	126,618	42,948	117,905	39,305
Portuguese	pt	125,680	42,415	113,526	38,446
Galician	gl	121,716	28,553	121,969	40,884
Spanish	es	126,016	42,594	118,473	39,627
Catalan	ca	123,915	27,525	122,456	41,039
French	fr	125,608	42,229	116,556	38,280
Latin	la	127,385	42,681	-	21,841
Romanian	ro	125,891	42,291	88,269	36,578
Italian	it	127,992	43,559	114,392	38,409
English	en	124,996	42,067	117,661	38,692
German	de	125,839	42,322	121,816	40,053
Luxembourgish	lb	141,208	77,794	141,208	77,794
Frisian	fy	-	-	40,868	40,867
Dutch	nl	125,588	42,257	128,417	40,960
Danish	da	154,486	42,717	121,472	40,837

Table A.2: 44 languages corpora token information for outlier language groups.

		Web Corpus		Bible Corpus	
Language/Variety	**Code**	**Train**	**Test**	**Train**	**Test**
Norwegian	nn	125,619	42,439	139,833	39,828
Swedish	sv	127,233	43,201	114,199	40,661
Finnish	fi	127,586	42,840	121,963	39,938
Estonian	et	68,260	43,276	116,221	39,555
Lithuanian	lt	129,181	43,601	118,406	38,985
Latvian	lv	112,426	21,846	-	35,687
Welsh	cy	129,141	28,639	123,492	57,549
Irish	ga	107,911	20,515	123,037	34,103
Gaelic	gd	31,189	31,175	126,030	36,732
Islandic	is	125,811	42,392	138,509	41,742
Maltese	mt	-	-	101,876	29,639
Basque	eu	110,965	14,413	-	86,844
Hungarian	hu	127,001	42,865	120,944	42,426
Albanian	sq	126,404	42,327	122,396	39,667
Greek	el	125,486	42,353	123,111	40,376
Azari	az	-	-	-	41,524
Turkish	tr	129,939	43,760	118,718	40,117
Armenian	hy	103,421	16,316	123,901	41,390
Georgian	ka	78,267	15,715	109,855	46,800

Table A.3: Detailed result on Linear SVM with features weighted by mutual information on TweetLID'14 dataset gold test data and comparison to the workshop best results.

Language	**P**	**R**	**F1**
pt	95.07	89.37	92.13
eu	96.11	71.93	82.28
ca	86.30	87.22	86.76
es	95.24	93.43	94.33
en	82.42	77.25	79.75
gl	63.59	51.72	57.04
amb	100.00	65.59	79.22
und	35.10	60.20	45.05
Global	81.84	74.59	**77.07**
TweetLID systems			
ELiRF UPV II	82.5	74.4	75.2
ELiRF UPV I	82.4	73.0	74.5
UB/UPC/URV	77.7	71.9	73.6

Table A.4: Confusion matrix on GDI'18 dataset gold test data.

True \ Predicted	BE	ZH	LU	BS	XY
BE	696	45	71	108	271
ZH	44	685	82	131	233
LU	299	47	432	89	319
BS	41	54	73	850	182
XY	238	102	165	19	266

Table A.5: Confusion matrix on GDI'19 dataset gold test data.

True \ Predicted	BE	ZH	LU	BS
BE	804	98	106	183
ZH	35	845	91	206
LU	381	101	570	124
BS	54	110	96	939

Table A.6: Confusion matrix on ILI'18 dataset gold test data.

True \ Predicted	AWA	BHO	BRA	HIN	MAG
AWA	1114	102	174	98	14
BHO	7	1848	32	98	21
BRA	14	0	2108	14	11
HIN	2	93	13	1725	2
MAG	16	41	42	24	2079

Appendix B

Additional results for CLSA experiments

The effect of language correction was presented on Fig. B.1.

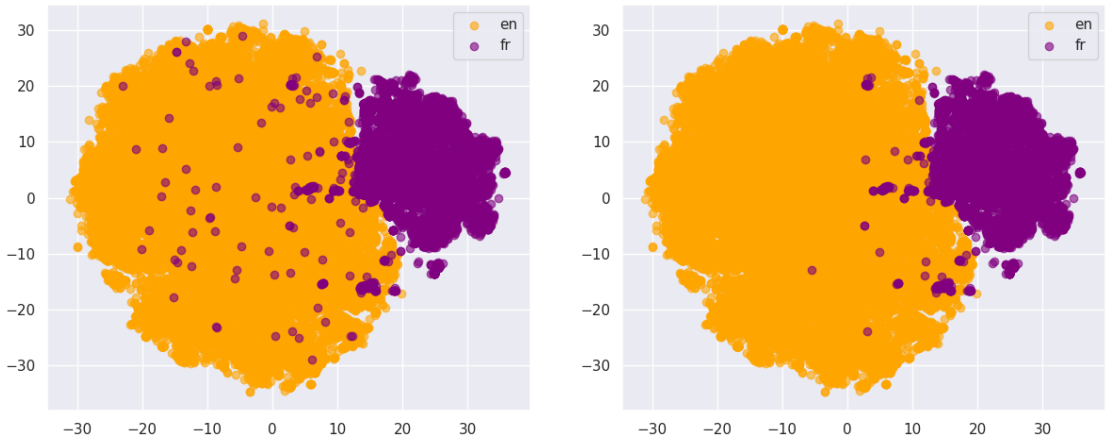

Figure B.1: Language correction in CLSA dataset. Before (left) and after (right). t-SNE visualization.