Automatic Language Identification in Online Multilingual Conversations

Thesis submitted in partial fulfilment of the requirements
for the award of the degree of

Doctor of Philosophy

in

Computer Science and Engineering

by

Neelakshi Sarma

Under the supervision of

Dr. Sanasam Ranbir Singh

&

Prof. Diganta Goswami

Department of Computer Science and Engineering

Indian Institute of Technology Guwahati

Guwahati - 781039 Assam India

भारतीय प्रौद्योगिकी संस्थान गुवाहाटी
Indian Institute of Technology Guwahati

May your choices reflect your hopes, not your fears

-Nelson Mandela (1918-2013)

Dedicated to my beloved parents

and

my dear brother

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisors, Dr. Sanasam Ranbir Singh and Prof. Diganta Goswami for their invaluable support and guidance throughout my Ph.D journey. Their continued encouragement and guidance has been instrumental in my growth both as a researcher and as a person. I am extremely grateful to have had this opportunity to work with them and will always be indebted to them.

I would also like to thank my doctoral committee members - Dr. Vijaya Saradhi, Dr. Ashish Anand and Prof. Priyankoo Sarmah for their insightful comments that added to the quality and clarity of my work. I would also like to thank the heads of the Dept. of CSE during my Ph.D at IITG - Prof. S.V. Rao and Prof. Jatin Deka. I also sincerely thank all the faculty members for their direct and indirect support. I would also like to acknowledge the technical and administrative staff in the Dept. of CSE for their constant help and support.

I would also like to express my sincere thanks to all my friends and seniors at IITG. I am particularly thankful and grateful to my fellow OSINT lab members - Gyanendro, Hemanta, Bornali, Anasua, Durgesh, Akash, Lenin, Rajib Sir, Mala and many more. They have been extremely supporting and encouraging and also made my stay at IITG lively and fun-filled. Last but not the least, my sincere gratitude to my family - my parents and my brother for being my pillars of strength always !

Neelakshi Sarma

Declaration

I certify that

- The work contained in this thesis is original and has been done by myself and under the general supervision of my supervisor(s).
- The work reported herein has not been submitted to any other Institute for any degree or diploma.
- Whenever I have used materials (concepts, ideas, text, expressions, data, graphs, diagrams, theoretical analysis, results, etc.) from other sources, I have given due credit by citing them in the text of the thesis and giving their details in the references. Elaborate sentences used verbatim from published work have been clearly identified and quoted.
- I also affirm that no part of this thesis can be considered plagiarism to the best of my knowledge and understanding and take complete responsibility if any complaint arises.
- I am fully aware that my thesis supervisor(s) are not in a position to check for any possible instance of plagiarism within this submitted work.

Neelakshi Sarma

Certificate

This is to certify that this thesis entitled "**Automatic Language Identification in Online Multilingual Conversations**" submitted by **Neelakshi Sarma**, in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy, to the Indian Institute of Technology Guwahati, Assam, India, is a record of the bonafide research work carried out by her under my guidance and supervision at the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, Assam, India. To the best of my knowledge, no part of the work reported in this thesis has been presented for the award of any degree at any other institution.

Thesis Supervisor
Dr. Sanasam Ranbir Singh
Department of CSE, IITG
Guwahati, Assam, India, 781039
ranbir@iitg.ac.in

Thesis Supervisor
Prof. Diganta Goswami
Department of CSE, IITG
Guwahati, Assam, India, 781039
dgoswami@iitg.ac.in

Abstract

With the abundance of multilingual content on the Web, Automatic Language Identification (ALI) is an important pre-requisite for different Natural Language Processing applications. While ALI of well-edited text over a fairly distinct collection of languages may be regarded as a trivial problem, ALI in social media text is considered to be a non-trivial task due to the presence of slang words, misspellings, creative spellings and special elements such as hashtags, user mentions, etc. Additionally, in a multilingual environment, phenomena such as code-mixing and lexical borrowing make the problem even more challenging. Further, the use of the same script to write content in different languages whether due to transliteration or due to shared script between languages imposes additional challenges to language identification. Also, many existing studies in ALI are not suitable for low resource languages due to either of the two reasons. First, the languages may actually lack the resources required like dictionaries, annotated corpus, clean monolingual corpus, etc. Second, the languages may consist of the basic resources in the native scripts, but due to the use of transliterated text, the available resources are rendered useless. Considering the challenges involved, this thesis work aims to address the problem of automatic language identification of code-mixed social media text in transliterated form in a highly multilingual environment. The objective is to use minimal resources so that the proposed techniques can be easily extended to newer languages with fewer resources.

Although the language identification techniques explored in this study are generic in nature and not specific to any languages, to conduct various experimental investigations, this study generates three manually annotated and three automatically annotated language identification datasets. The datasets are generated by collecting code-mixed user-comments from a highly multilingual social media environment. Altogether, the datasets are composed of six languages - Assamese, Bengali, Hindi, English, Karbi and Boro. Apart from dataset generation, this thesis

work makes four important contributions. First, it studies the language characteristics of user conversations in a highly multilingual environment. Interesting observations with regards to language usages and factors influencing language choices in a multilingual environment are obtained from this study. Second, a technique for sentence-level language identification is proposed taking advantage of the social and conversational features in user conversations. The proposed technique outperforms the baseline set-ups and enhances language identification performance in a code-mixed noisy environment. Third, a word-level language identification framework is proposed that makes use of sentence-level language annotations instead of traditionally used word-level language annotations. The proposed method focuses on learning word-level representations by exploiting sentence-level structural properties to build suitable word-level language classifiers. The proposed technique substantially reduces the manual annotation effort required while yielding encouraging performance. Fourth, a word-level language identification technique is proposed that makes use of a dynamic switching mechanism to enhance word-level language identification performance in a highly multilingual environment. The proposed switching mechanism attempts to make the correct choice between two different classification outcomes when one of the outcomes is incorrect. The proposed framework yields better performance than the constituent classifiers trained over a set of non-complementary features. The proposed set-up also outperforms the baseline set-ups using minimum annotated resources and no external resources thus making it suitable for low resource languages.

The various automatic language identification techniques proposed in this study make use of minimal resources. Information obtained from the same set of sentence-level annotated data is used to train both sentence-level as well as word-level classification models. As such, the proposed techniques are also deemed suitable for automatic language identification of low resource languages. The proposed techniques are also able to enhance language identification performance in a code-mixed noisy environment.

Contents

List of Figures

List of Tables

1

Introduction

Automatic Language Identification(ALI) is defined as the task of automatically identifying the language(s) present in a given piece of text, speech or image. A few early works on language identification include language identification by Nakamura [1971] and House and Neuburg [1977] that address automatic language identification over text and speech respectively. Subsequently, ALI has been explored over various domains viz. text (e.g. ALI over news articles (Baldwin and Lui [2010], Tiedemann and Ljubešić [2012]), ALI over Twitter data (Das and Gamback [2014], Zubiaga et al. [2016])), speech (e.g. ALI over spoken utterances (Hategan et al. [2009], Ali et al. [2016])) and images (e.g. ALI over handwritten documents (Hochberg et al. [1999], Zhu et al. [2009])). The scope of this thesis work is limited to ALI over textual content. In particular, this thesis focuses on addressing the challenges associated with ALI over code-mixed social media text in transliterated form in a highly multilingual environment.[1]

Automatic language identification of textual content is an essential pre-requisite for various text processing applications. Most text processing applications like part-of-speech tagging, named entity tagging or machine translation pre-suppose the language of the underlying text and make use of features that depend on the syntax and semantics of the underlying language. Therefore, in a multilingual environment where the incoming documents/text may belong to different languages and the lan-

[1] In this thesis, we use the terms transliteration and phonetic typing interchangeably.

guages are not known in advance, it is of utmost importance to correctly identify the language of the source text before it is fed into the target application. As such, the performance of a language identifier has direct implications on the performance of the target application. For example, the language of the source text wrongly identified in a machine translation system may result in incorrect translation, or the language of a query string wrongly identified in an information retrieval system may result in the retrieval of irrelevant documents.

Automatic language identification has always been relevant to text processing in a multilingual environment. However, in the pre-social media era, automatic language identification has mostly been centered around document level language identification of news articles or web documents. Documents in these domains are mostly characterized by monolingual content in clean and grammatical form. Therefore, language identification over such documents has often been regarded as a trivial task (McNamee [2005]). However, with the growing popularity of social media platforms, there is an outburst of multilingual content on the Web which is characterized by various forms of noise like mis-spellings, creative spellings, slang words and different social media elements like usernames and hashtags, often leading to out-of-vocabulary and word-form ambiguity issues. For example, consider the statements *"Pls share...plsss..plssss..plssssss"* or *"Grt!!! keep dat up!!!"* or *"FYI, the report is already out"*. Here, the words *pls*, *plsss*, *plssss*, *plssssss* are noisy variants of the English word *please*, *Grt* and *dat* are mis-spelled forms of the words *great* and *that* respectively, and *FYI* is an abbreviation for *For your information*. In the ideal scenario, an ALI system should be able to identify the languages of these words correctly, even though it may not have encountered some of these word forms before. Words that an ALI system encounters during run time or execution time but are not seen during the training phase of the system are known as out-of-vocabulary words. In the social media environment, handling out-of-vocabulary words is one of the primary requirements of an ALI system. Again, consider the examples, *"Finally arrived #love#joy#happiness"* and *"Incredible...#joy#mua"*. Here, in the former case, *joy* is an English word expressing a feeling of great happiness, while *joy* in the latter case is a named-entity referring to a make-up artist with the name *Joy*. Word-form ambiguities as these are common in a social media environment and

introduces challenges to ALI systems. Additionally, in a multilingual environment, shared vocabulary across languages, code-mixing and phonetic typing are commonplace. For example, consider *"Hate dhori, paye pori, jaio na go"*, *"take sobai hate kore"* and *"Hate the summers"*. Here, *hate* in the first case, *"Hate dhori, paye pori, jaio na go"*, is a phonetically typed Bengali word, *hate* in *"take sobai hate kore"* is an English word embedded into a Bengali sentence and *hate* in *"Hate the summers"* is an English word in a legitimate English sentence. These examples combined explain the non-trivial nature of the ALI problem as well as the reason that ALI over code-mixed social media content is one of the topical research problems.

Over the past few years, there has been much effort dedicated to addressing automatic language identification of code-mixed social media content. Different approaches starting from simple dictionary based approaches (Nguyen and Doğruöz [2013]) to more complex learning based approaches (Jaech et al. [2016]) have been explored. However, considering the wide range of challenges involved, there still exist numerous issues that require attention. Existing studies, in particular, do not adequately handle naturally embedded borrowed words and shared vocabulary. Further, special attention has not been given to challenges associated with transliterated text, particularly under low resource scenarios. For example, existing studies make use of different resources like dictionaries (Nguyen and Doğruöz [2013]), annotated corpora (Jaech et al. [2016]), monolingual corpora (King and Abney [2013]), transliteration tools (Singh et al. [2018a]) etc. While many languages may actually lack such resources, many others may be equipped with few basic resources like dictionaries and monolingual corpora in the native scripts but are rendered unusable due to the use of transliterated text.

In this study, we specifically focus on language identification over code-mixed social media text in a highly multilingual environment. In particular, this study considers language identification over transliterated text with special emphasis on minimal resource usage such that the proposed techniques are suitable for low resource or resource-scarce languages. Although there are language identification datasets that are publicly available, the majority of them do not fit well with the objective of the study and the proposed techniques. Therefore, to carry out various experimental investigations, we generate two sentence-level and one word-level language

identification dataset as a part of this thesis work. The datasets have been collected from popular social media platform channels in Assam, a state in North-East India. People in Assam speak Assamese, Bengali, Hindi, English and multiple ethnic languages. This multilingual nature is reflected in the social media conversations as well where users resort to conversations in different languages. Further, a large section of the social media communications is phonetically typed using Roman script that adds to the challenges and makes it an ideal choice for exploring the problem of automatic language identification. Altogether, the datasets consist of code-mixed comments in six languages - Assamese, Bengali, Hindi, English, Karbi and Boro. Except English, all other languages can be regarded as low resource languages due to very limited or no availability of publicly available language processing resources, particularly in the transliterated space.

1.1 Background

One of the most popular works in text ALI, also regarded as a benchmark in text ALI is that by Cavnar and Trenkle [1994]. They use character n-gram as features and a rank order metric to compute the similarity between a test document and the language profiles generated from the training documents. They achieve an accuracy of 99.8% over a test set comprising of 3478 documents in eight languages. Various other studies also report near perfect accuracy (close to 100%) (Grefenstette [1995], Yang and Liang [2010]). Therefore, McNamee [2005] argue that with simplest of the techniques achieving an accuracy of close to 100%, ALI is probably a solved problem.

However, with evolution in online content, there is also modification in the nature of the underlying problem. Although the monolingual assumption is a common and valid assumption in most early studies involving ALI, over time, taking into consideration the existence of multilingual documents started to become important. Few early studies addressing multilingual language identification include studies by Prager [1999], Lui et al. [2014] and Jauhiainen et al. [2015]. Further, the popularity of various easy to use social media platforms like Facebook, YouTube, Twitter etc. led to a complete transformation in the nature of the online text from clean and reg-

ular text to short, irregular, informal and multilingual text. With phonetic typing (transliterated text), homophonic confusions, creative spellings and code-mixing becoming common, effective solutions for ALI problem on social media content started to be regarded as a non-trivial problem. As such, various studies are dedicated to explore the problem of ALI over code-mixed social media content at various levels of granularities:

- **Sentence-level Language Identification**: Sentence-level language identification finds the language corresponding to a given sentence. The general assumption in sentence-level language identification is that the entire sentence is written in a single language. However, this is too rigid an assumption in the social media domain where code-mixing is a common phenomenon. In such scenarios, sentence-level language identification corresponds to identifying the underlying language sense of the sentence. For example, the sentence *Movie achi thi [Movie was good]* is regarded as a Hindi sentence even though it consists of the English word movie embedded into it.

- **Word-level Language Identification**: As the name suggests, word-level language identification corresponds to finding the language corresponding to each word in a sentence/document. When sentences are composed of words in different languages, like in social media posts, it is necessary to identify languages at the word level to extract complete information from the text. For example, consider the sentence *Tumake sobai hate kore [Everybody hates you]*. This is a phonetically typed Bengali sentence with the English word *hate* embedded in it. For an application like sentiment analysis, it is important to identify that the word *hate* is an English word. As such, accurately identifying the language corresponding to each word in a code-mixed sentence is crucial for the performance of the downstream applications.

- **Subword-level Language Identification**: Particularly in the social media domain, a quite often observed phenomenon is that of fusing together the root word of one language with an affix of another language. For example, *Ami classe achi [I am in the class]*, the word *classe* is formed by fusing an English root word *class* with a Bengali suffix *e*. To extract complete information from

such text, it is important to identify the languages at the subword-level. This is known as subword-level language identification where the objective is to identify the language corresponding to different sub-word units.

The approaches that have been explored for automatic language identification of code-mixed social media text include a variety of approaches starting from simple dictionary based (Nguyen and Doğruöz [2013]) to various supervised and unsupervised learning approaches using various features that include character features (Jaech et al. [2016]), word features (Barman et al. [2014]), context features (Barman et al. [2014]), non-textual features like POS tags (Bestgen [2017]) and social conversational features (Carter et al. [2013]). A detailed discussion on the existing studies is presented in Chapter 2.

1.2 Challenges

With an introduction to ALI and it's important aspects in the previous sections, we now focus on some of the important challenges associated with ALI, particularly in social media environment.

- **Noisy text**: ALI over clean regular text is considered a reasonably simple problem. However, the noisy form of text in social media platforms introduces various challenges to language identification. Irregular and informal word forms lead to lexical variants leading to challenges in the form of out-of-vocabulary words and ambiguous word forms. Another important characteristic of text in the social media domain is short text. Shorter text implies lesser contextual information which in-turn implies more challenges in automatic language identification.

- **Shared vocabulary**: In a multilingual environment, if languages in the text collection are fairly distinct, ALI is a reasonably simple problem. However, shared vocabulary across languages aggravates the challenges associated with automatic language identification. Shared vocabulary could either be due to inherent similarities between languages or could be the result of other factors such as irregular or creative spellings, acronyms, etc.

- **Code-Mixing**: Mixing of words or text fragments in different languages within the same discourse, also known as code-mixing, is a common phenomenon in a multilingual social media environment. Contextual information is regarded as one of the useful cues for identifying the languages in any given text. Code-mixing results in disruption of contextual information. Combined with noisy and ambiguous word forms, code-mixing imposes one of the major challenges in automatic language identification over social media text.

- **Transliteration**: Use of transliterated text or phonetic typing is yet another common phenomenon in social media and introduces multiple challenges. First, transliteration increases word form ambiguities. A legitimate word in one language can be confused with the transliterated form of a word in another language. This is further aggravated by issues such as code-mixing and irregular vocabulary. Second, resources for most languages are not available in transliterated form. The use of transliterated text for posting content renders the use of resources available in native scripts unusable in the absence of efficient transliteration tools.

- **Resource Challenges**: As we will see in Chapter 2, ALI makes use of different resources like dictionaries (Das and Gamback [2014]), annotated resources (Jaech et al. [2016]), monolingual corpora (King and Abney [2013]), transliterators (Singh et al. [2018a]), etc. However, not all languages are equipped with these resources. Generating these resources is an expensive and tedious task. While in many cases, resources may not actually be available, in many others, resources may be available in native scripts but due to the use of transliterated text, the available resources cannot be used.

1.3 Research Objective

The objective of this study is to *explore ALI for code-mixed social media text in a highly multilingual environment considering transliterated text.* We explore sentence-level and word-level language identification in this study. Subword-level language identification is not within the scope of this study. Specifically, this thesis work aims to address the following:

- In a highly multilingual society, code-mixing without affecting the underlying language sense has become a natural phenomenon. Existing studies exhibit shortfalls in accurately identifying the languages in such a dynamic environment. For example, consider the sentence *job contractual ne permanent [Is the job contractual or permanent]*. This sentence is an Assamese sentence even though three out of four words in the sentence are in English. While the embedded English words in the Assamese sentence can mislead the sentence-level language identification algorithms to predict the given sentence as English, this can also mislead a word-level language identification technique to predict the word *ne [or]* as an English word since all words in the context are English. Similarly, consider the sentence *Ami khub enjoy korisu [We are enjoying a lot]*. This is an Assamese sentence with the English word *enjoy* embedded into it. Again, the words in the context being Assamese, the word *enjoy* may be mislabeled as Assamese by a word language identification technique. Therefore, an objective in the current study is to *explore sentence-level and word-level language identification techniques that are robust to the various phenomena of multilingual social media environment such as naturally embedded borrowed words and shared vocabulary.*

- Obtaining large scale annotated data for training supervised models is an expensive task in terms of time and effort. Annotation of code-mixed data is more difficult because the annotator needs to be well versed in all the languages involved. Further, word-level annotation being sequential in nature is far more expensive than sentence-level annotation. The effort involved in obtaining annotated word sequences is proportional to the length of the sequence. Therefore, another objective of the current study is to *explore language identification techniques with reduced manual annotation effort such that the techniques can be easily adapted to new languages where annotated resources do not already exist.*

- In addition to annotated resources, existing studies also make use of external resources such as dictionaries, monolingual corpora, transliteration tools etc. However, it is difficult to obtain such resources for transliterated text. There-

fore, the objective in this study is to *explore how well a language identifier can perform with minimal annotated resources and no external resources.*

With the above stated objectives in mind, we make four major contributions with respect to automatic language identification over code-mixed social media text that are summarized below. Subsequent chapters discuss these contributions in detail.

1.4 Contributions

Summarized below are the contributions made in this thesis work:

- **Datasets**: We generate three manually annotated language identification datasets - two sentence-level language identification datasets - SLI-MI and SLI-MII and one word-level language identification dataset - WLI-M as a part of this thesis work. Additionally, two automatically annotated sentence-level language identification datasets - SLI-AI and SLI-AII and one automatically annotated word level language identification dataset - WLI-A has also been generated. These datasets are discussed in detail in Chapter 3. The datasets consist of code-mixed social media comments collected from a highly multilingual environment. The comments in the datasets are characterized by various forms of noise induced by different social media phenomena such as code-mixing, transliteration and shared vocabulary across languages that make ALI over social media data a challenging task.

- **Investigating Language Characteristics of Users in Multilingual Social Media Conversations**: We systematically investigate the language characteristics of user conversations in a multilingual social media environment. Considering the SLI-AI and WLI-A datasets, we investigate the language usage patterns of multilingual users and investigate the factors that are likely to affect the language choices of a multilingual user. Information as these are useful in various computational as well as linguistic and sociological analyses.

- **Sentence-level Language Identification using Social Conversational Features**: We propose a sentence-level language identification technique for

code-mixed social media text in a highly multilingual environment. The proposed technique makes use of different social and conversational factors to obtain improved language identification performance despite code-mix and code-similarity. The proposed techniques are trained and evaluated using the SLI-MI and SLI-MII datasets. Additionally, a Twitter dataset (Zubiaga et al. [2016]) available in the public domain has also been used in this study.

- **Word-level Language Identification of Code-Mixed Social Media Content**: Sentence-level language identification is often not sufficient in a code-mixed environment. Word-level language identification is required to extract complete information from the given text. Contrary to existing studies that use different resources for word-level language identification, we propose two word-level language identification techniques using minimal resources such that the proposed techniques can be easily extended to newer languages with fewer resources. The SLI-MI and WLI-M datasets are used to train and evaluate the proposed techniques.

  - **Obtain word-level language information using sentence-level language information**: Sentence-level language annotation is less expensive than word-level language annotation which is a sequential task. In this study, we propose techniques for obtaining word-level language information using sentence-level language information. The proposed techniques do not make use of word-level annotated data or other resources such as dictionaries and monolingual corpora.

  - **Word-level language identification using a dynamic switching mechanism**: In this study, we propose yet another word-level language identification technique that makes use of a dynamic switching mechanism to make the correct choice between two different classification outcomes when one of the classification outcomes is incorrect. This has been shown to be useful in correctly determining languages of words that are borrowed or embedded from other languages as well as languages of words that are valid across multiple languages.

1.5 Organization of the Thesis

The remaining part of the thesis is organized as follows.

1. **Chapter 2 Literature Survey**: In this chapter, we discuss in detail the available literature in ALI. The literature review is broadly divided into (i) ALI over regular text and (ii) ALI over social media text. Different problems of interest in each of these domains are pointed out and the literature surrounding those problems are discussed.

2. **Chapter 3 Datasets**: This chapter describes the different datasets that have been used in this thesis work. It discusses in detail the manually and automatically annotated sentence-level language identification datasets and word-level language identification datasets that have been generated as a part of this thesis work.

3. **Chapter 4 Language Characteristics of Users in Multilingual Social Media Conversations**: In this chapter, we discuss our findings of user language characteristics in a highly multilingual social media environment. What are the language choices of a user in a multilingual environment and what are some of the factors that can influence these choices form a major component of this study.

4. **Chapter 5 Sentence-level Language Identification of Code-Mixed Social Media Content**: This chapter discusses the proposed sentence-level ALI framework and the performances obtained using the proposed techniques in comparison to the baseline setups over various datasets.

5. **Chapter 6 Word-Level Language Identification of Code-Mixed Sentences** : This chapter discusses the two proposed word-level language identification techniques that use sentence-level language annotations to obtain word-level language information. The performances obtained therein and the important observations are discussed that also pave the way for further improvements.

6. **Chapter 7 SwitchNet : Learning to Switch for Word-Level Language**

Identification : In this chapter, we discuss yet another word-level ALI framework that has been proposed in this thesis work. The objective of the proposed framework is to make a correct choice between the outputs of two different classifiers when one of the outputs is incorrect.

7. **Chapter 8 Conclusion and Future Work** : In this chapter, we present our concluding remarks on the thesis work and some of the potential directions to work on in future.

2

Literature Survey

This chapter presents a review of existing literature related to automatic language identification. Although automatic language identification is applicable to input across different modalities (text, speech and images), the scope of discussion in this chapter is limited to the text domain. As briefly discussed in Chapter 1, most early studies in ALI focus on ALI over regular text. Statistical and classical machine learning techniques like Naive Bayes, Support Vector Machines and Decision Tress are some of the widely used methods. Over the years, with the growing popularity of various social media platforms, the domain of interest in ALI shifted from regular text to social media text. ALI over social media text has been investigated at different granularities viz. sentence-level, word-level and subword-level. In addition to traditional methods like dictionary look-ups, statistical models and classical machine learning models, neural network based architectures have been widely explored in recent times. One of the early studies using neural networks for ALI over text is that by MacNamara et al. [1998]. They use a Recurrent Neural Network (RNN) based architecture for ALI of library catalogue entries. However, they observe that a statistical model based on trigram statistics outperform the proposed RNN based architecture. However, in contrast to the observation by MacNamara et al. [1998], in the social media domain, neural network based architectures have shown promising performances and have outperformed traditional baselines. In this chapter, we discuss in detail the existing studies in ALI over text. The discussion is divided into

two major parts. First, ALI over the regular text domain is discussed. In the next sections, ALI over social media text is discussed.

2.1 ALI in regular text domain

One of the early and most popular approaches in text ALI is the character n-gram approach proposed by Cavnar and Trenkle [1994]. They achieve 99.8% accuracy over a corpus of newsgroup articles in 8 different languages. Their approach is based on the n-gram profiles of languages. Language profile for each language is created by considering the top k frequent character n-grams from the training documents. The language of the test documents are then determined by comparing the language profile of the test documents with that of each language using a rank order metric. Prior to Cavnar and Trenkle [1994], rule based language classification have been explored by Henrich [1989]. However, the objective there is to identify languages of foreign words in German documents. Unlike Henrich [1989], most language identification studies over regular text address document level language identification rather than sentence-level or word-level language identification. Martins and Silva [2005] combine the n-gram based approach of Cavnar and Trenkle [1994] with heuristics to classify languages of Web documents. Grefenstette [1995] also explore two language identification techniques - a trigram based technique based on the most frequent trigrams and a short word technique based on tokens of five or less characters. They observe an accuracy of close to 100% over sentences comprising of more than 15 words. Among the machine learning models, Naive Bayes models have been frequently used for language identification (Peng and Schuurmans, Baldwin and Lui [2010]). SVM and Naive Bayes with n-grams features for ALI over Web documents have been used by Baldwin and Lui [2010].

The works discussed so far are based on supervised techniques i.e., documents annotated with the corresponding language information are used for training. However, unsupervised approaches have also been explored. Instead of using training documents to identify discriminating terms, Mather [1998] use singular value decomposition over a matrix of word frequencies. The discriminating terms are used to obtain groups of documents belonging to the same language. However, they do

not compare their results with existing studies to evaluate the performance of their method with the existing supervised methods. Unsupervised ALI has also been investigated by Shiells and Pham [2010], Selamat and Ching [2008] and Amine et al. [2010].

Nevertheless, with many approaches achieving near perfect accuracy (Cavnar and Trenkle [1994], Grefenstette [1995]), McNamee [2005] labeled automatic language identification as a solved problem. However, different investigations reveal that such an assumption is only true under a set of tightly set constraints on the nature of the documents (long, clean , monolingual), nature of the languages (dissimilar languages with distinct vocabulary set) and the number of languages. However, when these assumptions do not hold true, there are indeed different aspects of ALI that require special attention for achieving acceptable performances in real world applications. Some of these are discussed below:

2.1.1 ALI for discriminating similar languages

When languages in the document collection are fairly distinct, ALI is often regarded as a trivial problem. However, when the languages involved are very similar in terms of their vocabulary, ALI is a comparatively challenging task. Therefore, different studies have specifically focused on exploring ALI over a set of similar languages or dialects. In this regard, Ljubesic et al. work on distinguishing between Serbian, Croatian and Slovenian. Over a collection of documents collected from a news portal, Slovenian is discriminated from the rest with 100% accuracy using a character based second order Markov model while Croatian and Serbian are identified with an accuracy of 96%. Further improvement upto 99% is obtained using a list of forbidden words. Tiedemann and Ljubešić [2012] obtain an accuracy of 90.3% over a collection of Bosnian, Serbian and Croatian documents using the second order model used in Ljubesic et al.. On the other hand, they achieve an accuracy of 95.7% and 97% using a Naive Bayes classifier and a weighted blacklisted token list respectively. However, given that the features for the proposed Naive Bayes model and the weighted blacklist model are obtained from a parallel corpora, the performance of these methods in the absence of parallel corpora has not been explored. Their observation also conclude that discriminating between similar languages re-

quire special attention. Other works dealing with discriminating between similar languages include that by Ranaivo-Malançon [2006] that discriminate between Indonesian and Malay language, Zampieri and Gebre [2012] that discriminate between two varieties of Portuguese language, Zampieri et al. [2016] that also discriminate between three different varieties of Portuguese, Lui and Cook [2013] that discriminate between Australian, Canadian and British English, Zečević and Vujicic-Stankovic [2013] that work on discriminating between variants of Serbian languages, van der Lee and van den Bosch [2017] that work on discriminating between variants of the Dutch language, Hollenstein and Aepli [2015] that work on discriminating between German dialects, Huang and Lee [2008] that work on three different varieties of Chinese, Zampieri et al. [2013] that work on varieties of Spanish and Ciobanu and Dinu [2016] that work on variations in Roman. Experiments on discriminating between several combinations of similar languages have also been explored by Tan et al. [2014] and King et al. [2014]. Clematide and Makarov [2017] explore three different models - Naive Bayes, Conditional Random Field and Support Vector Machine for classification of Swiss-German dialects. Rangel et al. [2016] propose a technique called Low Dimensionality Representation that uses different term weighting mechanisms to represent the probability with which a term belongs to different language varieties. They use the proposed technique to identify different variations of Spanish, Portuguese and English languages. Bestgen [2017] use a SVM classifier with character ngrams and POS tag ngrams weighted by BM-25 scheme as features for identifying language varieties over six different language groups.

2.1.2 ALI for large number of languages

An important aspect of automatic language classification is also the number of languages considered. While ALI over a closed set of small number of languages can in general be expected to achieve a significantly good performance, ALI considering large number of languages is a challenging task. In this line, Vatanen et al. [2010] evaluate the performance of two different language identification techniques over a test set comprising 281 languages and samples of size 5-21 characters. Results obtained show two important observations. First, language identification over short text is more challenging than regular text and require special attention. Second, the

character language model with smoothing techniques such as absolute discounting and Katz outperform the ranking based method of Cavnar and Trenkle [1994]. In the same line, Brown [2012] perform an evaluation over a test set comprising of 923 languages. Their evaluation is based on an n-gram based language identification technique. They observe that over a test set comprising of a large number of languages, identifying n-grams that are not valid in a particular language helps in significantly reducing the error rate. Other studies that perform similar evaluations over large number of studies include that of Brown [2013] that consider 1100 languages for evaluation, Brown [2014] that consider 1366 languages and Jauhiainen et al. [2017] that consider 285 languages for evaluation.

2.1.3 Multilingual ALI

The monolingual assumption is a convenient assumption for language identification. However, considering the multilingual nature of documents present in the Web, addressing multilingual ALI is important. The importance of multilingual language identification has also been emphasized upon by Hughes et al. [2006] in their study. One of the early studies that address multilingual language identification is that by Prager [1999]. His study considers the case of bilingual documents and uses a vector space based language classifier to identify bilingual documents and determine the proportion of the languages in the bilingual documents. Cowie et al. [1999] and Pethö and Mózes [2014] also address multilingual language identification by identifying segments of different languages within a document. Sliding window based technique for language identification of multilingual documents is used by Mandl et al. [2006] and Jauhiainen et al. [2015]. While Mandl et al. [2006] used a sliding window over a fixed number of words, Jauhiainen et al. [2015] use a sliding window over bytes. Lui et al. [2014] use generative models to determine the proportion of different languages in a multilingual document They however do not address segmentation of the multilingual documents into segments of different languages. Bidirectional RNN is used for multilingual ALI by Kocmi and Bojar [2017]. Unsupervised multilingual language identification has also been investigated by Biemann and Teresniak [2005].

2.2 ALI in the social media domain

Social media data is inherently known to be noisy and imposes challenges on any form of text processing including language identification. Carter et al. [2013] systematically demonstrate that automatic language identification over social media text is more challenging than that over regular text. They show that techniques that achieve near perfect accuracy over regular text exhibit a drastic drop in performance over the social media domain. In contrary to regular text domain where automatic language identification is mostly concerned with identifying languages over the document level, in the social media domain, owing to factors such as code-mixing and lexical borrowing, language identification need to be addressed at different levels of granularities like sentence-level ALI, word-level ALI and subword-level ALI. These are discussed below in details.

2.2.1 Sentence-Level ALI

Sentence-level ALI, as explained in Chapter 1, is the identification of language corresponding to each sentence. The general assumption in sentence-level language identification is that the sentences are made up of words all in the same language. However, in the social media domain, code-mixed sentences are a commonplace. In such situations, sentence-level language identification is concerned with identifying the language corresponding to the underlying sense of the language.

Carter et al. [2013] show that the performance of the language identifiers decreases from 99.4% in formal text to 92.4% in microblog text using the n-gram based approach proposed by Cavnar and Trenkle [1994]. Apart from the well known n-gram approaches, recent studies in ALI over social media text use graphical models (Tromp and Pechenizkiy [2011], Vogel and Tresner-Kirsch [2012]) and neural networks (Kocmi and Bojar [2017]). In a slightly different direction, Goldszmidt et al. [2013] attempt to adapt systems trained over regular text to social media text. They augment an ALI system initially trained with Wikipedia articles with Twitter data extracted using location information of the tweets. In contrast to the above studies that make use of only the textual content for language identification, information such as microblog characteristics and user profile information have also

been considered. An improvement of 5.5% is observed by Carter et al. [2013] after considering five microblog characteristics - blogger characteristic, link characteristic, mention characteristic, tag characteristics and conversation characteristic. Further Bergsma et al. [2012] explore meta features such as user name, screen name and self reported user location along with n-grams features, and observe an improvement in language classification performance using Logistic Regression. Similar attempts have also been made by Wang et al. [2015] for language identification over short chats in mobile games.

2.2.2 Word-level ALI

Code-mixing or mixing words from different languages within the same sentence necessitates identifying languages of individual words in order to extract complete information from a given text. Appropriately identifying languages of individual words in a code-mixed piece of text is of primary importance for the downstream text processing applications. In this section, we discuss the different word-level language identification techniques that have been explored in literature.

Dictionary Based Approaches

Dictionary based approach for language identification have been used by Barman et al. [2014] where the language of the word is classified based on its frequency of occurrence in multiple language dictionaries. Dictionary look-ups for language identification have also been used by Nguyen and Doğruöz [2013]. However considering the variations of word forms in social media content and unavailability of dictionaries for transliterated text, dictionary look-up is not an efficient approach for language identification in terms of both coverage as well as performance.

Supervised Approaches

Use of code-mixed data annotated at the word-level with the corresponding language information to train word-level classifiers is one of the most commonly used approaches. Classification based approaches like SVM, Naive Bayes and sequence classification based approach like CRF have been used by Barman et al. [2014] and Gundapu et al. [2018] to train word-level classifiers. CRF based approaches

have also been used by Nguyen and Doğruöz [2013], Chittaranjan et al. [2014], Xia [2016], Sikdar and Gambäck [2016] and Yirmibeşoğlu and Eryiğit [2018]. CRF classifier combined with post processing heuristics have been used by Banerjee et al. [2014]. In addition to addressing word language identification , Das and Gamback [2014] also introduce a metric called Code-Mixing Index that indicates the level of mixing between different languages in a given text. Vyas et al. [2014] also use CRF based classifier for word-level language identification where they also address other tasks like back-transliteration, normalization and part-of-speech tagging. Word-level language identification and prediction of code-switching points have also been addressed by authors in Piergallini et al. [2016]. Sequence classification using RNN have been used for word-level language identification by Samih et al. [2016]. Multi-channel CNN combined with a Bi-LSTM-CRF module have been used by Mandal and Singh [2018]. Jurgens et al. [2017] explore sequence to sequence models for word-level language identification . Instead of using sequence models, Zhang et al. [2018] propose a two-stage model wherein in the first stage, a distribution over the languages for given word is predicted using a feed forward neural model. This is followed by a decoding stage which along with the language distribution predicted in the first stage also takes into consideration global constrains over the entire sentence.

Most of the studies discussed above make use of only word-level information for language identification. However, character level information have also been found useful. Capturing character level information are associated with two advantages. First, they can be useful in handling unseen word variations. Second, it is most likely that character sequences of different languages will have different structural properties which can be used to disambiguate languages of words. Character embeddings and subword unit information have been used to capture this information by Jaech et al. [2016] and Mave et al. [2018]. In a slightly different approach, Singh et al. [2018b] use a transliteration model to transliterate romanized script to Devnagiri script. They use RNNs to train a character language model for each language. Given an annotated corpus, the output of the language models is combined with other features to train a word-level language classifier. In addition to character n-gram information, phonetic information have also been used for word language identification by Das et al. [2019]. Word-level language identification is also ad-

dressed by Dongen [2017], Giwa and Davel [2013], Eskander et al. [2014] and Shiells and Pham [2010].

Weakly supervised and unsupervised approaches

A drawback of using word-level annotated code-mixed data for training classification models is that they are expensive to obtain. Considering variations in word forms in social media text owing to factors like code-mixing, phonetic typing and spelling variations, in absence of large-scale annotated data, it may not be possible to capture word and language variations in a highly multilingual environment. Therefore, in an effort to reduce the dependence on annotated data, classification using monolingual corpus in respective languages have been proposed. King and Abney [2013] use a collection of monolingual texts and employ weakly supervised and semi-supervised methods for word-level language identification in multilingual documents. Rijhwani et al. [2017] also use unsupervised model using monolingual corpora and HMM for word-level language identification. Instead of using real code-mixed datasets, Gella et al. [2014] also create a synthetic code-mixed language identification dataset from a collection of monolingual text. However, a drawback associated with using monolingual corpus for word language classification is that it is not easy to obtain clean monolingual corpus for transliterated text. Considering that a large section of Indian social media conversations are in transliterated form, these approaches may not be feasible in many scenarios.

A number of shared tasks (Solorio et al. [2014], Molina et al. [2016]) have also been organized to address word-level language identification problems. Apart from word-level language identification, other closely related studies include analyzing different aspects of code-switched data (Rudra et al. [2019]), prediction of code-switching points from multilingual communications (Papalexakis et al. [2014]), predicting foreign language usage in social media posts (Volkova et al. [2018]), language informed modeling of code-mixed data (Chandu et al. [2018]), predicting the presence of matrix language (Bullock et al. [2018]) and predicting likeliness of word borrowing in social media (Patro et al. [2017]). Apart from language identification of short social media text, ALI have also been explored over other forms of short text such as search engine queries (Ceylan and Kim [2009]), proper names (Hakkinen

and Tian [2001], Konstantopoulos [2007]) etc.

2.2.3 Subword-level ALI

A lesser explored area in ALI is subword-level ALI. Subword-level language identification is identification of languages within a word. It is important when a word may be made up of morphological units from different language. For example in the sentence *Ami classe achi [I am in class]*, the word *classe [in class]* is obtained by combining the root word *class* which is in English with a suffix **e** which is in Bengali. In scenarios as such, subword language identification is important to extract complete information from the given text. Although comparatively lesser explored than inter-sentential code-switching or intra-sentential code-switching, Stefanich et al. [2019] argue that intra-word code-switching is not an infrequent phenomenon among multilinguals. Stefanich et al. [2019] also further examine the morphological and phonological patterns involved in intra-word code-switching. They observe that while the tendency to combine two morphological units from two different languages (e.g. root word from one language and affix from another language) seems to come naturally to bilinguals or multilinguals, the tendency to combine different units from different phonological systems does not seem to come naturally. Although Stefanich et al. [2019] examine the linguistic aspects of intra-word code-switching, from language identification standpoint, not much efforts have been directed towards subword language identification. Nguyen and Cornips [2016] address automatic detection of intra-word code-switching using a combination of morphological analysis and language identification. Though not addressing subword language identification explicitly, Barman et al. [2014] use a combination of k-nearest neighbor and linear SVM to identify what they define as *mixed* words, i.e., words that are formed by combining morphological units from different languages. Another study somewhat similar to identifying intra-word code-switching is by Yirmibeşoğlu and Eryiğit [2018] where they use a rule based technique to identify intra-word code-switching while addressing code-switching between Turkish-English language pairs. A more extensive study in subword-level language identification have been done by Mager et al. [2019]. They explore two different datasets - a Spanish–Wixarika dataset and a Turkish-German dataset. They observe that in comparison to different language

pairs that are explored in the ALI literature considering code-switching, the Spanish–Wixarika language pair consists of significantly large proportion of intra-word code-switches to the extend that it cannot be ignored.

2.3 Summary

This chapter presents a review of existing literature in the field of ALI. The relevant studies with respect to ALI over regular text and ALI over social media text are summarized in Tables 2.1 and 2.2 respectively. In the current times, ALI over social media text is of primary importance. Various methods of ALI over social media text at different levels of granularities have been explored. The features used include text features (e.g. character level features, Word-level features, sentence-level features), non-textual features such as POS tags as well as social and conversational features (user features, link features, conversation features etc.). However, the existing approaches still exhibit limitations in terms of their ability to handle naturally embedded borrowed words and ambiguous word forms commonly prevalent in social media content. Additionally, the existing approaches also make use of different resources such as annotated data, monolingual corpora, dictionaries, transliteration tools etc which are not easily obtainable for low resource languages. Hughes et al. [2006] consider ALI for low resource languages and transliterated text to be among the primary challenges for ALI. Over the years, though significant efforts have been dedicated for ALI over social media text, the above stated limitations still exist. Therefore, in this thesis work, we focus on automatic language identification for code-mixed social media text in transliterated form considering low resource scenario.

Objective	Year	Algorithms/Techniques	Paper(s)
Monolingual ALI	1994-1998	Statistical and Rule based	Cavnar and Trenkle [1994], Grefenstette [1995], Mather [1998]
		Neural Networks	MacNamara et al. [1998]
	2003-2014	Statistical	Brown [2012, 2013, 2014]
		Supervised Machine Learning (Naive Bayes, Nearest Neighbour, Support Vector Machines)	Peng and Schuurmans, Baldwin and Lui [2010], Giwa and Davel [2013]
		Unsupervised Machine Learning	Amine et al. [2010], Shiells and Pham [2010]
		Hybrid(Machine Learning+Rules)	Tiedemann and Ljubešić [2012]
Multilingual ALI	1999	Statistical	Prager [1999], Cowie et al. [1999]
	2005-2010	Randomized Graph clustering	Biemann and Teresniak [2005]
		Statistical	Martins and Silva [2005], McNamee [2005], Mandl et al. [2006]
		Statistical+Wikipedia information	Yang and Liang [2010]
	2014-2015	Statistical	Pethö and Mózes [2014], Jauhiainen et al. [2015]
		Machine Learning (Probabilistic Mixture Models)	Lui et al. [2014]
ALI of closely related languages	2006-2008	Statistical	Huang and Lee [2008]
		Statistical+Rule based	Ranaivo-Malançon [2006], Ljubesic et al.
		Neural Networks	Selamat and Ching [2008]
	2012-2016	Statistical	Zampieri and Gebre [2012], Zampieri et al. [2013]
		Machine Learning (Naive Bayes, Logistic Regression, Support Vector Machines)	King et al. [2014], Tan et al. [2014], Ali et al. [2016], Zampieri et al. [2016]

Table 2.1: Table showing relevant studies with respect to ALI over regular text

Objective	Year	Algorithms/Techniques	Paper
Sentence-Level ALI	2011-2016	Graph Based	Tromp and Pechenizkiy [2011], Vogel and Tresner-Kirsch [2012]
		Machine Learning (Using text features)	Bergsma et al. [2012], Zubiaga et al. [2016]
		Machine Learning(using conversational features)	Carter et al. [2013], Wang et al. [2015]
Word-Level ALI	2009-2015	Dictionary	Nguyen and Doğruöz [2013], Barman et al. [2014], Das and Gamback [2014]
		Statistical	Hategan et al. [2009], Vatanen et al. [2010], Nguyen and Doğruöz [2013], Das and Gamback [2014]
		Supervised Machine Learning (Decision Trees, Naive Bayes,Support Vector Machines, Logistic Regression)	Ceylan and Kim [2009], Vatanen et al. [2010], Barman et al. [2014], Das and Gamback [2014], Eskander et al. [2014], Gella et al. [2014], Vyas et al. [2014]
		Semi-supervised and Weakly Supervised Machine Learning	King and Abney [2013], Rijhwani et al. [2017]
		Sequential Machine Learning (CRF)	Barman et al. [2014], Chittaranjan et al. [2014], Mandal et al. [2015], Al-Badrashiny et al. [2015]
		Hybrid (CRF classifier + Heuristics)	Banerjee et al. [2014]
	2016-2019	Machine Learning(SVM, Logistic Regression,Naive Bayes, Random Forest)	Piergallini et al. [2016], Gundapu et al. [2018]
		Sequential Machine Learning(CRF,HMM)	Sikdar and Gambäck [2016], Gundapu et al. [2018], Mave et al. [2018], Yirmibeşoğlu and Eryiğit [2018]
		Neural Networks (CNN,Feed Forward Networks)	Jaech et al. [2016], Mandal and Singh [2018], Mave et al. [2018], Singh et al. [2018a], Zhang et al. [2018]
		Sequential Neural Networks(LSTM)	Samih et al. [2016], Das et al. [2019]
Subword-level ALI	2016	Statistical	Nguyen and Cornips [2016]
	2018	Sequential Machine Learning(CRF)	Yirmibeşoğlu and Eryiğit [2018]
	2019	Sequential Neural Model (RNN)	Mager et al. [2019]

Table 2.2: Table showing relevant studies with respect to ALI over social media text

3

Datasets

In this chapter, we discuss the different datasets that have been used for various experimental investigations carried out as a part of this thesis work. Although there are language identification datasets that are available publicly, the majority of them do not fit well with the objective of the study and the proposed techniques. Therefore, as a part of this thesis work, we generate three manually annotated language identification datasets - two sentence-level language identification datasets and one word-level language identification dataset. We also additionally generate two automatically annotated sentence-level language identification datasets and one automatically annotated word level language identification dataset. The datasets consist of code-mixed social media text collected from a highly multilingual environment. The datasets have been collected considering Assam, a state in the North-Eastern part of India, as the target location. The North-Eastern part of India is known for being one of the most language diverse regions in the world. The datasets altogether consist of six languages - Assamese, Bengali, Hindi, English, Karbi and Boro. To the best of our knowledge, this is also the first attempt to generate publicly available datasets for languages in Assam. The comments comprising the datasets are characterized by various forms of noise like irregular and colloquial word forms, code-mixing and transliteration. Therefore, these datasets can be very useful in estimating the performances of various language identification techniques over a code-mixed noisy environment. In this chapter, we discuss in detail the data

collection and annotation process used for creating the new datasets. We also further discuss the characteristics of the publicly available Twitter dataset that has been used in this thesis work.

3.1 Introduction

The challenges associated with automatic language identification over social media text has been discussed in detail in Chapter 1. The various forms of noise in social media text like mis-spellings and creative spellings, slang words and hashtags, user-mentions etc. make automatic language identification over social media data a challenging task. Additionally, in a multilingual environment, phenomena such as code-mixing, lexical borrowing and transliteration aggravates the challenges associated with automatic language identification. Different techniques have been proposed in literature to address various challenges associated with code-mixed data processing and language identification. Different language identification datasets have also been made available publicly for different experimental investigations. A large number of these datasets are released as part of different shared tasks. A shared task on language identification over code-switched data organized as a part of First Workshop on Computational Approaches to Code Switching[1] released word-level language identification datasets for four language pairs - Nepalese-English, Spanish-English, Mandarin-English and Modern Standard Arabic-Arabic dialects. As a part of tool contest on pos tagging for code-mixed Indian social media[2], datasets in three language pairs - Hindi-English, Bengali-English and Telegu-English from three different social media platforms - Facebook, WhatsApp and Twitter are released. Code-mixed Hindi-English tweets with language information are also made available as part of different studies conducted by Singh et al. [2018a] and Singh et al. [2018b]. Zubiaga et al. [2016] and Carter et al. [2013] on the other hand release datasets comprising sentence or tweet level language information. The dataset by Zubiaga et al. [2016] comprises six languages (Spanish, Portuguese, Catalan,Basque, Galician and English) and that by Carter et al. [2013] comprises five languages (Dutch, English, French, German, and Spanish).

[1] https://mirror.aclweb.org/emnlp2014/workshops/CodeSwitch/call.html
[2] http://www.amitavadas.com/Code-Mixing.html

The goal of this study is to explore sentence-level and word-level language identification over code-mixed social media text in transliterated form in a highly multilingual environment. The goal is also to explore language identification techniques that use minimal resources so that they are also suitable for low resource languages. However, only few of the publicly available datasets appropriately fit with the target objective and the proposed techniques. Therefore, as a part of this thesis work, we locally generate two sentence-level language identification datasets and one word-level language identification dataset. Considering the objectives of this thesis work, we consider the following criteria while identifying the experimental datasets.

- Dataset should be collected from multi-lingual environment where chances of users participating in multi-lingual conversations is high.
- To increase the possibility of code sharing (embedding borrowed words) across languages, conversations should be related to common topics.
- Participants in the conversations should probably belong to a same geographical region (multi-lingual and multi-ethnic community).
- All the conversations should be communicated using common script (roman script in this study).

Considering the above criteria, this study considers Assam, a state in the North-Eastern part of India as the target location. The North-Eastern part of India is known for being one of the most language diverse regions in the world. Assam alone is home to a large number of languages and dialects. People of Assam speak English, Hindi, Bengali, Assamese and few other ethnic based languages in Assam like Karbi, Boro etc. As such, the conversations that participants in this region indulge in over various online social media platforms are also characterized by code-mixed and multilingual messages. Also, a large volume of the user conversations are phonetically typed using Roman script. Taking all these facts into consideration and also considering that there have not been many significant studies considering languages in Assam, this study finalizes upon Assam as the target location for generating the experimental datasets. Besides being characterized by various forms

of noise like irregular, informal and colloquial word forms, code-mixing and phonetic typing, the affect of native languages on the phonological and consequently the written forms of non-native languages in such a multilingual environment also adds to the challenges of automatic language identification.

The remaining part of the chapter discusses in detail the data collection and annotation process followed for generating the new datasets. The characteristics of the new datasets as well as the existing Twitter dataset (Zubiaga et al. [2016]) that have been used in this thesis work are discussed in detail.

3.2 Data Collection

We identify various channels (related social and political discussions) from Facebook and YouTube where community can participate in conversations with multi-lingual content. Since channels in Facebook and YouTube do not mandatorily maintain location information and the location information even though maintained is not standardized, instead of selecting channels by their geolocation, we select channels based on their local popularity. Firstly, Facebook channels (Facebook pages and Facebook groups) are identified manually that are popular (involve large user participation) in the state of Assam. From the set obtained, channels that are more frequently updated with user posts and comments are shortlisted. Data from these channels are collected using the Facebook Graph API. For collecting data from YouTube, we identify YouTube channels that cover topics similar to those of the selected Facebook channels. Video information (including user comments) from these channels are collected using the YouTube data API. The information collected include the following : (i) Comment ID (unique identification ID assigned to each comment) (ii) Text content in the comment (iii) User information (name and id of the user who posted the comment) (iv) Comment hierarchy information (Complete hierarchy information that includes the comments corresponding to a post (Facebook)/video (YouTube) and the replies corresponding to a comment) (v) Time of posting.

Table 3.1: General corpus characteristics

	Facebook	**YouTube**
Total number of comments	659,315	19,367
Total number of users	95,332	13,386
Total number of posts/videos	37,363	1,097
Total number of conversations[3]	50,525	3,750
Average number of comments per user	6.9	1.4
Average number of comments per posts	17.6	16.88
Average number of comments per conversations	4.6	5.01
Average message length (in words)	19.67	11.53

3.3 Sentence-Level Language Identification Dataset

The general characteristics of the collected data is shown in Table 3.1. A total of 659,315 and 19,367 number of phonetically typed microblog conversational messages have been collected from Facebook and YouTube respectively. From this collection, sentence-level and word-level language identification datasets are created by manually annotating a subset of these messages. These are discussed below in details.

3.3.1 Data Pre-processing and Annotation

All messages go through basic pre-processing before annotation that involves removal of all special symbols and numbers. Since the study considers transliterated text and all languages share the same script, symbols and numbers do not add any information useful for language classification. Also, we use the terms sentence and message interchangeably here because (i) messages in social media are generally short comprising of a single sentence, and (ii) even when messages are comprised of multiple sentences, sentences are not often clearly demarcated with proper sentence boundaries. Therefore considering the entire message as a single sentence is a convenient option. The annotation guidelines followed to annotate the dataset are as follows :

- Messages are annotated with one of the six languages : Assamese, Bengali, Hindi, English, Karbi and Boro. Messages in languages other than the above

six languages are ignored.

- In a highly multilingual environment, phonetic typing and mixing of words from multiple languages in a sentence without changing the underlying language sense is a common phenomenon. For example, embedding of English word *success* in a phonetically typed Assamese sentence *Ami edin success hom [we will be successful one day]*. This does not affect the language sense. We consider such sentences as monolingual sentences and assign the label corresponding to the underlying language. For example, with this assumption, the above sentence *Ami edin success hom* will be labeled as Assamese.

- There are messages where two or more legitimate sentences in different languages are concatenated to form a conversational message. For example, *Plz don't mind Ami amna e koisi* is a message formed by the concatenation of two sentences - *Plz dont mind* which is an English sentence and *Ami amna e koisi [I just said like that]* which is a transliterated Bengali sentence. We regard such messages as multilingual messages. Such messages are currently not annotated manually.

- Messages containing only user mentions, named entities, urls, incomprehensible words or expressions like hehe, haha etc. are tagged as ambiguous and ignored from the study.

Two annotators familiar with all the languages are deployed for annotation. At the end, samples that receive the same annotation from both annotators are retained. Figure 3.1 shows a screenshot of the sentence-level annotated dataset finally obtained.

3.3.2 Dataset Characteristics

The details of the sentence-level language identification datasets are shown in Table 3.2. For ease of reference, the sentence-level language identification dataset generated from the set of Facebook comments is named SLI-MI and the sentence-level language identification dataset generated from the set of YouTube comments is named SLI-MII. As also pointed out earlier, sentences that contain embedded

post_id	comment_id	user_id	text	conversation_id	date_time	language
E_451835_79138	C_535009	U_12760	kunu leader er fan howa o vala nay r hater howa o na	R_89958	2016-09-20T10:44:48 +0000	bn
E_451835_79138	C_206547	U_74883	Ami eta believe kri	R_89958	2016-09-20T10:46:41 +0000	bn
E_451835_91330	C_034251	U_46768	Bro hv u got any appointment shedule	R_65571	2016-06-22T13:23:11 +0000	en
E_451835_96739	C_279663	U_93761	Lot of thnx dear	R_87153	2016-10-31T14:30:34 +0000	en
E_451835_62983	C_945189	U_74736	Me agre with U	R_17742	2017-03-11T15:08:34 +0000	en

Figure 3.1: Screenshot of the sentence-level annotated dataset SLI-MI

Table 3.2: Characteristics of the sentence-level language identification datasets - SLI-MI and SLI-MII

	SLI-MI		**SLI-MII**	
Language	**#Messages**	**Avg Length**	**#Messages**	**Avg Length**
Assamese(as)	5,198	15 words	2,429	10 words
Bengali(bn)	1,594	12 words	129	8 words
Hindi(hn)	663	12 words	613	15 words
English(en)	21,531	24 words	8,315	11 words
Boro(br)	467	11 words	-	-
Karbi(kb)	822	14 words	-	-
Total	30,275	-	11,486	-

words from other languages without changing the underlying language sense are considered as monolingual sentences. Thus, the datasets described in Table 3.2 consist of monolingual sentences and may either be composed of all words in the same language or may also contain embedded words from other languages.

3.4 Word-Level Language Identification Dataset

Word-level language identification corresponds to determining the language corresponding to each word in a sentence/document. In this section, we discuss the word-level language identification dataset that we have generated from the collection of code-mixed comments collected from Facebook social media platform. The word-

comment_id	text
C_535869	i/en think/en i/en know/en it/en well/en enough/en but/en still/en check/en it/en out/en once/en more/en
C_206697	onno/bn job/en try/en koro/bn
C_034451	wish/en her/en a/en bright/en future/en
C_279686	varification/en er/bn ata/bn list/en ni/bn
C_945969	hahaha/amb kya/hn baat/hn hai/hn

Figure 3.2: Screenshot of the word-level annotated dataset WLI-M

level language identification dataset is used to train and evaluate various word-level language identification techniques.

3.4.1 Data Pre-processing and Annotation

All messages go through basic pre-processing before annotation that involves removal of all special symbols and numbers. Words are obtained by splitting sentences by white spaces. The word-level language identification dataset comprises four languages - Assamese, Bengali, Hindi and English. The guidelines followed for annotation are as discussed below :

- Words are annotated according to the context. The same word in different context can belong to different languages. For example consider the word *ache* in *stomach ache* and *koyek din ache [Few days left]*. The word ***ache*** in the former context is annotated as English and in the later context as Bengali.

- Words are annotated with one of the four languages : Assamese, Bengali, Hindi and English.

- Named entities do not belong to any language. Universal expressions like *haha*, *hehe* etc. also do not belong to any language. These words are annotated with a different label - *amb* (meaning ambiguous)

Like in sentence-level language annotation, two annotators familiar with all the languages are deployed for annotation. At the end, samples that receive the same annotation from both annotators are retained. Figure 3.2 shows a screenshot of the final dataset obtained. The dataset consists of the unique ID and the text content with the words and the labels separated by a slash.

Table 3.3: Characteristics of the word language identification dataset - WLI-M

Language	Total Words
Assamese(as)	18,467
Bengali(bn)	5,479
Hindi(hn)	3,449
English(en)	54,348
Total	81,743

3.4.2 Dataset Characteristics

The characteristics of the word-level language identification dataset is shown in Table 3.3. For ease of reference, this dataset is named as WLI-M. A total of 88,101 tokens are manually annotated following the annotation guidelines in Section 3.4.1.

The 88,101 words account for 17,402 unique words of which 9,637 are words that occur only in sentences of one particular language i.e., these are words that are valid in only one particular language and occur only in native language sentences. These words have not been borrowed in sentences of other languages. From the language identification standpoint, these are words that are in general easy to classify and a simple dictionary based method may be sufficient.

The remaining 7765 words are words that occur in sentences of multiple languages. This could be due to either of the following reasons - (i) these could be words that are valid across multiple languages and therefore they occur across sentences of multiple languages, (ii) these could be words that are valid only in one particular language and their occurrences in sentences of other languages are due to code-mixing or embedding and (iii) these could be distinct words in different languages but attain the same surface form due to noise induced by mis-spellings, creative spellings or irregular transliteration. Automatically identifying languages of such words that occur across sentences of different languages is a challenging task. Therefore in our proposed word-level language identification techniques proposed in Chapter 6 and 7, the proposed techniques are evaluated based on their performance over words in this category.

Table 3.4: Characteristics of the automatically annotated sentence-level datasets - SLI-AI and SLI-AII

Language	SLI-AI		SLI-AII	
	# Sentences	Avg Length	# Sentences	Avg Length
Assamese	130,687	17.53 words	688	8.64 words
Bengali	35,238	11.88 words	32	13.46 words
Hindi	26,154	12.88 words	577	18.69 words
English	386,726	21.54 words	4924	11.82 words
Karbi	3,880	14.77 words	-	-
Boro	801	12.88 words	-	-
Total	583,486	-	6,221	-

Table 3.5: Characteristics of the automatically annotated word-level dataset - WLI-A

Language	Total Words
Assamese	1,846,716
Bengali	319,412
English	8,860,800
Hindi	354,254

3.5 Automatically Annotated Datasets

Although this study makes use of only the manually annotated datasets for training and evaluating various language identification techniques, we also decide to release a larger automatically annotated dataset for the benefit of the language identification community. The labels for the larger datasets are obtained by using the classifiers trained over the manually annotated datasets - SLI-MI, SLI-MII and WLI-M described in Sections 3.3 and 3.4 respectively. Both the sentence-level and word-level classifiers are Convolutional Neural Networks (CNN) classifiers described in detail Chapters 5 and 6 respectively. The sentence-level language classifier takes as input entire sentences. Words in the sentences are represented by the corresponding word embeddings obtained by training a word2vec (skipgram) model over the entire collection of code-mixed sentences. The dimensions of the word embeddings are fixed

at 50. The CNN classifier is comprised of one convolution layer with filters of sizes 2,3 and 4. The word-level classifier is also a CNN based classifier. The input to the classifier is the target word and the words in the context represented by the corresponding word embeddings. The final dataset obtained comprises of three classes of information. First, it comprises of the social conversational information which includes the user information, the post/video information, the conversation information and the time information. Second, it comprises the sentence-level language information, i.e., the language corresponding to the entire sentence/message. Third, it comprises the word-level language information i.e., the language corresponding to each word in the sentence/message. The characteristics of the automatically obtained sentence and word-level classifiers are described in Tables 3.4 and 3.5 respectively. The automatically annotated sentence-level datasets comprising Facebook and YouTube comments are respectively named SLI-AI and SLI-AII respectively. The automatically annotated word-level annotated dataset is named WLI-A.

It should be noted here that since the labels obtained for the larger dataset are based on a classifier trained over a fixed sized manually annotated dataset, the labels are subjected to classification error. Therefore, in future, we plan to use semi-supervised or active learning strategies making use of the different levels of information available to obtain labels with greater accuracy and confidence.

3.6 Publicly available Twitter Language Identification Dataset

In addition to the datasets that are locally generated, we also use the publicly available Twitter sentence-level language identification datasets for investigating the performance of the sentence-level language identification techniques discussed in Chapter 5. Although there are other datasets that are publicly available, only the dataset discussed by Zubiaga et al. [2016] fulfil the required criteria. This dataset is described in detail in Table 3.6.

Table 3.6: Characteristics of the Twitter sentence-level language identification datasets (Zubiaga et al. [2016])

Language	#Tweets	Avg Length
Basque (eu)	754	6 words
Catalan (ca)	2,959	13 words
Glacian (gl)	963	10 words
Spanish (es)	21,417	11 words
English (en)	1,970	9 words
Portuguese (pt)	4,320	10 words
Total	32,383	-

3.7 Summary

In this chapter, we discuss different language identification datasets that are used in this thesis work which include an existing Twitter dataset and three locally generated datasets. The locally generated datasets comprising code-mixed user comments collected from a highly multilingual environment is divided into two parts - a manually annotated dataset and a larger dataset automatically annotated dataset. Along with the textual content, the dataset also comprises additional information in the form of user information, conversation information and date-time information.

The limitations of generating large scale annotated resources are well-known. Nevertheless, different computational as well as linguistic analyses require large amounts of annotated resources. In this chapter, we use the manually annotated datasets to obtain labels for the larger dataset. However, the labels obtained are subject to classification error considering that they are based on a fixed sized manually annotated datasets. In future, efforts will be directed towards semi-supervised learning and active learning techniques to obtain more accurate labels for the larger dataset.

4

Language Characteristics of Users in Multilingual Social Media Conversations

This chapter focuses on exploring the multilingual characteristics of user conversations, focusing particularly on Facebook. Understanding the language dynamics of online conversations is important for multiple applications. For example, past studies have shown that using language switching patterns in online conversations as features led to an improvement in the performance of natural language processing applications like humour detection and hate-speech detection. Although language dynamics in social networks have already been explored in literature from multiple perspectives, in the context of Indian languages, most studies have been around few languages like Hindi, Bengali, Tamil and Kannada. There have so far been no study considering languages in Assam. This study, therefore, focuses on characterizing user conversations in a multilingual environment considering Assam as the target location. Studying the language usage patterns of users in a multilingual environment and the influence of fellow users on the language choices of users in a multilingual environment is the primary focus area of this chapter. Although language usages in a multilingual environment can be affected by a large number of factors like fellow participants, topic, gender, age-group etc., the scope of this

thesis is limited to exploring the influence of fellow users. The observations from this study can pave the way for advanced analyses like information diffusion across language communities, understanding cross-lingual dynamics etc.

4.1 Introduction

The importance of language in social interactions is self-evident. Particularly in multilingual societies, the languages spoken by a user are instrumental in determining his/her social connections. Likewise, in online social networks as well, languages play an important role in determining the reach, influence and ties of a user with other users in the network. It is a well observed phenomenon that multilingual users use different languages while posting content in social network. Kim et al. [2014] have also shown that the languages used by a user in communicating online plays an important role in determining the influence of the user and attracting a target audience towards a discussion or a topic. Therefore, the choice of language by a user in a multilingual environment is not random. The language chosen by multilingual users in any particular context is actually a result of a conscious and judicious decision and is a function of various social, conversational and personal factors. For example a celebrity may probably choose to post a message in English to reach a wider international audience but may post a message in a regional language as well to express closeness or solidarity with a regional population (Nguyen et al. [2015]). Similarly, in a multilingual conversational environment, a user may more often tend to align his/her language choice with the languages used by fellow participants in a conversation (Nguyen et al. [2015]).

Studying language dynamics in multilingual social networks has multiple use-cases. Systematic patterns in language usage can be used to supplement noisy social media text in different NLP applications. Such supplementary information has been seen to yield improved performance (Carter et al. [2013], Agarwal et al. [2020]). Similarly information about how multilingual users are connected in a network and how different types of connections influence the language choices of users can be useful in designing solutions for different computational problems (e.g. language identification) as well as in different sociological (e.g predicting languages at the risk of

falling out of use) or linguistic (e.g. predicting induction of new vocabulary) analyses. Lantz-Andersson [2018] also suggest that social media language interaction of students can also be instrumental in improving students' ability to successfully use second language outside school. Understanding how different language users are connected and how they influence/get influenced by other users' language choices can also be useful in designing solutions for preservation of minority languages or identifying target user groups to diffuse information among different language communities.

When it comes to Indian languages, very few studies have explored the social and the conversational features in an Indian multilingual social media environment. Considering that India is known for it's language diversity with 1369 rationalized mother tongues according to the 2011 census [1], it is imperative that studying language usage characteristics in an Indian social media environment can be very insightful. Several studies have made interesting observations of the language interplay by multilingual users. For example Rudra et al. [2016] explore a bilingual Hindi-English environment and found that the languages used by a user is associated with the underlying emotion expressed in the message. They observe that in the case of Hindi-English bilinguals, the tendency to swear in Hindi is higher than that in English and concluded that swearing could be one of the strong reasons for code-switching. Few other studies exploring user-language association in an Indian multilingual social media environment include that Agarwal et al. [2017], Bawa et al. [2018] and Agarwal et al. [2020]. However, most of these studies have been centered around a few languages like Hindi, Bengali, Tamil, Kannada etc. Therefore, in this chapter, we attempt to understand the language characteristics in an online multilingual environment considering the languages in Assam. For the purpose of this dataset, we use the automatically annotated dataset SLI-AI discussed in Chapter 3. This study limits itself to exploring only the user information. The primary objective of this study is to explore how multilingual are social media users in such a language diverse environment. What are the language choices of a user in a multilingual environment and what are some of the factors that can influence these choices form a major component of this study. This study can pave the way

[1] https://en.wikipedia.org/wiki/Languages_of_IndiaCensus_of_India_figures

for advanced analyses like information diffusion among different communities, cross lingual dynamics etc.

4.2 Related Studies

Various studies have been conducted over online social platforms to understand various factors of importance in determining the linguistic behaviour of users online. Ndubuisi-Obi et al. [2019] show that the linguistic choice of a user while posting content online is influenced by a large number of factors like the emotional valence, the scope of the target audience etc. Similar observations are reported by Nguyen et al. [2015] that Twitter users adapt their languages to accommodate the interests of the target audience. Further, analyzing the characteristics of multilingual Twitter users, Hale [2014] suggests that multilingual users are more authoritative and active than monolingual users. Kim et al. [2014] also show that users who are bilingual in nature are less prone to form closely connected communities in comparison to monolingual users. Gavilanes et al. [2015] also show that multilingual users tend to show more diverse interactions than monolingual user. Hale [2014] also observe that multilingual social media users play an important role in bridging the gap between different monolingual communities. While Coats [2017] show that English acts as the bridge language between different monolingual communities, according to Hale [2014], multilingual users form a bigger and stronger bridging force than English language. Similar observation have also been reported by Eleta and Golbeck [2014]. A similar study have also been conducted by Eleta and Golbeck [2012] where they draw a similar inference that multilingual users in Twitter play an important role in diminishing the gap between multiple language communities.

Understanding language characteristics in a multilingual environment has also been found to be useful for different applications. For example Jin [2017] shows that understanding how information in social media propagates across languages and regions can be very useful for various applications like marketing and advertisement. Stewart et al. [2018] consider code-switching between Catalan and Spanish language and explore the association between language used and the political stance of a user. Catalan is the local language of the Catalonia which is a semi-autonomous region

in Spain. The authors observe that tweets in support of Catalan independence are more likely to consider words from Catalan language. A similar study is the study reported by Shoemark et al. [2017] with respect to tweets regarding Scottish independence. They also observe that tweets in support of Scottish independence are more likely to consist of Scottish terms than tweets in opposition of independence. Information as these can be of use for multiple applications.

Considering the Indian multilingual environment, Rudra et al. [2016] also consider a bilingual environment to understand if there is a preferred language for expression of emotion. They observe that in a bilingual environment comprising Hindi and English, people prefer to use Hindi for expressing negative sentiment. Similar observation is also shown by Agarwal et al. [2017] that in a code-switched environment comprising of Hindi-English bilinguals, the tendency of swearing in Hindi is much higher than that in English. Infact, they infer that swearing could be one of the factors leading to a code-switch within a dialogue. Bawa et al. [2018] also consider a bilingual Hindi-English dataset and observe that multilingual users are in general accommodating of the language choices of the fellow participants in a conversation. However, they also observe that factors like topic and sentiment also influence the language choices of a user. Multilingual characteristics of an Indian social media environment have also been explored by Agarwal et al. [2020] considering the ShareChat social media platform. They explore how information diffuse across multiple language communities. They observe that certain topics like politics and cricket disseminate faster across different language groups while many others are limited to a closed language group. Apart from exploring the multilingual characteristics from the social conversational viewpoint, most other studies considering multilingual social media content in Indian languages focus on addressing different computational problems like language identification Vyas et al. [2014], offensive language detection Kapoor et al. [2019], sentiment analysis Phani et al. [2016] and hate-speech detection Bohra et al. [2018].

As discussed in Chapter 3, the dataset considered in this study have been collected from a multilingual social media environment considering Assam as the target location. The goal is to characterize the user conversations in such a highly multilingual environment. This is to the best of our knowledge, the first such study

considering languages in Assam. This study explores the characteristics of multilingual users in a highly multilingual environment and how the choices of languages are influenced by other users in such a multilingual environment. Effect of factors like topic, sentiment or target audience on the language choices of multilingual users are not explored in this study. We plan to explore these in future.

4.3 User characteristics

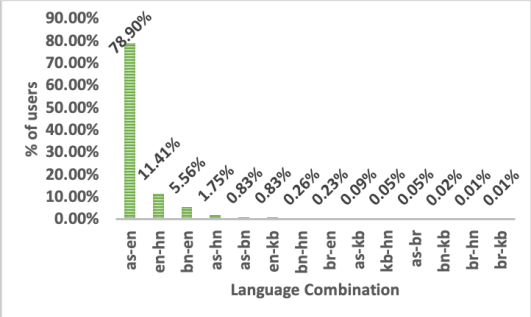
In this section, we analyze the multilingual characteristics of user conversations considering the automatically annotated dataset SLI-AI described in Chapter 3. The details of the dataset are given in Table 3.4. Table shows that English is a dominant language comprising of 66.27% of the total communications while the remaining languages account for 33.72%. It is interesting to observe that English is used as a dominant language even in political discussions related to Assam. Usage of English may have been influenced by the factors such as personal preference, target audience etc.

Although it is implicit that the unrestricted nature of the social media platforms is leveraged by the users to post content with ease in languages of their choice, it is also perceptible that there are a wide range of factors that can influence the linguistic behaviour of a user in an online multilingual environment. In the following discussion, we dive deep into the language characteristics of user conversations in a multilingual environment.

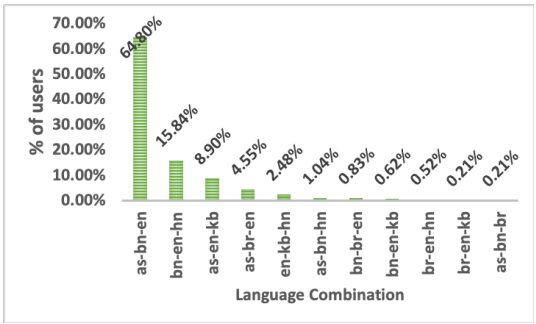
4.3.1 How many languages does a user use while participating in online conversations?

Table 4.1: User break-up in terms of the number of languages used in conversations

Number of languages	Percentage of users
1	49.08
2	36.48
3	10.70
4 or more	3.72



(a) Language combinations used by users using two languages for communicating online



(b) Language combinations used by users using three languages for communicating online

Figure 4.1: Language combinations used by users who communicate online using (a) two or (b) three languages

Table 4.1 shows the break up of the users in terms of the number of languages they use while participating in conversations over social media platforms. This analysis considers only those users who have posted atleast two comments. From Table 4.1, it is observed that 49.08% of the users use only one language at the sentence-level for communicating online while the remaining 50.91% use more than one language. Amongst the users who use more than one language, it is seen that majority of the users use two languages with a small percentage of users using three or more languages. Therefore, it is interesting to note that even in a highly multilingual environment, majority of the users use one or two languages for communicating online with a very small number of users using three or more languages.

Further, we explore the languages used by those users who use two or three languages. This is shown in Figure 4.1. It is seen that users that use two languages in online communications are more likely to use a local language along with English language than using two local languages. For example, we can see in Figure 4.1(a) that the percentage of users using both Assamese and Bengali languages is less than the percentage of users using Assamese and English or Bengali and English. The same is also true for other local languages like Karbi and Boro. Therefore, even in a highly multilingual environment, majority of the multilingual users are more likely to use one of the local languages with a global language (English in the case of Assam) than using multiple native languages.

Further, we explore how users who use only one language for online communi-

cation interact with other users in a highly multilingual environment. We observe that out of the total users that use only one language while communicating online, 77.8% participate only in posts of the same language. While this is an expected behaviour for users communicating with only one language, it is interesting to investigate the behaviour of the remaining 22.1% and how they participate in posts of other languages. It is observed that out of the remaining 22.1%, 64.4% of the times, there have been a comment by a user who use multiple languages in their conversations. Kim et al. [2014] have pointed out that bilinguals act as bridge for communicating with users belonging to other language groups in a multilingual environment. Although it seems to be a valid assumption in this case as well, we refrain from making a conclusion here and keep this analysis open for future work. Further, Bawa et al. [2018] also observe that the topic of discussion or the associated sentiment also determine the language choice of a user. Therefore, there is a possibility that a user might have knowledge of multiple languages but prefers to communicate in only one particular language due to the associated sentiment or emotion. We also plan to explore this in future.

In contrast to users that communicate using only one language, users that communicate using multiple languages participate in posts of multiple language and also switch languages within the same conversation. In the following discussion, we explore what is the language switching pattern of multilingual users within a conversation and how are the language choices of multilingual users influenced by other participants.

4.3.2 How users switch between languages within the same conversations?

We have seen from Table 4.1 that nearly 50.1% of the users use more than one language while communicating online. We investigate the characteristics of this set of users further in this section. Specifically, we are interested in investigating whether a user participates in any conversation in a single language or switches language even within the same conversation. An example of a conversation is shown in Figure 4.2. The highlighted entries in the figure show multiple comments by the same user within the same conversation. We see that there is a switch in language by the

post_id	comment_id	user_id	conversation_id	date_time	text	language
E_51995_89870	C_608511	U_01034	R_85118	2015-08-12T05:50:56+0000	nw tet techer apntmnt debo	bn
E_51995_89870	C_085048	U_68986	R_85118	2015-08-12T05:55:20+0000	I think that is permanent	en
E_51995_89870	C_084889	U_01034	R_85118	2015-08-12T07:37:45+0000	mntn to kore nai	bn
E_51995_89870	C_418115	U_68986	R_85118	2015-08-12T08:00:35+0000	Bikal aa paibai kbr	bn
E_51995_89870	C_751421	U_01034	R_85118	2015-08-12T11:59:39+0000	okk wait e achi	bn
E_51995_89870	C_750700	U_01034	R_85118	2015-08-12T17:17:30+0000	tumi ki parmanent	bn

Figure 4.2: Example illustrating change of user language within the same conversation

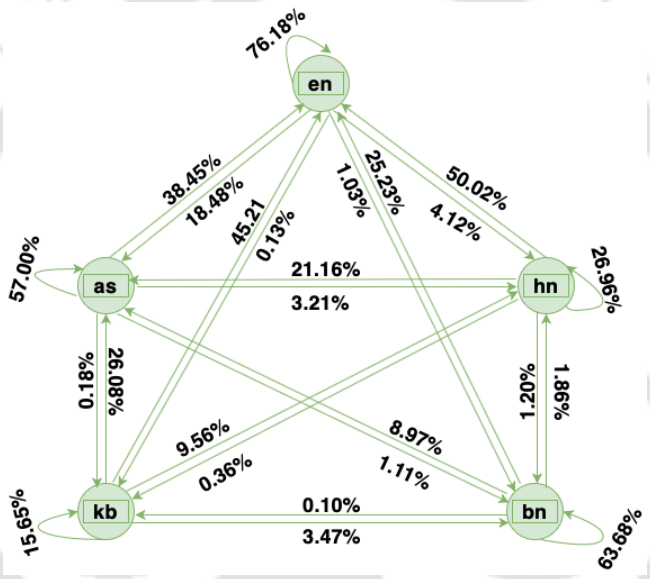


Figure 4.3: Language transition probabilities of multilingual users within the same conversation

user from English (en) to Bengali (bn) within the same conversation. To empirically investigate the chances of a user switching languages within the same conversation, a transition diagram as shown in Figure 4.3 is obtained. A directed edge between two nodes represents the percentage a user switches language from the source language to the target language within a conversation. We have ignored the Boro language for this analysis because of very less number of comments in Boro language in our dataset. From the figure, it is seen that although for majority of the cases (3 out of 5 languages), the chances of staying in the same language is higher than switching to another language, for Karbi and Hindi languages, the reverse is true. It should also be noted that for Hindi and Karbi languages, the number of comments is far

less than the remaining languages. Therefore, each sample can be over-represented for these languages. Nevertheless, the likelihood of changing languages within a conversation is also quite significant (between 18.4% to 50.2%). Further, whenever there is a language switch, the percentage of switching from English to one of the local languages or vice-versa is always greater than switching between multiple local languages. In the subsequent discussions, we want to understand the factors that trigger the switching of language within a conversation.

4.3.3 Is the language choice of a user influenced by other users participating in the conversation?

It is observed in Figure 4.3 that a user switches languages within the same conversation. An intuition is that the switch in language within a conversation could be triggered by the language used by other users within the same conversation. We validate our assumption by empirically investigating the same in our dataset. It is observed that out of the total number of times a user changes language within a conversation, 64% of the times, there is a preceding comment by another user in the new language. In other words, if a user u_i changes his language within a conversation from l_1 at time t_1 to l_2 in time t_2, then 64% of the times there is a comment by another user u_j in language l_2 between time t_1 and t_2. This validates our assumption that the language choice of a multilingual user in a conversation might have been influenced by the choice of other users participating in the conversation. However, it has also been seen that for the remaining 36% of the times, the user switches to a language other than the one used in the conversation so far. Ndubuisi-Obi et al. [2019] have pointed out that apart from fellow users, the language choice of a user is also influenced by other factors such as emotional valence, target audience etc. Influence of these factors have not been explored in this study and will be explored in future.

4.3.4 How are words from different languages mixed within the same sentence?

In the previous sections, we have discussed the multilingual characteristics of users by analysing the languages at the sentence-level. However, a very common phe-

Table 4.2: Language mixing characteristics of sentences in multilingual environment

Language	Total number of sentences	Number of sentences with embeddings	Percentage
English (en)	386,726	49,950	12.91%
Assamese (as)	130,687	90,542	69.28%
Bengali (bn)	35,238	25,787	73.17%
Hindi (hn)	26,154	15,272	58.39%
Karbi (kb)	3,880	1,227	31.61%
Boro (br)	801	236	29.42%

nomenon in social media communications is the phenomenon of code-mixing where some word(s) may belong to a different language than the overall language of the sentence. In this section, we analyze how words from different languages are mixed in the same sentence. Table 4.2 shows the mixing characteristics across sentences of different languages. Each row in the table shows the percentage of sentences in a particular language that consists of words embedded/borrowed from other language. It is observed that English sentences in general have lesser chances of embedding from other languages than other local language sentences. Infact, from Table 4.2, we see that the percentage of embedding/borrowing of other language words in sentences of native languages is quite significant (29.42% to 73.17%). This observation also motivates the importance of efficient multilingual models for processing text from such multilingual environment.

We have seen in Table 4.2 that there are high chances of mixing different languages within the same sentence in a multilingual environment. We further extend this analysis to study how users mix languages within their messages. Table 4.3 shows the users categorized according to the number of languages they use within a message. We find that only 34.51% of the users post messages with all words in a single language while the majority (65.47%) mix words from multiple languages in their messages. However, it is seen that amongst the users using multiple languages, majority of them use two or three languages with a very small percentage (9.55%) using more than three language. This observation is also in coherence with previous observations with regards to user language choices where it is observed that even in a highly multilingual environment, majority of the users' language usage is restricted to two or three languages and often involves mixing of one of the local languages

Table 4.3: User classification based on language mixing characteristics

Number of languages	**Percentage of users**
1	34.51%
2	36.01%
3	19.91%
4 or more	9.55%

with a global language.

4.4 Summary

This study explores the language dynamics in a multilingual social media environment, focusing primarily on Facebook. We observe that the even in a highly multilingual environment, most users resort to a very small number of languages. Amongst the users that use more than one language in online conversations, majority (71.66%) of them use two languages with a very small percentage (28.73%) using three or more languages. It is also observed that the language choice of a multilingual user is often oriented with the languages used by other users participating in the conversation. While from our observations, it can be safely assumed that a change in language within a conversation initiated by a participant triggers other users to orient themselves to the change. However, what triggers the first change of language within a conversation has not been explored in this study. Such changes could be influenced by multiple factors like emotional factors, the target audience or personal preference. We plan to explore this in future. It has also been observed that embedding words from other languages within a sentence of a different language is quite a common phenomenon. Further, it has been observed that embedding of words is more common in native language sentences than English sentences. What factors trigger the embedding of words is also a direction to explore. Systematic patterns, if at all exists in the embeddings, can be used to supplement the noisy textual information for different NLP applications.

5

Sentence-Level Language Identification in a Multilingual Environment Using Social Conversational Features

This chapter addresses sentence-level language identification in a collection of code-mixed social media messages collected from a highly multilingual environment. The proposed techniques make use of social conversational features to obtain improved language identification performance. Although social conversational features for ALI have been explored previously using methods like probabilistic language modeling, these models often fail to address issues related to code-mixing, phonetic typing, out-of-vocabulary etc. which are prevalent in a highly multilingual environment. This study differs in the way that the social conversational features are used to propose text refinement strategies that are suitable for ALI in highly multilingual environment.

5.1 Introduction

With the explosion of multilingual content on Web, particularly in social media platforms, identification of languages present in the text is becoming an important task for various applications. ALI on well-written text has been seen to achieve high

accuracy (up to 99.8% accuracy by Cavnar and Trenkle [1994]). However, automatic language identification (ALI) in social media text is considered to be a non-trivial task due to the presence of slang words, misspellings, creative spellings and special elements such as hashtags, user mentions etc. In a multilingual environment, factors such as code-mixing and phonetic typing adds to the noise making ALI an even more challenging problem. This chapter focuses on sentence-level language identification of code-mixed social media text in a highly multilingual environment. The proposed methods in particular attempt to address issues related to code-mixing, phonetic typing, out-of-vocabulary etc. which are prevalent in a highly multilingual environment.

Though ALI on well-written text has been seen to achieve high accuracy (up to 99.8% accuracy (Cavnar and Trenkle [1994])), traditional ALI methods based only on the textual content fail to exhibit comparable performances in social media content (Carter et al. [2013]). Some of the recent studies on ALI in microblogs have explored various approaches; (i) Content based features such as character ngrams, word ngrams, dictionary based features, minimum edit distance and context information (Zubiaga et al. [2016], Barman et al. [2014]), (ii) user's historical information such as language profile for language identification in user generated content in social media platforms (Carter et al. [2013], Wang et al. [2015]), and (iii) click-through information in query log for search engine applications (Ceylan and Kim [2009]). Though the above studies have shown promising performances, they exhibit shortfalls while handling cases like out-of-vocabulary, naturally embedded borrowed words, code similarity across languages and uncertainty of language usage in a highly multilingual environment. In spite of the above shortfalls, social and conversational features such as language prior tend to contribute in enhancing ALI performances (Carter et al. [2013], Wang et al. [2015]). This chapter proposes a method for ALI which is more resistant to code mix, code similarity, and out-of-vocabulary (new unseen irregular text) by refining the original text with more language bias text taking advantages of various social and conversational features into consideration. The idea is to generate a new representation of the original text without distorting underlying language sense.

Unlike the studies reported by Carter et al. [2013] and Wang et al. [2015], the

objective of this study is to refine a noisy conversational text to a language bias text which can potentially capture the underlying language of the original message by exploiting the social and conversational features. This study considers three types of social features related to highly multi-lingual conversations; (i) user's language profile, (ii) language distribution in ego-centric conversations, and (iii) language distribution in comment-reply threads. Conversations on a microblogging platform are often driven by a theme such as a post in Facebook, or a tweet in Twitter or a video in YouTube. Inspired by ego-centric network in social network analysis, we refer to conversations driven by such theme as an *ego-centric conversation* in this study[1]. Considering the phonetically typed Bengali message *hbse koitelgbo ofiicaly notification dite lgbo [hbs should say officially notification should be given]* with borrowed English words *ofiicaly [officially]* and *notification* embedded in it, our refinement technique transforms it to *koitelgbo amrra koitelgbo paitm hobe dite [should say we should say get ok give]* which is a collection of words in Bengali language. The new message, as we see in this case, may not be a valid sentence or may not have any semantic relation with the original message. But the newly generated text can capture the underlying language sense better than original text. In other words, the refinement technique can be regarded as a feature extraction technique that represents a noisy code-mixed text by a collection of terms all of which most likely belong to the same language (the underlying language of the original message). In summary, this chapter makes the following contributions :

- Propose three text refinement methods using the social features mentioned above; (i) term co-occurrence score, (ii) weighted term co-occurrence score and (iii) global language profile based term co-occurrence score. Among these, the global language profile based term co-occurrence score outperforms the others.

- Investigate the effects of the proposed text refinement methods over ALI using seven state-of-the-art text classification frameworks. From the results, it is evident that the social conversational features boost the performance of the

[1]Ego-centric network in social network analysis is a network formed from the point of view of a single node which is called the ego. An ego is the central point or focal point of the network. In this study, the term *ego-centric conversations* is used to refer to conversations driven a theme. When considering the Facebook platform, the term ego refers to a Facebook post. Similarly, in YouTube and Twitter platforms, the term ego refers to a video and a tweet respectively.

ALI systems. Further, it is observed that under the proposed framework, the comment-reply feature outperforms the other social features (user features and ego features).

Some of the advantages of the proposed approach are (i) once the refined text is obtained, the standard text classification frameworks can be deployed, and (ii) an effective refinement method can potentially address the out-of-vocabulary problem often experienced in text classification.

5.2 Related Studies

Social media content is noisy and less grammatically correct than well edited formal text imposing challenges to different text processing applications including language identification. Baldwin et al. [2013] collect data from various social media sources and perform an elaborate statistical and linguistic analysis of the social media content to confirm the above observation. Similar observation is also obtained by Carter et al. [2013]. They show that the performance of the language identifiers decreases from 99.4% in formal text to 92.4% in microblog text using the n-gram based approach proposed by Cavnar and Trenkle [1994]. These observations validate that ALI over social media content is a challenging problem. Existing literature can be divided into two categories based on the nature of the information used - first, approaches that make use of only the textual content and second, approaches that also take into account additional information like the social and conversational features associated with a given text.

Content based ALI systems commonly make use of features such as character ngrams, word ngrams, dictionary based features, minimum edit distance and context information (Barman et al. [2014]). Few of the recent studies in ALI also use neural networks (Kocmi and Bojar [2017]) and graphical models (Tromp and Pechenizkiy [2011], Vogel and Tresner-Kirsch [2012]). In another direction, Goldszmidt et al. [2013] attempt to adapt the ALI system trained with Wikipedia articles to social media content by augmenting it with Twitter data extracted using location information of the tweets.

While the noisy nature of social media content imposes various challenges to

text processing, the supplementary information in the form of various social and conversational features provide additional cues that have been explored for various text processing applications including language identification. Carter et al. [2013] consider five different microblog characteristics - blogger characteristic, link characteristic, mention characteristic, tag characteristic and conversation characteristic. These characteristics are used as priors that associate the given text with one or more languages. Different approaches to combine the priors are explored. They observe an improvement of 5.5% using the social conversational characteristics compared to the text based baseline. In similar lines, Wang et al. [2015] also explore the use of user language profiles for language identification on game chats.

In brief, almost all of the studies discussed above use information extracted directly from the content (for example, character or word n-gram) or social features (for example language prior) as the features to build an ALI classifier. Unlike these studies, we use social and conversational features to refine a noisy text and generate a new language bias text. The new transformed text is then subjected to various classification framework. Among the existing studies using social features, that of Carter et al. [2013] is conceptually more close to our study i.e., generate new representation of the messages using social features, and therefore it is considered as one of the baseline methods. Apart from traditional text classification methods, we also consider multi-channel convolutional neural network reportedly used for sentence-level sentiment classification task by Kim [2014]. This framework is capable of capturing different representations of a text.

5.3 Proposed Methodology

This section presents the proposed framework for generating textual features capable of capturing underlying language of a target message by exploiting social conversational characteristics namely (i) *user's language profile*, (ii) *language dynamics in ego-centric conversations* and (iii) *comment-reply threads* as shown in figure 5.1. The proposed model works in two phases.

1. Given a message, generate a new message which can potentially represent the underlying language. The new message may not be a valid meaningful sentence

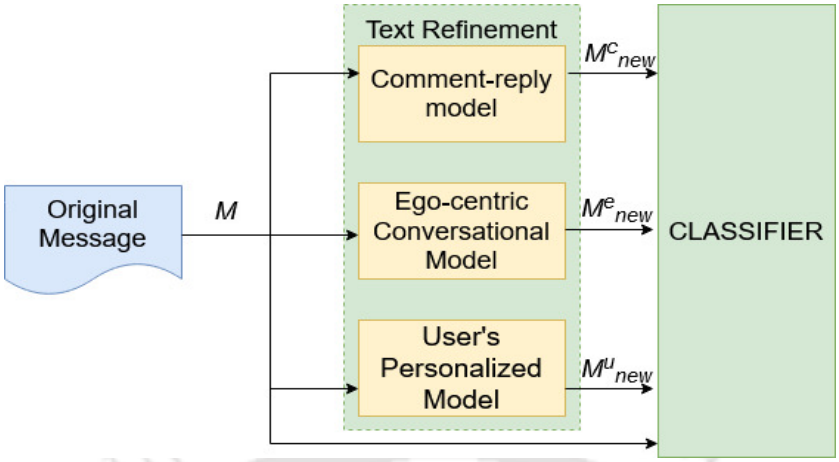


Figure 5.1: Schematic diagram of the proposed sentence-level language classification model

or may not have any semantic relation with the original message.

2. The refined messages and/or the original messages are then fused to build classifiers for language identification task.

5.3.1 Refining text using social conversational characteristics

Table 5.1 shows the list of symbols used in this chapter. Given a conversational text message, the task is to generate a refined text which can potentially represent the underlying language of the message. Let $\mathcal{D}$ denote the global collection of the messages in our training dataset, $\mathcal{L}$ denote the set of languages of the messages in $\mathcal{D}$, and $\mathcal{V}$ denote the set of unique terms in $\mathcal{D}$. At the core of the refinement model, it uses a term co-occurrence metric for mining new terms by exploiting the following three classes of social conversational characteristics.

1. **User's Personalized Messages:** If the task is to refine a message M posted by an user u and generate a new message M^u_{new}, then the collection of messages posted by the user u (denoted as $\mathcal{D}^u \subset \mathcal{D}$) and the vocabulary set present in the collection (denoted as $V^u_{\mathcal{D}} \subset \mathcal{V}$) are used for refining the message.

2. **Ego-Centric Conversational Messages:** If the message M belongs to an ego e, all the messages belonging to the ego e (denoted as $\mathcal{D}^e \subset \mathcal{D}$) and respective unique vocabulary set (denoted as $V^e_{\mathcal{D}} \subset \mathcal{V}$) are considered for generating M^e_{new}.

Table 5.1: List of symbols and notations

Notations	Description
$\mathcal{D}$	Message collection ($\mathcal{D}^s$ is the message collection satisfying the condition s)
$\mathcal{D}_l$	Set of messages belonging to the language $l \in \mathcal{L}$
$\mathcal{L}$	Set of languages present in $\mathcal{D}$
$\mathcal{L}_t$	Set of languages containing the term t
$\mathcal{V}$	Vocabulary set present in $\mathcal{D}$. $\mathcal{V}_{\mathcal{D}}^s$ is the vocabulary set present in $\mathcal{D}^s$
s	Type of social conversational characteristics, which can be either u, e or c.
u	User oriented social conversation
e	Ego/Post oriented social conversation
c	Comment-reply oriented social conversation
M	A conversational message
M_{new}^s	Transform message obtained using $s \in \{u, e, c\}$ type of social conversation
$\mathcal{S}(t_1, t_2)$	Similarity function between two words t_1 and t_2
$\mathcal{R}(t)$	Refinement function of text t
k	Number of terms to be returned by $\mathcal{R}(t)$
$\mathcal{F}$	Transformation function to obtain M_{new}^s from M

3. **Comment-Reply Conversation Thread:** If the message M belongs to a comment-reply thread c, all the messages belonging to the thread c (denoted as $\mathcal{D}^c \subset \mathcal{D}$) and respective unique vocabulary set (denoted as $V_{\mathcal{D}}^c \subset \mathcal{V}$) are considered for generating M_{new}^c.

In Chapter 4, we observe that even in a highly multilingual environment, majority of the users limit their language usages only upto three languages. We will observe in Section 5.4 that the same applies for egos and comment-reply threads i.e., even in a highly multi-lingual environment (up to six languages in our study), majority of the egos or comment-reply threads limit their language usages only up to three languages. While refining a message M, the intuition is that the terms extracted from $\mathcal{D}^u$, $\mathcal{D}^e$ and $\mathcal{D}^c$ may potentially represent user's language usage pattern better, rather than considering the entire corpus.

Now, we provide a generalized function to describe the proposed model. Let s denote the type of social characteristics (u or e or c defined above) that the message

M belongs to. Given a term t in message M, a refinement metric $\mathcal{R}$ extracts one or more terms in $V_{\mathcal{D}}^s$ (different from t), which potentially capture the underlying language of the message M, by exploiting the social and conversational relations inherently present in $\mathcal{D}^s$. The function $\mathcal{R}$ further uses another similarity function $\mathcal{S}$ to mine potential candidates by considering the collection $\mathcal{D}^s$. As mentioned above, this study considers *term co-occurrence* as the core similarity measure, and proposes the following three similarity estimates.

Term Co-occurrence in $\mathcal{D}^s$:

As described above, $\mathcal{D}^s$ represents a personalized set of messages contributed only by a specific user or ego or comment-reply thread. Therefore, term obtained from $\mathcal{V}_{\mathcal{D}}^s$ is likely to capture the characteristics of the associated user, ego or comment-reply thread. Now, we define the similarity score $\mathcal{S}(t_i, t_j)$ between two terms t_i and t_j as the number of times both the terms t_i and t_j occur in $\mathcal{D}^s$ with Δ distance. For a term $t \in M$, if $t_1, t_2, t_3, ..., t_{|V_{\mathcal{D}}^s|}$ denotes the ordered term sequence arranged in the decreasing order of $\mathcal{S}(t, t_i)$ scores, $\mathcal{R}(t)$ returns top k terms i.e., $\mathcal{R}(t) = \{t_1, t_2, t_3, ..., t_k\}$ where $k > 0$.

Weighted Term Co-occurrence Score

A problem with the above similarity estimate S is that it may return shared codes[2] in the refined text. A simple strategy to address this issue is to impose a condition that the new term should not belong to the set of shared codes. However, many of the messages are formed only by the shared codes. We therefore propose the following weighted variant of the above estimate.

$$\mathcal{S}_w(t, t_i) = \frac{\mathcal{S}(t, t_i)}{|L_{t_i}|} \tag{5.1}$$

where L_{t_i} is the set of languages that contain the term t_i. The similarity score for a term is inversely proportional to the number of languages that contain the term.

[2]Shared codes : Words that are common to more than one language

Global Language Profile Based Term Co-occurrence Score

The above two estimates may suffer from data sparsity. Size of the $|\mathcal{D}^s|$ for many of the users or egos or comment-reply threads may be small.Because of such sparse nature of the dataset, there may not be much variation between the terms in the refined message and the original message. The estimate in this section attempts to address the data sparsity problem without losing the characteristics of the underlying target language.

The similarity scores defined above are independent of the language profile of the associated user or ego or comment-reply thread. We will observe in section 5.4 that in majority of the cases, language usage in any social and conversational feature limit to one or two or three (in the decreasing order from one language to two and then three). Based on this observation, we propose the following language profile based similarity estimate $\mathcal{S}_{pro}(t_i, t_j)$.

Let $\mathcal{L}^s \subset \mathcal{L}$ be the set of languages of the messages in $\mathcal{D}^s$ and $\mathcal{D}_l \subset \mathcal{D}$ be the set of messages belonging to the language $l \in \mathcal{L}$. Then, we define profile based message collection $\mathcal{D}_{pro}$ as follows.

$$\mathcal{D}_{pro} = \bigcup_{\forall l \in \mathcal{L}^s} \mathcal{D}_l$$

Now, the language profile based similarity estimate $\mathcal{S}_{pro}(t_i, t_j)$ is defined as the number of times t_i and t_j occur together in $\mathcal{D}_l$ with Δ distance i.e., distance between t_i and t_j is at most Δ.

After defining the various co-occurrence based similarity estimates, we now define the message refinement function. For a given message M without specifying the language information, the task of generating a new message characterized by the social and conversational characteristics s i.e., M^s_{new} can be realized by a transformation function $\mathcal{F}$ as defined below.

$$M^s_{new} = \mathcal{F}(M, s, \mathcal{D}, \mathcal{R}, \mathcal{S}, \mathcal{C}) \tag{5.2}$$

where $\mathcal{C}$ denotes list of additional parameters applicable for the model. In the above discussion while defining $\mathcal{R}$ and S, the parameters k and Δ are denoted by $\mathcal{C}$ i.e.,$\mathcal{C} =< k, \Delta >$.

Now, the M^s_{new} in equation 5.2 can be further expressed as the concatenation of $\mathcal{R}(t)$ for all the terms t in M as defined below.

$$\begin{aligned} M^s_{new} &= \mathcal{F}(M, s, \mathcal{D}, \mathcal{R}, \mathcal{S}, \mathcal{C} =< k, \Delta >) \\ &= \bigoplus_{\forall t \in M} \mathcal{R}(t) \end{aligned} \tag{5.3}$$

where $\bigoplus$ denotes concatenation operator. Thus, $|M^s_{new}| = k.|M|$.

Figure 5.2 shows an example of how a refined text is obtained from a given input text. For the sake of illustration, let us consider the user feature. Then, from the manually annotated dataset, user personalized messages are obtained which are the messages posted by that particular user (in case of *term co-occurrence* similarity estimate or *weighted term co-occurrence* similarity estimate) or the collection of messages based on the language profile of the user (in case of *global language profile based term co-occurrence* similarity estimate). From the user personalized messages, a user personalized term co-occurrence matrix is obtained. The Figure 5.2 shows the example of a raw *term co-occurrence* matrix. In case of *weighted term co-occurrence* or *global language profile based term co-occurrence*, the entries in the matrix, i.e., the co-occurrence scores are divided by the total number of languages in which the terms appear. Given an input sample, each term in the original message are then replaced by the most frequently co-occurring term.

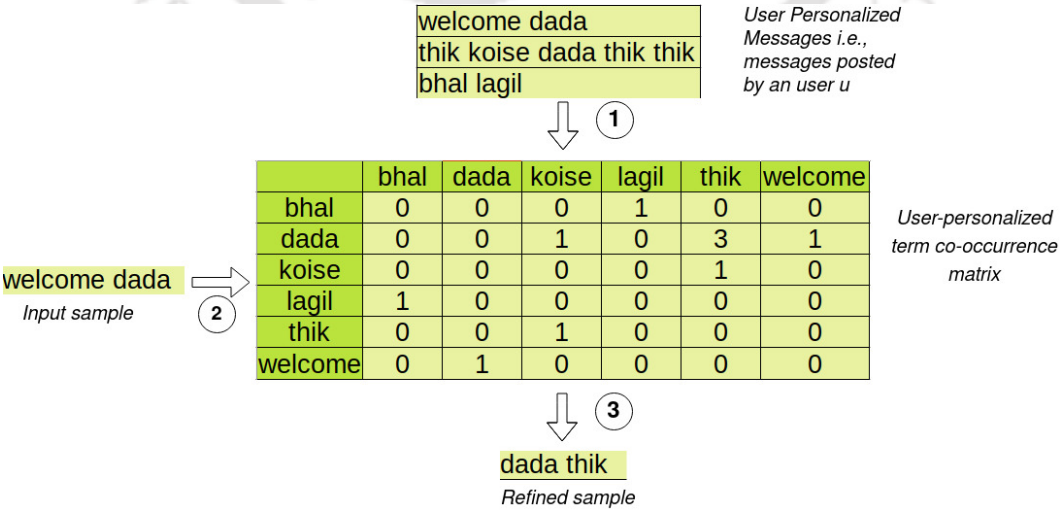


Figure 5.2: Example illustrating text refinement using user personalized messages

5.3.2 Sentence-Level Language Classification

After obtaining refined messages for all the experimental text messages, to investigate the effect of social conversational characteristics, the original and transformed texts are further subjected to various classification frameworks for building sentence-level ALI systems. Seven (7) different popularly used classification frameworks are chosen for this study, namely Naive Bayes (NB), Support Vector Machines (SVM), Logistic Regression (LR), Decision Tree (DT), Random Forest (RF), Convolutional Neural Network (CNN) and Bidirectional Long Short Term Memory Network (BiLSTM). To incorporate the text generated from social conversational characteristics, we propose to exploit two types of feature fusion strategies; (i) *early fusion*, and (ii) *late fusion*. In our experimental setup, CNN is subjected to both early and late feature fusions and all the other classifiers are subjected to only early feature fusion.

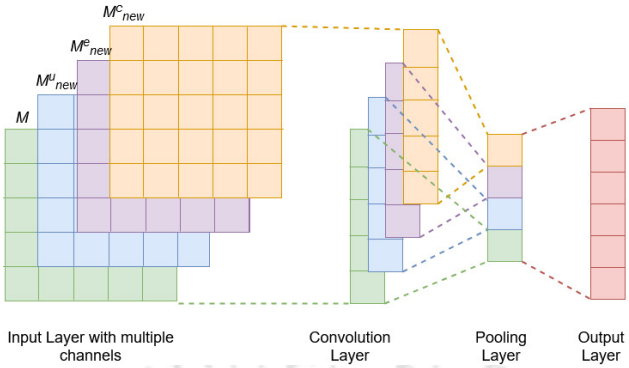
Except for CNN and BiLSTM, the early fusion strategy concatenates features extracted from original message and transformed messages. Feature fusion strategies in CNN and Bi-LSTM are discussed below.

5.3.3 Multichannel Convolutional Neural Network

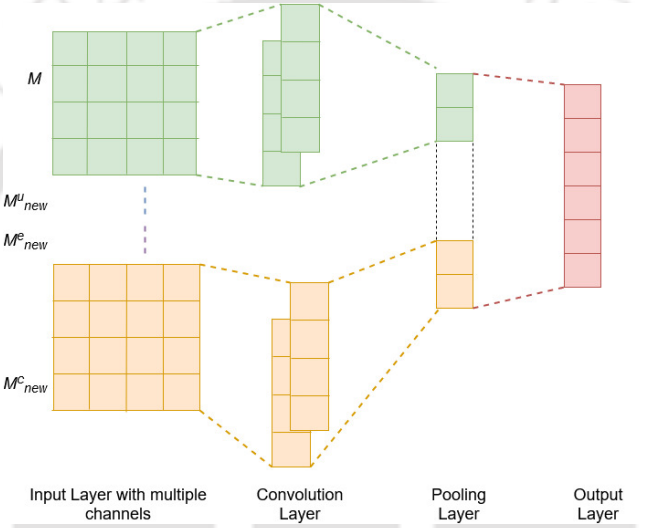
A multichannel CNN consists of multiple channels representing multiple views of the input data. This study considers variants of the two Multichannel Convolutional Neural Networks (MCNN) for sentence-level classifications reported by Kim [2014] and Shi et al. [2016]. We refer the first model as *early fusion* MCNN and the second model as *late fusion* MCNN. In these studies, different channels are created using different word embedding techniques. For a given message, idea of using different embeddings is to capture different message representation. Unlike these studies, the objective of the proposed framework is to capture *different representations of the language of a target message* through the features characterized by different channels, rather than message representation.

Early Fusion MCNN (EFCNN)

This model is inspired by the MCNN model proposed for sentence classification by Kim [2014]. Instead of different word embedding channels, we apply different social



(a) EFCNN



(b) LFCNN

Figure 5.3: Proposed multi-channel CNN for sentence classification

characteristics channels. Unlike Kim [2014], our model has one channel for the original message and one channel for each social feature as shown in Figure 5.3(a). For a given message M, the input to its channel is a two dimensional matrix where each row corresponds to a term in the message and is represented by the embedding vector of the term. Similarly, the channels of refined messages M^u_{new}, M^e_{new}, and M^c_{new} are also created. Each row in $M^{u/e/c}_{new}$ corresponds to the term in the refined message obtained using the transformation function $\mathcal{F}$. Further, different channels in our model may have different number of rows depending on the value of the parameter k. If k= 1, then the number of rows in $M^{u/e/c}_{new}$ is equal to the number of rows in M. If k=2, then the number of rows in $M^{u/e/c}_{new}$ will be 2 times the number of rows in M. In short, the number of rows in $M^{u/e/c}_{new}$ will be k times the number of

rows in M. Since different values of k can lead to matrices (channels) with different number of rows, therefore at the time of fusion, the matrices are padded with extra rows such that the number of rows in all channels are equal. Except for channel characteristics, we maintain almost similar network structure as defined by Kim [2014] to maintain comparability.

Late Fusion MCNN (LFCNN)

This model is also inspired by the MCNN model commonly used in image processing (Barros et al. [2014]). The same model has also been used in sentence classification by Shi et al. [2016]. In the late fusion model (shown in Figure 5.3(b)), features extracted from different channels are not combined until the penultimate layer.

5.3.4 Early Fusion Bi-directional Long Short Term Memory Network (EF-BiLSTM)

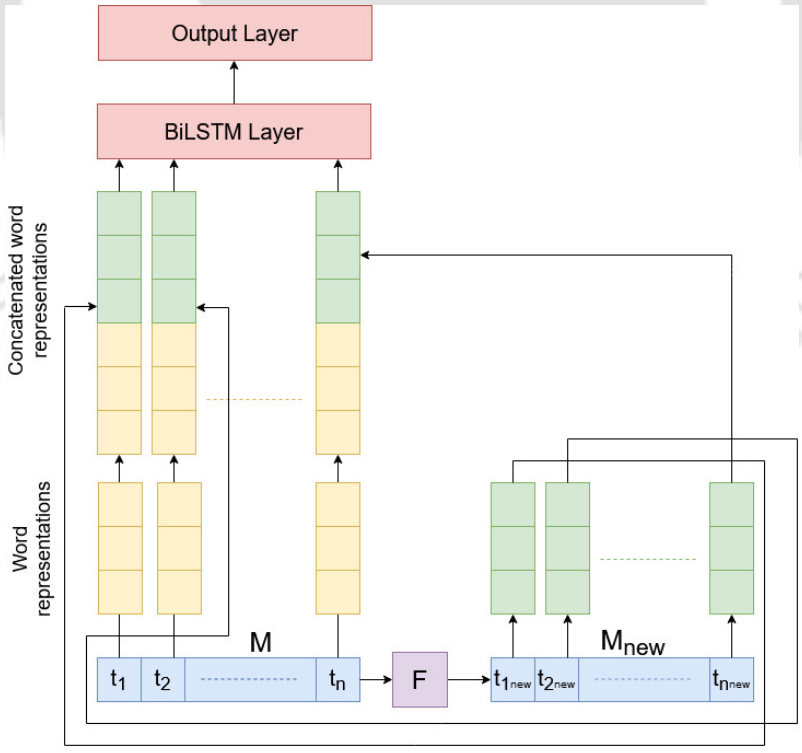


Figure 5.4: Proposed Early Fusion BiLSTM for sentence classification

Bi-directional Long Short Term Memory (BiLSTM) Networks are used in sequential learning problems where information from previous timestamps can be use-

ful in predicting the output at a later time-stamp. Given a message M comprising of terms $t_1, t_2, ..., t_n$ in the sequence, a BiLSTM network takes as input the vector sequence $x_1, x_2, ..., x_n$ where x_i is a vector representation corresponding to a term t_i, $i\epsilon\{1, 2, ..., n\}$. The memory cells in the hidden layers of the BiLSTM network maintains the sequential information in both the forward and the reverse directions. The proposed framework creates a new message M^s_{new} from M based on one of the social characteristics (user's language profile, ego-centric conversations, comment-reply conversation thread or a combination of all). M^s_{new} loses the sequential or the semantic information in M but provides a better representation of the underlying language sense in M. The early fusion BiLSTM combines the advantages in M and M^s_{new} by concatenating the vector representation of each word in M with the vector representation of the corresponding word in M^s_{new}. If $M = t_1, t_2, ..., t_n$ and $M_{new} = t_{1_{new}}, t_{2_{new}}, ..., t_{n_{new}}$such that $t_{i_{new}}$ is the term in M^s_{new} corresponding to the term t_i in M , then the input to the BiLSTM layer in the early fusion BiLSTM is a vector sequence $v' = v_1, v_2, ..., v_n$ where each v_i in v' is obtained by concatenating the vector representations of t_i and $t_{i_{new}}$. Further the length of M and M^s_{new} may vary depending on the value of k. If $k = 1$, then the length of M and M_{new} will be equal. Otherwise the length of M_{new} will be k times the length of M. In such a case, the original message M may be either be padded to match the length of M^s_{new} or each word may be replicated k times

5.4 Experimental Dataset

The desired criteria for selecting the exprimental datasets for this study are discussed in detail in Chapter 3. Among the publicly available datasets, the Twitter dataset reported in Zubiaga et al. [2016] is found suitable for the study. However, while considering the conversational features discussed above, except for the user's information, this dataset has certain issues; (i) it does not have enough number of comment and reply threads, and (ii) ego-centric information is not available. Therefore, in addition to the Twitter dataset, we also consider the locally datasets - SLI-MI and SLI-MII as the experimental datasets for this study. The characteristics of these datasets have been discussed in detail in Chapter 3. However, for the con-

Table 5.2: Characteristics of the experimental datasets. Tweets with multiple class labels are ignored from the Twitter dataset

Facebook			YouTube			TwitterZubiaga et al. [2016]		
Language	#Messages	Avg Length	Language	#Messages	Avg Length	Language	#Tweets	Avg Length
Assamese(as)	5198	15 words	Assamese(as)	2429	10 words	Basque(eu)	754	6 words
Bengali(bn)	1594	12 words	Bengali(bn)	129	8 words	Catalan(ca)	2959	13 words
Hindi(hn)	663	12 words	Hindi(hn)	613	15 words	Glacian(gl)	963	10 words
English(en)	21531	24 words	English(en)	8315	11 words	Spanish(es)	21417	11 words
Boro(br)	467	11 words	-			English(en)	1970	9 words
Karbi(kb)	822	14 words	-			Portuguese(pt)	4320	10 words
Total	30275		Total	11486		Total	32383	
#User	12851		#User	7918		#User	7587	
#Comment-Reply	2344		#Comment-Reply	713		#Comment-Reply	62	
#Ego	2125		#Ego	805		#Ego	-	

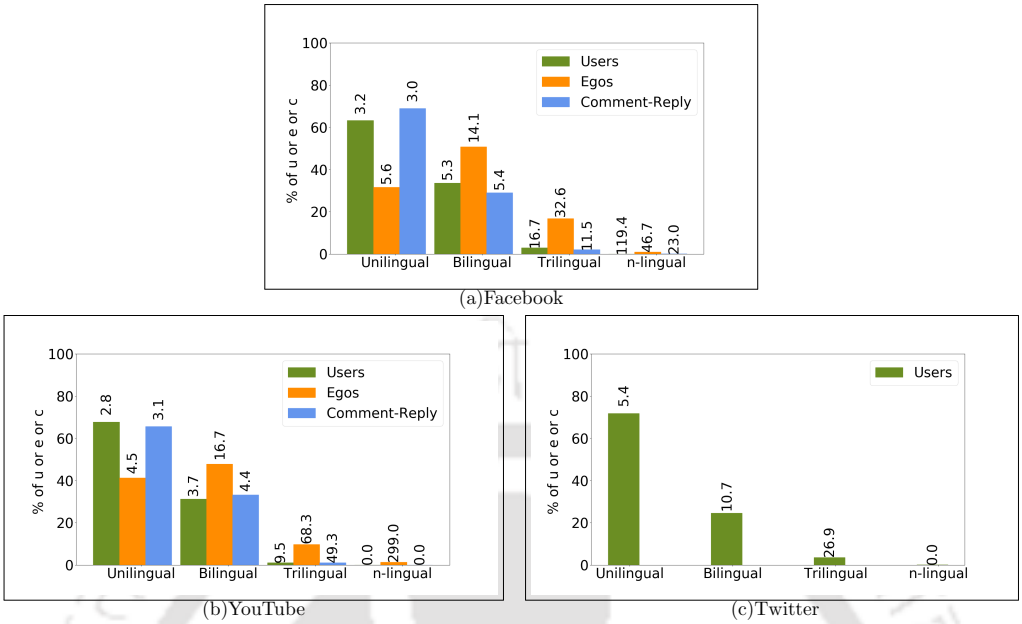
Table 5.3: Probability of a term in a language (given in row) sharing with another language (given in column)

	Facebook							YouTube					Twitter							
	as	en	bn	kb	br	hn	any	as	en	bn	hn	any		en	ca	es	gl	eu	pt	any
as	-	0.25	0.1	0.04	0.03	0.07	0.31	-	0.22	0.04	0.11	0.27	en	-	0.16	0.35	0.06	0.03	0.13	0.40
en	0.17	-	0.06	0.03	0.02	0.05	0.21	0.14	-	0.02	0.1	0.20	ca	0.08	-	0.31	0.08	0.02	0.11	0.34
bn	0.34	0.29	-	0.07	0.06	0.13	0.43	0.39	0.35	-	0.27	0.53	es	0.05	0.09	-	0.07	0.02	0.08	0.21
hn	0.43	0.49	0.24	0.11	0.08	-	0.59	0.22	0.31	0.05	-	0.39	gl	0.08	0.22	0.61	-	0.05	0.27	0.66
br	0.18	0.21	0.11	0.06	-	0.09	0.26	-	-	-	-		eu	0.05	0.06	0.16	0.05	-	0.06	0.18
kb	0.22	0.27	0.11	-	0.06	0.11	0.32	-	-	-	-		pt	0.08	0.15	0.37	0.13	0.03	-	0.42
Average							0.35	Average				0.35	Average							0.37

venience of the reader, characteristics of all three datasets are summarized in Table 5.2. We further analyze two important characteristics related to the experimental datasets :

- **Code sharing characteristics** : Table 5.3 shows probability of a word in a language (in row) having sharing with another language (in column). It clearly shows that there is a high volume of code sharing between different languages for all three datasets (Facebook, YouTube, Twitter). On an average, the probability of a random word in a language sharing with other languages are 0.35, 0.35, and 0.37 for Facebook, YouTube and Twitter datasets respectively.

- **Distribution of the participation of users in multi-lingual conversations** : Figure 5.5 shows the distribution of the participation of users or egos or comment-reply threads in multi-lingual conversations in all the experimen-



(a)Facebook

(b)YouTube

(c)Twitter

Figure 5.5: Percentage of users(u)/egos(e)/comment-reply(c) participating in unilingual/bilingual/trilingual conversations. The n-lingual in above figure represents more than three language conversations.

tal datasets. It clearly shows that more than 35%, 28% and 29% of the users in Facebook, YouTube and Twitter datasets respectively participate in multilingual conversation. Similarly, more than 55% and 40% egos, and more than 20% and 21% of the comment-reply threads in Facebook and YouTube datasets respectively participate in multi-lingual conversations. It further shows that the average number of messages of the users engaged in unilingual conversations is lesser than that of the users engaged in multilingual conversations. It is also true for egos and comment and reply. There are few users, egos and comment-reply threads having more than three languages.

Both the analyses above reveal two important characteristics. First, the datasets considered exhibit high code-similarity across languages. Second, from the distribution of the participation of users in multilingual conversations across all three datasets, it is clear that these datasets also exhibit uncertainty of language usage in a highly multilingual environment. Therefore, the datasets considered in this study are ideal for considering the performances of different language identification techniques in a highly multilingual environment.

Table 5.4: The eleven different classification techniques considered in the study

No.	Notation	Classifier
1	NB	Naive Bayes
2	LR	Logistic Regression
3	SVM	Support Vector Machine
4	DT	Decision Tree
5	RF	Random Forest
6	CNN	Convolutional Neural Network
7	MCNN	Multichannel Convolutional Neural Network proposed in Kim [2014]
8	MLI	Microblog characteristics based language identification proposed in Carter et al. [2013]
9	EFCNN	Early Fusion Convolutional Neural Network shown in Figure 5.3(a)
10	LFCNN	Late Fusion Convolutional Neural Network shown in Figure 5.3(b)
11	Bi-LSTM	Bidirectional Long Short Term Memory
12	EF-BiLSTM	Early Fusion Bidirectional Long Short Term Memory

5.5 Experimental Set-ups

This study considers twelve different classification frameworks shown in Table 5.4 and investigate the effect of the proposed text refinement methods for language identification task. The performance of the classifiers (1 to 5 in Table 5.4) while applying over original text (i.e., without applying the proposed text refinement techniques) are considered as baseline counterparts. Further, the single channel CNN, the multichannel variant of CNN by Kim [2014](referred to as MCNN), Bi-LSTM and MLI have also been considered as baseline. While considering MLI, we have chosen to reproduce the post dependent method of combination of microblog characteristics using beam confidence metric as reported in the paper by Carter et al. [2013]. Due to lack of enough information in our datasets, we have used only two of their characteristics in our implementation - *the user characteristic* and *the comment-reply characteristic*.

Before applying the above classifiers, all the datasets are pre-processed to convert the text to lowercase, and remove special symbols and numbers. For the Twitter dataset, we consider the predefined training and test sets whereas we perform 5-fold cross validations for Facebook and YouTube datasets. All the CNN based classifiers (CNN, MCNN, EFCNN and LFCNN) and the Bi-LSTM classifiers (Bi-LSTM and

EF-BiLSTM) use *word2vec* word embedding. For the NB, LR, SVM, DT and RF classifiers, we consider unigrams features.

Table 5.5: Comparison of macro-average F -score of different classification frameworks over different datasets. * denotes the baseline setup i.e., without text refinement. MCNN (Kim [2014]) have been considered as the baseline setup for EFCNN

Classifiers	Facebook				YouTube				Twitter			
BiLSTM	**0.90**	-	-	-	**0.87**	-	-	-	**0.84**	-	-	-
CNN	**0.90**	-	-	-	0.75	-	-	-	0.80	-	-	-
MCNN (Kim [2014])	**0.90**	-	-	-	0.84	-	-	-	0.83	-	-	-
MLI (Carter et al. [2013])	0.77	-	-	-	0.74	-	-	-	0.54	-	-	-
User Oriented Features												
	*	$(u, \mathcal{S})$	$(u, \mathcal{S}_w)$	$(u, \mathcal{S}_{pro})$	*	$(u, \mathcal{S})$	$(u, \mathcal{S}_w)$	$(u, \mathcal{S}_{pro})$	*	$(u, \mathcal{S})$	$(u, \mathcal{S}_w)$	$(u, \mathcal{S}_{pro})$
NB	0.88	0.89	0.90	**0.95**	0.76	0.75	0.76	0.91	0.76	0.76	0.80	0.86
LR	0.84	0.85	0.85	0.92	0.81	0.81	0.81	**0.93**	0.78	0.79	0.80	0.86
EFCNN	**0.90**	0.90	0.91	0.93	0.84	0.79	0.79	0.82	0.80	0.81	0.82	0.84
EF-BiLSTM	**0.90**	**0.92**	**0.92**	0.94	**0.87**	**0.86**	**0.82**	0.89	**0.84**	**0.84**	**0.85**	**0.87**
Ego Oriented Features												
	*	$(e, \mathcal{S})$	$(e, \mathcal{S}_w)$	$(e, \mathcal{S}_{pro})$	*	$(e, \mathcal{S})$	$(e, \mathcal{S}_w)$	$(e, \mathcal{S}_{pro})$				
NB	0.88	0.89	0.90	**0.93**	0.76	0.76	0.77	0.78				
LR	0.84	0.85	0.86	0.90	0.81	0.81	0.81	0.82				
EFCNN	**0.90**	**0.91**	**0.92**	**0.93**	0.84	0.77	**0.85**	0.78				
EF-BiLSTM	**0.90**	**0.91**	**0.92**	**0.93**	**0.87**	**0.86**	0.76	**0.86**				
Comment-Reply Oriented Features												
	*	$(c, \mathcal{S})$	$(c, \mathcal{S}_w)$	$(c, \mathcal{S}_{pro})$	*	$(c, \mathcal{S})$	$(c, \mathcal{S}_w)$	$(c, \mathcal{S}_{pro})$				
NB	0.88	0.88	0.89	**0.96**	0.76	0.76	0.77	0.92				
LR	0.84	0.84	0.84	0.94	0.81	0.81	0.82	**0.93**				
EFCNN	**0.90**	0.90	0.90	0.95	0.84	0.75	0.80	0.86				
EF-BiLSTM	**0.90**	**0.91**	**0.93**	0.95	**0.87**	**0.86**	**0.86**	0.91				
Combination of all Features												
	*	$(uec, \mathcal{S})$	$(uec, \mathcal{S}_w)$	$(uec, \mathcal{S}_{pro})$	*	$(uec, \mathcal{S})$	$(uec, \mathcal{S}_w)$	$(uec, \mathcal{S}_{pro})$				
NB	0.88	0.88	0.90	**0.95**	0.76	0.76	0.77	0.91				
LR	0.84	0.86	0.86	0.94	0.81	0.82	0.82	**0.93**				
EFCNN	**0.90**	0.90	0.90	0.94	0.84	0.81	**0.85**	0.89				
EF-BiLSTM	**0.90**	**0.92**	**0.92**	0.94	**0.87**	**0.85**	**0.85**	0.90				
$\mathcal{S}$ dominance	-	0%	28%	100%	-	6%	28%	84%	-	0%	50%	100%
p value		6.81E-06	3.50E-06	3.35E-08		0.01	0.07	0.002		0.09	0.02	0.02

Table 5.6: Macro average F-measure score of different classifiers for text refinement using $S_{pro}(t_i, t_j)$ similarity estimates over different datasets. u =user,e =ego,c=comment-reply,con =content only, $con1$ =CNN and $con2$ =MCNN.

Dataset	LFCNN						Decision Tree					Random Forest					SVM				
	con_1	con_2	u	e	c	uec	con	u	e	c	uec	con	u	e	c	uec	con	u	e	c	uec
Facebook	0.90	0.90	0.94	0.93	0.95	0.94	0.73	0.86	0.81	0.89	0.88	0.79	0.89	0.85	0.91	0.89	0.38	0.50	0.45	0.54	0.47
YouTube	0.75	0.84	0.87	0.79	0.86	0.90	0.76	0.84	0.74	0.84	0.87	0.76	0.84	0.75	0.86	0.86	0.41	0.56	0.45	0.58	0.64
Twitter	0.80	0.83	0.85	-	-	-	0.70	0.79	-	-	-	0.70	0.78	-	-	-	0.75	0.76	-	-	-

5.6 Results and Observations

Tables 5.5 and 5.6 show the macro-average F-score performance of the different classifiers using different social conversational features (user, ego, comment-reply and combination) and different text similarity metrics for text refinement over different datasets. Among the baseline setups, Bi-LSTM outperforms the other baseline classifiers over YouTube and Twitter datasets with an F-score of 0.87 and 0.84 respectively. On the other hand, Bi-LSTM, CNN and MCNN provide comparable performance of 0.90 F score in the Facebook dataset. In contrast to Carter et al. [2013] where the microblog characteristics add to the performance of the language identification system as compared to the content only set-up, their method underperform in our case. The may be due to the reason that at the core of their method is the n-gram text similarity method that is based on the most frequent n-grams. Since our data exhibits significant amount of code-mix, the n-gram text similarity method produces inaccurate results. Also, in contrast to Kim [2014] where MCNN under-performs CNN in most cases, in our dataset, MCNN shows either comparable or better performance than CNN.

Next we investigate the performance of our proposed methods on language identification. The experimental analysis reported in this section attempts to understand the followings.

- Effect of the proposed social conversational features; user, ego and comment-reply.
- Effect of the proposed text refinement similarity estimates.
- Language-wise detail analysis

- Strength of proposed refinement methods on code-sharing and unseen text

5.6.1 Response of text refinement methods

As mentioned in section 5.3.1, the u, e, c, and uec in Tables 5.5 and 5.6 refer to user, ego, comment-reply and their combination. For each of these features, we further deploy three similarity estimates S, S_w and S_{pro} (defined in section 5.3.1). The tables 5.5 and 5.6 clearly show that all the classifiers with S_{pro} (except EFCNN(u, s_{pro}), EFCNN(e, S_{pro}), EF-BiLSTM(e, S_{pro}), DT(e, S_{pro}) and RF(e, S_{pro}) in YouTube dataset) outperform their content based counterparts for all datasets. To compare the responses of the proposed three similarity metrics, we compute the dominance of the similarity metrics where the dominance of a similarity metric indicate the percentage by which the target similarity metric outperforms the other similarity metrics. We observe the following :

- S_{pro} outperforms the other similarity metrics with a dominance of 100%, 84% and 100% over the Facebook, YouTube and Twitter datasets respectively.
- S_w outperforms the other similarity metrics with a dominance of 28%, 28% and 50% over the Facebook, YouTube and Twitter datasets respectively.
- S is the least performing similarity estimate with a dominance of 6% in the YouTube dataset and a dominance of 0% in the Facebook and Twitter datasets.
- From the above points, it is clear that S_{pro} is the best performing similarity estimate followed by S_w and S.

Marginal differences in performance of different classifiers, when S similarity estimate and content are used, may be due to *data under specificity* (i.e., $|\mathcal{D}^s| << |\mathcal{D}|$ and $|\mathcal{V}^s| << |\mathcal{V}|$ are small). Because of this, following two cases are possible; (i) chances of selecting a shared code as refined text is high, and (ii) chances of getting repeated texts for different messages is high. The S_w similarity estimate tries to reduce the effect of the above two cases by taking the likelihood of not sharing words across languages into account. Since the number of candidate words are still limited, variations in the newly generated text may still be less resulting in only

marginal improvement over S.

The above two issues are addressed in the S_{pro} similarity estimate by considering the language profile of the conversational features into account. The profile based term co-occurrence estimate has two advantages. First, the number of candidate words to be considered for the estimate is increased i.e., $|\mathcal{D}_l| > |\mathcal{D}^s|$. Second, search space for the co-occurred term is reduced i.e., $|\mathcal{L}^s| < |\mathcal{L}|$. As evident in the above discussion, S_{pro} outperforms the other similarity estimates. It clearly indicates that the modified messages using S_{pro} similarity estimate can effectively capture the language characteristics of underlying messages.

5.6.2 Effect of social conversational features

As we observed above, S_{pro} outperforms other similarity estimates, analysis reported in this section considers only the S_{pro} similarity metric. We first present performance comparison of different social conversational features.

- *Comment and reply* feature outperforms *user* and *ego* features in 87.5 % and 100% of the cases.
- *User* outperforms *ego* feature in 87.5% of the cases.

It is evident that text refinement using the language distribution associated with *comment and reply* social features can capture underlying language of the message better as compared to *user* and *ego*. Further we present performance comparisons of individual feature with their baseline counterparts.

- For all classifiers over all datasets, *user* based social feature outperforms its content based counterparts in 95% of the cases. Except for DT, RF and SVM, all other classifiers with user features and S_{pro} estimates outperform BiLSTM, MCNN and MLI in almost all the cases.
- Similarly, classifiers with *ego-centric feature* also outperform its content based counterparts in 81.25% cases.
- As observed in the above discussions, *comment and reply feature* dominates both the *user* and the *ego centric* features.

- Combining other social features with comment-reply does not show promising influence.

- Among the three features, *ego feature* is found to be weakest.

Overall, combination of the *comment-reply* feature and S_{pro} similarity estimate provide the best performance in almost all experimental setup. We are able to achieve a maximum improvement of 12%, 20% , 13% as compared to its content based counterpart over Facebook, YouTube and Twitter (NB for YouTube, LR for Facebook and NB classifier on Twitter dataset). Among the baseline methods, BiLSTM outperforms the others. Our proposed approach shows a maximum improvement of 5.5%, 6.8% and 2.3% for the Facebook, YouTube and Twitter datasets respectively.

5.6.3 Statistical significance of our proposed methods

In the discussions in sections 5.6.1 and 5.6.2, we compare the performance of our proposed methods compared to the baseline methods. To validate the strength of our comparisons, we perform significance tests using t-test. Table 5.5 shows the respective p values of our observations obtained using different social characteristics and different similarity metrics. With a confidence interval of 95%, we find that except for two cases, our proposed methods significantly outperform the baselines.

5.6.4 Analysis by language

Table 5.7 shows the language wise performance of the baselines and the three best performing classification frameworks over the three datasets. To compare the performance across the languages, we also compute the dominance of the languages where dominance of a language is the percentage by which the target language outperforms the other languages. To co-relate the performance of the languages with respect to the dataset size, average message length and code-sharing characteristics of the dataset, we compute the Spearman's coefficient between the dominance of the languages and each of the these three factors. For the Facebook dataset, we obtain a Spearman's coefficient of 0.714, 0.829 and 0.657 with the number of examples in each language, average message length in each language and the code-sharing probability respectively. Likewise for the YouTube dataset, we obtain a Spearman's coefficient

of 1, 0.4 and 1 and for the Twitter dataset, we obtain a Spearman's coefficient of 1, 0.6 and 0.26 with the number of examples, average message length and the code-sharing probability respectively. The positive Spearman's coefficient obtained indicates an increasing monotonic relationship between the language dominance and the three factors mentioned above. On an average across the three datasets, we find that the language dominance is highly co-related to the number of examples in the language followed by the code-sharing characteristics and lastly by the average message length.

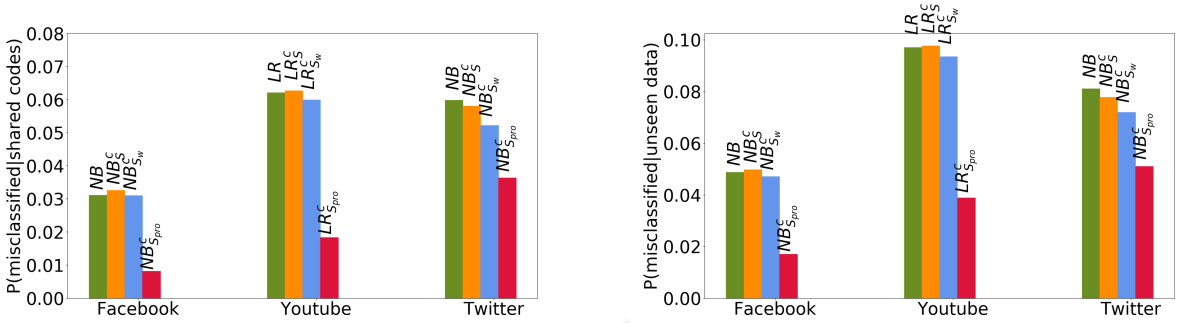
From table 5.7, we see that English shows the best performance in both the Facebook and the YouTube datasets with a dominance percentage of 99% and 100% while Spanish shows the best performance in the Twitter dataset with a dominance of 95%. If we look at the dataset characteristics in table 5.2, we see that English has the maximum number of examples in the Facebook and the YouTube datasets while Spanish has the most number of examples in the Twitter dataset.

5.6.5 Effect of S_{pro} estimate on code sharing

Figure 5.6(a) compares the probability of a message having shared code being misclassified before and after fusing refined features using *comment-reply* social features. *Comment-reply* is considered as it dominates other social features. It is evident that misclassification rate is reduced after fusing refined features. Among the three similarity estimates, S_{pro} provides the maximum reduction over it's content based counterparts - 73.8% on Facebook dataset, 70.38% on YouTube dataset and 39.16% on Twitter dataset.

5.6.6 Effect of S_{pro} estimate on unseen text

Figure 5.6(b) compares the probability of a message having unseen text being misclassified before and after fusing refined features using *comment-reply* social features.Even for unseen text, S_{pro} is able to significantly reduce the misclassification rate - 64.8% on Facebook dataset,60.0% on YouTube dataset and 37.4% on Twitter dataset.



(a) Shared Codes (b) Unseen

Figure 5.6: Probability of misclassification of a message with (a)shared codes and (b) unseen text using NB, NB^c_S, $NB^c_{S_w}$ and $NB^c_{S_{pro}}$ over Facebook dataset, LR, LR^c_S, $LR^c_{S_w}$ and $LR^c_{S_{pro}}$ over YouTube dataset and NB, NB^u_S, $NB^u_{S_w}$ and $NB^u_{S_{pro}}$ over Twitter dataset.

5.7 Summary

In this chapter, we propose three social features (*user*, *ego* and *comment-reply*) and three similarity estimates (*term co-occurrence, weighted term co-occurrence* and *global language profile based term co-occurrence score*) for refining a noisy microblog for enhancing language identification performance in a highly multilingual conversation. The proposed social features and similarity estimates are incorporated with different classification methods using early feature fusion and late feature fusion methods. From various experimental observations, it is observed that social features help in enhancing the language identification performance. Among these three social features, *comment-reply* provides the highest improvement. Further, among the three similarity estimates, *global language profile based term co-occurrence score* provides the highest improvement. Overall, using *comment-reply* feature and *global language profile based term co-occurrence score* similarity estimate, we are able to improve language identification performance up to 12%, 20% , 13% as compared to its content based counterpart for Facebook, YouTube and Twitter. Compared to the best performing baseline, the proposed approach shows a maximum improvement of 5.5%, 6.8% and 2.3%.

Despite the significant performance of the proposed methods, one drawback is the dependence on the social conversational features. Therefore in scenarios where not enough social conversational information surrounding a conversational message

is available, the methods may show a drop in the performance. In situations where the social conversational information lacks completely, the performance obtained by the proposed methods will be equivalent to that obtained considering the textual content only. In such cases, strategies to leverage pretrained word embeddings to address out-of-vocabulary and shared vocabulary issues may be explored. Another limitation of this work is not identifying multilingual messages or not identifying languages at the word-level. In a highly multilingual conversational environment where conversations exhibit high degree of code-mixing , identifying languages at the word-level is an extremely important task. Therefore, in the next chapter, Chapter 6, word-level language identification using sentence-level language information is addressed.

Table 5.7: Language wise performance comparison of different classifiers using F-measure over different datasets.

Classifier	Facebook						YouTube				Twitter					
	as	en	bn	kb	br	hn	as	en	bn	hn	en	ca	es	gl	eu	pt
CNN	0.95	**0.99**	0.91	0.91	0.82	0.81	0.93	0.98	0.23	0.89	0.87	0.91	0.96	0.41	0.73	0.90
MCNN (Kim [2014])	0.95	**0.99**	0.90	0.93	0.85	0.78	0.93	0.98	0.57	0.89	0.88	0.91	0.96	0.52	0.76	0.93
MLI Carter et al. [2013]	0.77	0.92	0.73	0.92	0.82	0.49	0.84	0.94	0.44	0.75	0.39	0.58	0.88	0.26	0.26	0.87
BiLSTM	0.94	0.98	0.88	0.91	0.86	0.82	0.92	0.98	0.68	0.90	0.89	0.92	0.96	0.59	0.78	0.92
EFCNN$^{u}_{S}$	0.95	**0.99**	0.91	0.92	0.83	0.80	0.93	0.98	0.37	0.89	0.87	0.91	0.96	0.47	0.72	0.90
EFCNN$^{u}_{S_w}$	0.95	**0.99**	0.93	0.93	0.85	0.82	0.93	0.98	0.37	0.90	087	0.93	0.96	0.50	0.75	0.91
EFCNN$^{u}_{S_{pro}}$	0.97	**0.99**	0.95	0.95	0.89	0.85	0.94	0.98	0.45	0.94	0.89	0.95	**0.97**	0.53	0.76	0.97
EFCNN$^{e}_{S}$	0.95	**0.99**	0.91	0.93	0.87	0.81	0.94	0.98	0.27	0.88						
EFCNN$^{e}_{S_w}$	0.95	**0.99**	0.93	0.93	0.88	0.83	0.94	0.98	0.56	0.91						
EFCNN$^{e}_{S_{pro}}$	0.96	**0.99**	0.94	0.94	0.88	0.82	0.94	0.98	0.29	0.90						
EFCNN$^{c}_{S}$	0.94	**0.99**	0.90	0.92	0.86	0.80	0.93	0.98	0.20	0.88						
EFCNN$^{c}_{S_w}$	0.95	**0.99**	0.92	0.92	0.84	0.80	0.94	0.98	0.40	0.90						
EFCNN$^{c}_{S_{pro}}$	**0.98**	**0.99**	0.97	0.96	0.91	0.88	0.96	0.98	0.55	0.94						
EFCNN$^{uec}_{S}$	0.94	**0.99**	0.90	0.93	0.86	0.80	0.93	0.98	0.46	0.88						
EFCNN$^{uec}_{S_w}$	0.95	**0.99**	0.92	0.92	0.84	0.81	0.94	0.98	0.59	0.90						
EFCNN$^{uec}_{S_{pro}}$	0.97	**0.99**	0.97	0.96	0.91	0.87	0.96	**0.99**	0.67	0.95						
LR	0.89	0.97	0.82	0.90	0.78	0.70	0.87	0.96	0.55	0.85	0.82	0.88	0.94	0.48	0.73	0.88
LR$^{u}_{S}$	0.89	0.96	0.82	0.91	0.78	0.68	0.87	0.96	0.58	0.84	0.81	0.89	0.94	0.48	0.75	0.88
LR$^{u}_{S_w}$	0.89	0.96	0.84	0.91	0.80	0.70	0.87	0.96	0.56	0.84	0.82	0.88	0.94	0.54	0.74	0.90
LR$^{u}_{S_{pro}}$	0.94	0.96	0.93	0.95	0.89	0.83	0.94	0.98	0.86	0.93	0.86	0.94	0.96	0.60	**0.83**	**0.98**
LR$^{e}_{S}$	0.89	0.96	0.82	0.92	0.81	0.70	0.88	0.96	0.58	0.84						
LR$^{e}_{S_w}$	0.89	0.96	0.86	0.93	0.82	0.69	0.87	0.96	0.58	0.85						
LR$^{e}_{S_{pro}}$	0.93	0.97	0.93	0.96	0.87	0.76	0.88	0.96	0.59	0.85						
LR$^{c}_{S}$	0.89	0.97	0.82	0.90	0.79	0.70	0.87	0.96	0.57	0.84						
LR$^{e}_{S_w}$	0.89	0.97	0.83	0.90	0.78	0.69	0.88	0.96	0.58	0.85						
LR$^{c}_{S_{pro}}$	0.96	0.98	0.96	0.97	0.90	0.90	0.95	0.98	**0.87**	0.94						
LR$^{uec}_{S}$	0.90	0.96	0.86	0.93	0.81	0.70	0.87	0.96	0.59	0.84						
LR$^{uec}_{S_w}$	0.89	0.96	0.86	0.93	0.81	0.71	0.87	0.96	0.61	0.85						
LR$^{uec}_{S_{pro}}$	0.95	0.96	0.96	0.96	0.90	0.87	0.95	0.98	0.85	0.94						
NB	0.93	**0.99**	0.86	0.93	0.84	0.77	0.93	0.98	0.23	0.91	0.89	0.93	0.96	0.20	0.54	0.92
NB$^{u}_{S}$	0.93	**0.99**	0.90	0.94	0.83	0.74	0.93	0.98	0.20	0.90	0.89	0.93	0.96	0.21	0.66	0.93
NB$^{u}_{S_w}$	0.94	**0.99**	0.92	0.94	0.84	0.76	0.93	0.98	0.24	0.91	0.89	0.94	0.96	0.41	0.67	0.93
NB$^{u}_{S_{pro}}$	0.96	**0.99**	0.94	0.96	**0.94**	0.88	0.96	**0.99**	0.74	**0.97**	**0.91**	**0.97**	**0.97**	0.51	0.81	**0.98**
NB$^{e}_{S}$	0.92	**0.99**	0.88	0.94	0.86	0.73	0.93	0.98	0.24	0.90						
NB$^{e}_{S_w}$	0.93	**0.99**	0.90	0.95	0.87	0.78	0.93	0.98	0.26	0.90						
NB$^{e}_{S_{pro}}$	0.95	**0.99**	0.93	0.96	0.90	0.83	0.93	0.98	0.31	0.91						
NB$^{c}_{S}$	0.93	**0.99**	0.86	0.93	0.84	0.88	0.93	0.98	0.22	0.90						
NB$^{e}_{S_w}$	0.93	**0.99**	0.87	0.93	0.84	0.89	0.93	0.98	0.27	0.91						
NB$^{c}_{S_{pro}}$	0.98	**0.99**	**0.98**	**0.98**	**0.94**	**0.92**	**0.97**	**0.99**	0.75	0.96						
NB$^{uec}_{S}$	0.93	0.98	0.91	0.94	0.85	.70	0.93	0.98	0.23	0.90						
NB$^{uec}_{S_w}$	0.93	0.98	0.93	0.95	0.87	0.75	0.93	0.98	0.27	0.91						
NB$^{uec}_{S_{pro}}$	0.97	0.99	0.96	0.97	0.92	0.89	**0.97**	**0.99**	0.73	**0.97**						
BiLSTM$^{u}_{S}$	0.95	0.98	0.92	0.94	0.88	0.84	0.92	0.97	0.67	0.89	0.88	0.92	0.96	0.57	0.77	0.92
BiLSTM$^{u}_{S_w}$	0.95	**0.99**	0.93	0.92	0.89	0.84	0.92	0.97	0.59	0.91	0.88	0.94	0.96	0.60	0.77	0.93
BiLSTM$^{u}_{S_{pro}}$	0.97	**0.99**	0.95	0.93	0.89	0.88	0.94	0.98	0.68	0.94	0.89	0.95	**0.97**	**0.63**	0.80	0.97
BiLSTM$^{e}_{S}$	0.95	0.98	0.93	0.94	0.91	0.82	0.92	0.98	0.61	0.90						
BiLSTM$^{e}_{S_w}$	0.94	0.98	0.90	0.91	0.82	0.78	0.92	0.98	0.69	0.92						
BiLSTM$^{e}_{S_{pro}}$	0.96	**0.99**	0.95	0.96	0.91	0.85	0.93	0.98	0.65	0.92						
BiLSTM$^{c}_{S}$	0.95	**0.99**	0.92	0.93	0.90	0.86	0.92	0.98	0.69	0.92						
BiLSTM$^{c}_{S_w}$	0.95	**0.99**	0.95	0.94	0.89	0.85	0.91	0.97	0.65	0.88						
BiLSTM$^{c}_{S_{pro}}$	0.97	**0.99**	0.97	0.96	0.91	0.87	0.95	**0.99**	0.78	**0.97**						
BiLSTM$^{uec}_{S}$	0.95	**0.99**	0.92	0.94	0.91	0.82	0.91	0.98	0.65	0.90						
BiLSTM$^{uec}_{S_w}$	0.95	**0.99**	0.93	0.95	0.85	0.83	0.90	0.97	0.66	0.90						
BiLSTM$^{uec}_{S_{pro}}$	0.97	**0.99**	0.97	0.97	0.91	0.86	0.94	0.98	0.71	0.94						
Dominance	65%	99%	41%	61%	19%	1%	60%	100%	0%	34%	40%	67%	95%	0%	18%	70%

6

Word-Level Language Identification of Code-Mixed Social Media Content

Chapter 5 addresses sentence-level language identification over code-mixed social media content in a highly multilingual environment. However, sentence-level language information alone is often not sufficient to extract complete information from code-mixed content. It is also important to identify languages at the word-level. Word-level language identification of code-mixed content has already been addressed in literature. However, most existing studies are not suitable for low resource languages due to the unavailability of required resources like dictionaries, annotated resources, parsers, taggers, etc. This chapter, therefore, aims to address the problem of word-level language identification of code-mixed social media content in a highly multilingual environment with special emphasis on low resource languages. Different word-level language identification strategies are proposed and their performances are evaluated over a corpus of code-mixed transliterated text discussed in Chapter 3. From the experimental observations, it is evident that the proposed framework can effectively handle out-of-vocabulary and shared vocabulary problem which are major challenges for language identification over social media text in a multilingual environment. The proposed framework also uses minimal resources making it suitable for low resource languages.

6.1 Introduction

Identifying languages at the word-level is imperative for processing useful information from code-mixed social media text. While ALI in the regular text domain (e.g. news articles, web documents, etc.) is restricted to document or sentence-level language identification, in the social media domain, this is often not enough. Phonetic typing (transliterated text) and code-mixing are common phenomena in social media. Therefore, to extract important information from the text, it is necessary to identify languages at the word-level. For example, consider the text *Tumak sobe hate kore [Everybody hates you]* and *hate kame koribo lagibo [We have to work hands down]*. In the first case, *hate* is an English word embedded into a transliterated Assamese sentence while in the second case, *hate* is a transliterated Assamese word. For an application like sentiment analysis, it is important to identify that the word *hate* in the first case in an English word while the second is not.

Most existing literature in word-level language identification suggests the use of different tools and resources for language identification like dictionaries (Barman et al. [2014]), annotated resources (Barman et al. [2014], Jaech et al. [2016]), monolingual corpora (King and Abney [2013]), transliteration tools (Singh et al. [2018a]) etc. However, these approaches have a few shortcomings. First, languages of words in a multilingual social media environment are context dependent. As such, traditional methods like dictionary look-up are not suitable. Considering the variations in word forms in a multilingual social media environment owing to factors such as mis-spellings, creative spellings, transliteration etc., large amounts of word-level annotated data are required to capture such variations. However, obtaining word-level annotations are very expensive in terms of time and effort. Second, approaches that do not use word-level annotated data make use of clean monolingual corpora in individual languages (King and Abney [2013]). However, obtaining clean monolingual corpora is not available in transliterated text.

Motivated by these challenges, this chapter aims to address the problem of word-level language identification for social media text in low resource languages in a highly multilingual environment. The work is motivated by the fact that obtaining sentence-level annotations is far less expensive than obtaining word-level anno-

tations. Hence, more feasible for low resource languages. Therefore, this chapter proposes a word-level language identification framework using sentence-level annotations. The proposed method focuses on learning word-level representations by exploiting sentence-level structural properties to build suitable word-level language classifiers. Since social media text is noisy and evolving in nature, an annotated corpus is likely to miss out irregular word forms, new vocabulary or out-of vocabulary words. Therefore, the objective in this chapter is to make use of the structural properties in the sentences to capture the language characteristics such that language labels can be obtained for new evolving words.

For our experimental study, we consider the datasets SLI-MI and WLI-M described in Chapter 3. Different word-level language identification are proposed and evaluated and their pros and cons are investigated. The experimental results reveal that the proposed framework can address out-of-vocabulary and shared vocabulary problem.

6.2 Related Studies

Word-level language identification has already been explored in literature. Both supervised and unsupervised approaches for word-level language identification have been explored.

Supervised approaches for word-level language identification have been addressed by Barman et al. [2014] where authors experiment with different methods like dictionary look-up, supervised learning approach (SVM classifier) and sequence classification approach (CRF) over a set of transliterated Facebook comments. Nguyen and Doğruöz [2013] also explore different methods for word-level language identification like dictionary look-up, language models, logistic regression classifier and conditional random fields classifier. Features like modified edit distance, word frequency, character n-grams and part of speech tag and language tag of neighboring words for word-level language identification have been used by Jhamtani et al. [2017]. Vyas et al. [2014] create a corpus of Facebook comments and also address back-transliteration, normalization and part-of-speech tagging in addition to language identification. CRF model using features like word features (word vec-

tor of the current word and that in the context), spelling features and intra-word features are used by Xia [2016]. CRF based system for word-level language identification has also been used by Chittaranjan et al. [2014]. Sequence classification using RNN have been used by Samih et al. [2016]. Multichannel CNN combined with a Bi-LSTM-CRF module has been used by Mandal and Singh [2018]. Sequence to sequence model has also been used by Jurgens et al. [2017]. Instead of sequence models, Zhang et al. [2018] propose a two-stage model wherein in the first stage a distribution over the languages for a given word is predicted followed by a decoder in the second stage which along with the language distribution predicted in the first stage also takes into consideration global constraints over the entire sentence.

Unsupervised or weakly supervised approaches have been explored in Gella et al. [2014], King and Abney [2013], Rijhwani et al. [2017]. Gella et al. [2014] create a synthetic code-mixed language identification dataset from a collection of monolingual text and build binary classifiers for each language. Monolingual text has also been used by King and Abney [2013] where they use a collection of monolingual texts and employ weakly supervised and semi-supervised methods for word-level language identification in multilingual documents. Rijhwani et al. [2017] also use an unsupervised model using monolingual corpora and HMM for word-level language identification.

As evident from the discussion above, most existing studies make use of different resources like dictionaries, annotated resources and transliteration tools which is not easy to obtain for low resource languages. Instead of using word-level annotated data, King and Abney [2013] and Rijhwani et al. [2017] use monolingual corpus to obtain word-level language labels in multilingual documents. However, this study concerns with word-level language identification over a set transliterated text. Obtaining clean transliterated monolingual corpus is not possible because (i) transliteration in social media is noisy and irregular and (ii) developing accurate transliteration data and tools for such noisy data is an additional overhead. Therefore, this study proposes to obtain word-level language annotations using sentence-level annotations. Since obtaining sentence-level annotations is less expensive than obtaining word-level annotations which is a sequential task, the proposed method can quickly be adopted for languages with not enough resources.

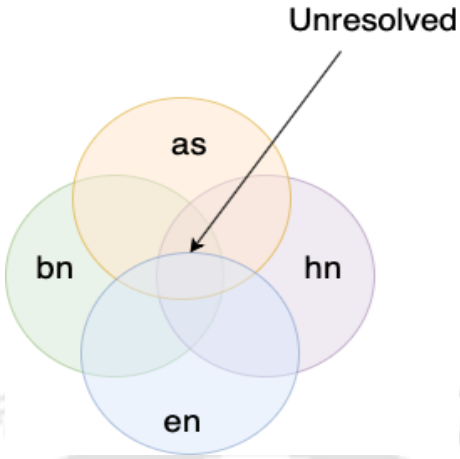


Figure 6.1: Obtaining word-level annotations using sentence-level annotations

6.3 Proposed Framework

This section presents the proposed framework to address word-level language identification using sentence-level annotations. The proposed method focuses on following key points ; (i) obtaining word-level language information from the sentence-level information (ii) language identification using global semantic similarity and (iii) language identification using local contextual similarity.

6.3.1 Obtaining word-level language information from the sentence-level information

Word-level language identification frameworks in the traditional set-up make use of word-level annotated data or monolingual corpora. While word-level annotated data is expensive to obtain, it is also not possible to obtain clean monolingual transliterated text for low resource languages. A comparatively less expensive task is to obtain annotations at the sentence-level. Therefore, the proposed model uses a dataset of sentences annotated with their corresponding languages to obtain word-level annotations.

In the first step, the model uses the sentence-level language information to group the words into two disjoint sets - *resolved set* and *unresolved set*. The *resolved set* of words are those words that occur only in sentences of one particular language and therefore these words can be assigned the language of the corresponding sentences. The *unresolved set* of words on the other hand is the set of words that

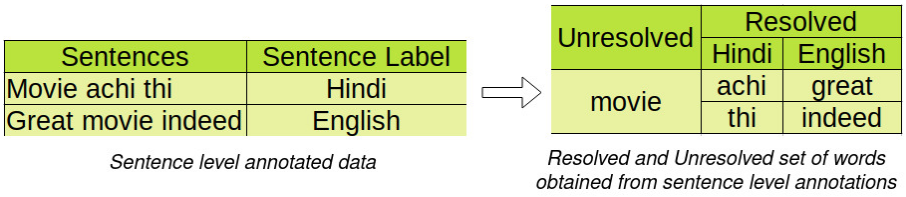


Figure 6.2: Example illustrating use of sentence-level annotations to obtain word-level annotation

occur in sentences of more than one language. Therefore, it is not clear at this stage whether the words belong to one particular language and have been borrowed or embedded in sentences of other languages or whether the words are valid in multiple languages. This is shown in Figure 6.1.

Let $\mathcal{S}$ be the set of annotated sentences, $\mathcal{V}$ be the set of all vocabulary in the annotated sentences and let $\mathcal{L}$ be the set of languages with which the sentences have been annotated. Let l_v be the set of languages of the sentences in which the word v is occurring. Then if $|l_v| = 1$, i.e., the word v occurs in sentences of only one particular language, then we can safely assume that the word v belongs to language l. However, if $|l_v| > 1$, then the language of the word cannot be inferred at this stage. This word may be a legitimate word in more than one language. This word can also be a borrowed word belonging to another language. With $|l_v|$, $\mathcal{V}$ can be divided into two disjoint subsets ; (i) *resolved* set $\mathcal{R} = \{v : |l_v| = 1, v \in \mathcal{V}\}$ and (ii) *unresolved* set $\mathcal{U} = \{v : |l_v| > 1, v \in \mathcal{V}\}$ where $\mathcal{R} \cap \mathcal{U} = \phi$ and $\mathcal{R} \cup \mathcal{U} = \mathcal{V}$. An example is shown in Figure 6.2. Assuming a hypothetical corpus comprising two sentences with the corresponding sentence-level language information, words that occur only in sentences of one particular language are categorized as belonging to that particular language and put into the resolved set $\mathcal{R}$. Language of the word *movie* that occur in sentences of multiple languages are not known at this stage and hence put into the Unresolved set $\mathcal{U}$.

From manual verification over a selected set of words, we found that about 95% of the words in the resolved set $\mathcal{R}$ are correct. Considering this high accuracy, this paper focuses on resolving the words in the unresolved set $\mathcal{U}$.

6.3.2 Language identification using global semantic similarity

Word embeddings like word2vec (Mikolov et al. [2013]) or Glove (Pennington et al. [2014]) is a well explored area for generating word representations from a given text corpus. They generate low dimensional vector representations of words that capture the syntactic and semantic characteristics of the words in the corpus by making use of the distributional characteristics of words within the corpus. Therefore the similarity or dis-similarity between two words can be determined by their vector representations. Due to their ability to capture latent relationships between the words, these representations have been found to be very useful in various text processing applications.

In this study, we make use of these representations to determine the language of a new word. The intuition is that since these representations capture the semantic similarity by using distributional characteristics in a text corpus, the language of a new word can be determined by measuring the similarity of its representation with representations of other words whose languages have already been determined. It is expected that any word will be semantically more similar to other words of the same language than words belonging to different language.

For obtaining the word representations, this study uses the word2vec (skip-gramm) word embedding model. The word embeddings obtained are then used as features for training different classification frameworks. This study explores three different popularly used classification frameworks - Support Vector Machines (SVM), Naive Bayes (NB) and Convolutional Neural Networks (CNN). While building the classifier for word-level language identification, we have considered the words in the resolved set $\mathcal{R}$ as the training samples. These classifiers are used to resolve the words in the unresolved set $\mathcal{U}$. Since $\mathcal{U} \cap \mathcal{R} = \phi$, none of the unresolved examples in $\mathcal{U}$ are present in the training set.

6.3.3 Language Identification using local contextual similarity

An important issue that needs to be addressed while identifying languages of words is to disambiguate the language of words based on their context. This is particularly important when the languages in consideration are similar in terms of vocabulary

used. Also, with transliterated text, the transliterated version of a word is often confused with valid words in a different language. This makes it important to consider the local context of the word in addition to the global semantics of the word. Let w be the target word that needs to be classified. Then, to classify the language of a word based on its local contextual similarity, a word is represented by a vector which is a concatenation of the vector representation of the target word w and the vector representations of the words in the context of w, i.e., the words on either side of w within a fixed window size h. As discussed in Section 6.3.2, the word representations are obtained using the word2vec (skip-gram) model and words in the resolved set $\mathcal{R}$ are used as training samples to classify the words in the unresolved set $\mathcal{U}$.

6.4 Datasets and Experimental Set-up

The objective of this chapter is to study word-level language identification for low resource languages in a highly multilingual environment. Though a few word-level language identification datasets have been made available online, such datasets directly provide the word-level language information without providing the sentence-level language information. Since such datasets do not fit into our proposed framework, we therefore consider the locally generated datasets discussed in Chapter 3. The dataset of annotated sentences, SLI-MI is used to obtain the set of *resolved* and *unresolved* words. Further, the word-level dataset, WLI-M, described in Chapter 3 (Table 3.3) is divided into training and evaluation set. The training set is used to build a word-level classifier with a corpus annotated with word-level language information and compare the performance with the proposed framework using sentence level information. It should be noted that WLI-M only consists of words from the *unresolved* set.

As mentioned in Section 6.3, three different classification frameworks, CNN, SVM and Naive Bayes have been explored. Word embeddings are obtained by training a word2vec (skipgram) model over the entire collection of code-mixed Facebook messages discussed in Chapter 3 (Table 3.1). The skipgram model is trained for 100 iterations considering window size as 3 and embedding dimensions as 50. Basic

Table 6.1: Sentence-level annotated dataset used to obtain words in the *Resolved* set and *Unresolved* set

Language	**#Sentences**	**Avg Length**
Assamese (as)	5,198	15 words
Bengali (bn)	1,594	12 words
Hindi (hn)	663	12 words
English (en)	21,531	24 words

Table 6.2: Word-level annotated dataset used to train and evaluate word-level language identification techniques

Training Set		**Evaluation Set**	
Language	**#words**	**Language**	**#words**
Assamese (as)	4,273	**Assamese**	7,344
Bengali (bn)	1,061	**Bengali (bn)**	2,412
Hindi (hn)	988	**Hindi (hn)**	1,674
English (en)	23,015	**English (en)**	22,147

pre-processing such as removal of special symbols and numbers and converting all text to lowercase are performed before feeding the text to the skipgram model as well as the classification models.

The CNN classifier is made up of a single convolution layer followed by a max-pooling layer and a fully connected output layer. 50 filters of size 1 are used in the global semantic similarity based classifier and 50 filters each of size 2,3 and 4 are used in the local contextual similarity based classifier. The input to the CNN is represented by the pre-trained word embeddings. The pre-trained word embeddings are also used as features in the Naive Bayes and the SVM classifier. In the global semantic similarity based set-up, only the embedding vector of the target word is used as input while in the local contextual similarity based classifier, the embedding vector of the target word and that of the words within a fixed window size of 3 concatenated in order of their occurrence are used as features. All hyperparameters including the window sizes for the classification set-ups are tuned based on performance over the validation set that is generated by keeping aside 20% of the training data.

Table 6.3: Resolved and unresolved words obtained using sentence-level annotated data

Language		**Total words**
Resolved	Assamese (as)	26,021
	Bengali (bn)	5,662
	Hindi (hn)	1,640
	English (en)	94,421
Unresolved		508,858

6.5 Results and Observations

With respect to the proposed framework, this study makes the following investigations :

- Confidence of the word-level annotations obtained using the sentence-level annotations
- Analyzing the characteristics of the pre-trained word embeddings
- Performance of word-level language identification using global semantic similarity
- Performance of word-level language identification using local contextual similarity
- Comparison of the performance of the proposed word-level language identification using sentence-level annotations to that of word-level language identification using manually annotated word-level data
- Language wise performance analysis
- Performance of the proposed framework in addressing out-of-vocabulary and shared vocabulary

6.5.1 Confidence of word-level annotations obtained using the sentence-level annotation

Section 6.3.1 describes the division of the words from the annotated sentences into resolved and unresolved set. Table 6.3 shows the number of words in different languages in the resolved set $\mathcal{R}$ and the unresolved set $\mathcal{U}$. It is important to evaluate the language classifications obtained at this stage for the resolved set because these words serve as the training samples for the subsequent stages. On comparison with the manually labeled data, the agreement on the language labels is 95.34%. Some of the cases where the annotations have gone wrong are : (i) words in their noisy form have been borrowed into a sentence of another language but have not been used in their native language sentences e.g. words like handsme [handsome], vrfy [verify] occurring in sentences of languages other than English, (ii) words that have been misspelled when embedded in sentences of other language e.g. verivication [verification], reluxation [relaxation], (iii)low frequency words e.g. words like carnivorous, application which are legitimate English words but not occurring in English sentences but embedded in sentences of other languages.

From these observations, it is expected that a larger dataset with a minimum support for all words will yield better results.

6.5.2 Analyzing the characteristics of the pre-trained word embeddings

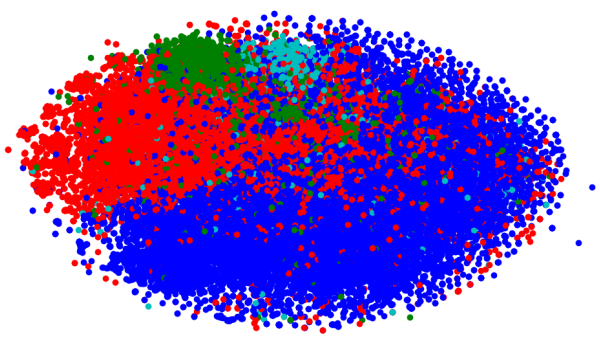


Figure 6.3: 2D projection of word2vec word embeddings

Before diving further into the results obtained using the different proposed strategies, we first investigate the characteristics of the pre-trained word embeddings

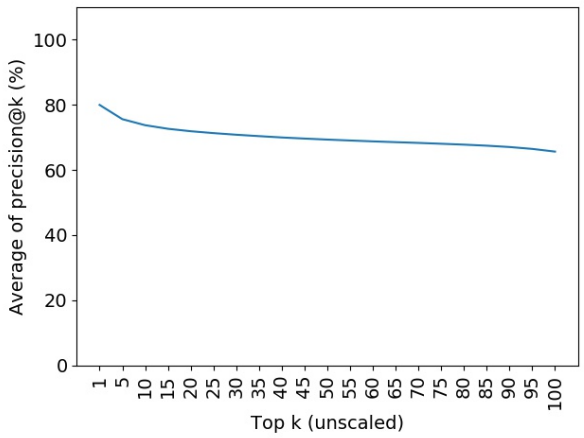


Figure 6.4: Average of precision@k for words in $\mathcal{R}$

that are used as implicit or explicit features in different classification set-ups. The word-level language classification strategies discussed in this chapter are founded on the strong assumption that the word embeddings obtained reflect the language information of the words. To validate this assumption, a 2D projection of the word embeddings (obtained using t-SNE[1]) is shown in Figure 6.3. It should be noted that the plot has been obtained only for words in the resolved set $\mathcal{R}$ since the languages for words in the unresolved set $\mathcal{U}$ are not known. Figure 6.3 clearly shows that words in the same language are well clustered. Further, the plot in Figure 6.4 shows the average of the precision@k considering all words in $\mathcal{R}$. Given a query word from $\mathcal{R}$, the precision at k is obtained by calculating the ratio of the number of words that belong to the same language as the query word to the total number of words retrieved (k). The average of the precision at k is obtained by taking the average over all words in $\mathcal{R}$. It is observed that an average precision of 80% is obtained at k=1, an average precision greater than 70% is achieved for $k \leq 45$ and an average precision greater than 65% is achieved for $k \leq 100$. Therefore, combining the observations from Figure 6.3 and Figure 6.4, it can be safely assumed that the pre-trained word embeddings capture the language information of the words.

Table 6.4: Word-Level Language Identification Performance using Global Semantic Similarity and Local Contextual Similarity

Method	Annotation	Classifier	as	bn	hn	en	MacroF
Global Semantic Similarity	Sentence-level annotation	SVM	79.8	49.0	95.3	60.8	72.6
		CNN	78.3	49.5	94.6	58.7	**72.8**
		NB	78.1	49.5	94.8	62.9	72.3
Local Contextual Similarity	Sentence-level annotation	SVM	84.7	77.5	92.5	81.0	84.6
		CNN	79.5	49.7	95.8	63.6	**85.1**
		NB	82.9	72.7	91.0	81.3	82.6
Global Semantic Similarity	Word-level annotation	SVM	79.7	48.7	96.3	63.9	**74.4**
		CNN	84.5	78.5	91.9	81.4	73.9
		NB	71.9	50.6	90.6	54.0	67.3
Local Contextual Similarity	Word-level annotation	SVM	91.5	86.3	97.2	85.4	82.5
		CNN	92.8	88.0	97.79	87.5	**91.5**
		NB	83.4	73.7	90.9	79.3	82.5

6.5.3 Performance of word-level language identification using global semantic similarity

Word-level language identification using global semantic similarity leads to independent word classification i.e., language classification of a word independent of the context. Thus, in this set-up, a word can belong to at-most one language. Using vector representations of words obtained using methods discussed in Section 6.3 as features, classifiers are trained using the words in $\mathcal{R}$. The words in $\mathcal{U}$ are then fed to the classifiers to obtain a language label for that word. Then given any sentence, the words in the sentence are labeled according to the label obtained using the classifier. However, the evaluation have been done considering the context in which the word has occurred. Results have been shown in table 6.4. It is seen that across all the classifiers, Convolutional Neural Networks show the best performance. A maximum macro-average F-score of 72.8% is achieved. The not-so remarkable performance in this set-up can be attributed to the dependence of a word on the context. For example, the word *bad* has been classified as English. While it is correct in the English context e.g. *Bad day*, it is incorrect in the Assamese context e.g. *Bad dia [Leave it]* and in the Bengali context e.g. *bad dau [Leave it]*.

We further analyze the classification performance of the global similarity based

[1]https://lvdmaaten.github.io/tsne/

method with regards to the embedded words i.e., words from a particular language that have been embedded or borrowed in sentences of other languages and the legitimate words i.e., words that are valid in multiple languages. It is observed that the percentage of embedded works correctly classified is 96.17% and the percentage of legitimate words correctly classified is 76.59%. Thus it is evident that the independent word classification is capable of correctly identifying embedded words as compared to legitimate words. Thus if a corpus contains more number of embedded words than legitimate words, this method can be expected to perform better.

6.5.4 Word-level language identification using local similarity

This set-up takes into account the fact that a word can belong to different languages depending on the context. Using the word vectors of the current word and the words in the context as features, classifiers are trained using the words in $\mathcal{R}$. The words within a window size of 3 on either side of the target word has been considered as features. It is seen that in this set-up as well, Convolutional Neural Networks show the best performance. A maximum macro-average F-score of 85.1% using CNN classifier . This signifies an improvement of 16.91% over independent word classification. Further, analyzing the performance with regard to the embedded words, i.e., words that belong to one particular language and have been borrowed or embedded in sentences of other languages and legitimate words i.e., words that are valid in more than one language, we observe that percentage of legitimate words correctly classified is 91.22% and the percentage of embedded words correctly classified is 86.98%. While there is an improvement in the overall performance as well as in the performance of legitimate words correctly classified, there is a drop in the performance of classification of the embedded words as compared to that using global semantic similarity. However, an average F-score of 85.08% without manually annotated word-level corpus is an encouraging performance.

6.5.5 Sentence-level vs word-level

The classifiers described in the previous two setups have been trained using labels obtained using sentence-level supervision. We repeat the same set-ups using a word-level manually annotated dataset shown in Table 6.2. We see that though CNN and

SVM show comparable performance, SVM slightly outperform CNN in global semantic similarity while CNN shows the best performance in the local contextual similarity. A maximum macro-average F-score of 74.4% and 91.5% is obtained using the global semantic similarity and local contextual similarity respectively. It is observed that using global semantic similarity, the proposed framework using sentence-level information obtain comparable performance performance to that using manually annotated word-level corpus. Using local contextual similarity , the performance using sentence-level information and word-level annotated data are respectively 85.1% and 91.5%. Clearly, training using word-level annotated is giving better performance but considering that the proposed method is less expensive in terms of the required resources, we regard the performances as encouraging and plan to explore more in this direction in future.

6.5.6 Analysis by language

It is seen in Table 6.4 that in all cases, English (en) shows the best performance. If we look at the training set statistics, we see that the number of training samples for English far exceeds the other languages. This can be the reason of comparatively lesser performance for the other languages. Therefore, future efforts will be directed towards building a balanced training set.

6.5.7 Performance of the proposed framework in addressing out-of vocabulary and shared vocabulary

The resolved set $\mathcal{R}$ and the unresolved set $\mathcal{U}$ are disjoint sets. Hence, none of the words in the test set are seen in the training set. Hence the macro-average F-score of 85.1% show that the proposed method can effectively handle unseen or out-of-vocabulary words. Further, all the words in the unresolved set $\mathcal{U}$ are words that occur in sentences of multiple languages i.e., they are shared across multiple languages. Therefore the performance obtained also show that the proposed framework is capable of addressing the shared vocabulary problem.

6.6 Summary

This chapter proposes different word-level language identification techniques (using global semantic similarity and local contextual similarity) for low resource languages in a multilingual environment. The proposed methods make use of the sentence-level structural properties and sentence-level annotations to obtain word-level annotations. Experimental observations show that performance comparable to that obtained using word-level annotated data is obtained. The minimal resource requirements make the framework suitable for low resource languages. However, it is seen that while global semantic similarity is capable of identifying borrowed or embedded words, local contextual similarity is capable of resolving languages of words that are valid in multiple languages. Therefore, one important direction for future work is to explore methods that can combine the advantages of both the global semantic similarity and local contextual similarity. One such approach is discussed in Chapter 7.

7

Switchnet : Learning to Switch for Word-Level Language Identification

Chapter 6 discusses the importance of word-level language identification over code-mixed social media text. Chapter 6 also discusses the limitations of the existing approaches in word-level language identification and proposes different word-level language identification techniques. It shows two important observations. First, the local context is an important indicator of the language of a word when a word is valid in multiple languages. Second, considering the word in isolation from its context leads to more effective language classification when a word is borrowed or embedded into sentences of other languages. In practice however, an integrated framework is desired that can accurately identity both words that are valid in multiple languages as well as words that are embedded into sentences of other languages. In this chapter, we propose a framework for language identification that attempts to achieve the above objective by making use of a dynamic switching mechanism. For a given input, the proposed switching mechanism makes a dynamic decision to bias its prediction either towards the prediction obtained by the contextual information or that obtained by the word in isolation. The proposed approach uses minimal annotated resources and no external resources making it easily extendible to newer languages. From various experimental observations, it is evident tat the proposed techniques outperform the baseline approaches that include the techniques proposed

in Chapter 6 as well as a neural ensemble framework.

7.1 Introduction

Word-level language identification of code-mixed social media text is crucial for any downstream text processing applications like sentiment analysis, machine translation etc. Although word-level language identification have already been addressed in literature, existing studies in word-level language identification exhibit certain limitations.

First, most existing language identification studies necessitate the use of different resources like dictionaries (Das and Gamback [2014]), annotated data (Nguyen and Doğruöz [2013]), monolingual corpora (King and Abney [2013]), transliteration resources (Singh et al. [2018b]) etc. Acquiring these resources is in itself an expensive and tedious task and stands as a major drawback in the applicability of the existing methods to newer languages. We address this limitation in chapter 6 by using sentence level annotated dataset to obtain word-level language information.

Second, existing studies in word-level language identification do not adequately handle naturally embedded borrowed words and shared vocabulary. For example, consider the comment on a celebrity page *What a dum-daar performance ! [What an impactful performance!]*. Here, the word *dum-daar* is a Hindi word that is embedded into an English sentence. On the other hand, consider the examples *stomach ache* and *koyek din ache [Few days left]*. Here the language of the word *ache* varies with the context. In the first case (*stomach ache*), it is an English word whereas in the later case, it is a Bengali word in a phonetically typed Bengali sentence (*koyek din ache [Few days left]*). In a multilingual environment, contextual information plays an important role in disambiguating languages of words, particularly when words are valid across multiple languages. The language of the word *ache* in each of the examples above can be determined based on the contextual information. However, when words are borrowed from a different language (for example the word *dumdaar* in *What a dumdaar performance?*), i.e., when the language of the target word and that of the words in the context are different, taking into consideration the contextual information can be misleading. Therefore, an important challenge

in word-level language identification is to learn when to prioritize and deprioritize contextual information. Existing studies (Nguyen and Doğruöz [2013]) use a combination of word and context features to identify languages of words in a multilingual environment. However, considering variations in word forms and context in social media text owing to factors like code-mixing, phonetic typing and spelling variations, in absence of large-scale annotated data, it may not be possible to capture word and language variations in a multilingual environment. Such a requirement is in contradiction to the objective of the study which is to minimize the requirement of manually annotated resources or any other external resources.

In Chapter 6 , language identification using sentence level annotated data has been proposed. Obtaining sentence level annotations is far less expensive in terms of manual effort than obtaining word-level annotations. Further, in Chapter 6, language identification considering the words in isolation (*global semantic similarity*) and considering the contextual information (*local contextual similarity*) have also been explored. It shows two important observations. Considering the words in isolation helps in correctly identifying languages of words that are embedded or borrowed into sentences of other languages. Considering contextual information on the other hand is helpful in identifying languages of words when words are valid in multiple languages. However, in practice, what is desired is a single unified classification framework that can effectively identify both words that are embedded or borrowed into sentences of other languages as well as words that are valid in multiple languages. It is to be noted that the requirements (word in isolation vs word with contextual information) are complementary in nature that stand as a major obstruction in achieving a single model that addresses both challenges. In this chapter, we therefore further investigate the output of the two classifiers reported in Chapter 6. We observe that selecting the correctly classified examples from each of the two classifiers can lead to a significant boost in the performance. Motivated by this observation, we propose to build a dynamic switching mechanism between the classifier built considering the contextual characteristics of words and the classifier built considering the words in isolation. By designing a dynamic switching mechanism, the proposed method attempts to choose the output of one of the two classifiers based on the input characteristics. An alternate way to integrate the information

from multiple classifiers is to build an ensemble classifier. The performance of the proposed framework is also compared to that of a neural ensemble framework details of which are reported in Section 7.5.

Frameworks similar to the one proposed in this chapter have been also been explored in other tasks ; Rei et al. [2016] use an attention layer to shift the attention between character and word embeddings to handle out-of-vocabulary and in-frequent words in neural sequence labelling models. A similar model is also used by Miyamoto and Cho [2016] to find an optimal combination of character level and word-level information for their language modeling task. The difference between these models and the proposed model is that while these models use the attention mechanism to choose between different input representations, the proposed model uses a similar mechanism to choose between classifier outputs for the same input. The advantage of the proposed model is that it can be used with any baseline classifiers. In summary, this chapter has the following contributions:

- Proposes a word-level language identification framework for code-mixed social media content that can dynamically switch decisions between different classification outputs based on a given input. Experimental results show that the proposed framework is able to achieve better performance compared to the baseline components as well as the neural ensemble framework.
- Proposes the use of non text based features for language identification. Experimental results show that the non-textual features yield performance that is comparable to text based features.

7.2 Related Studies

Word-level language identification over code-mixed social media text have already been explored in literature. Existing studies in this direction have also been discussed in Chapter 6. We have seen that existing studies in word-level language identification can be broadly categorized into two categories - supervised approaches and unsupervised or weakly supervised approaches. Supervised approaches make use of word-level annotated data along with external resources like dictionaries and transliteration tools. Different classification algorithms and different feature sets

have been explored. For example, Nguyen and Doğruöz [2013] explore methods both for classifying words independent of the context as well as based on context. Dictionaries and character n-gram language models are used for classifying words independent of the context. On the other hand, models such as logistic regression and CRF with neighbouring words as features have been explored for classifying words according to the context. While language models yield improved performance compared to dictionaries, particularly in identifying languages of noisy word forms, using contextual information yield better performance compared to language model based classification. Chittaranjan et al. [2014] use a range of features such as character n-gram features, word features and contextual features with a CRF based classification model. Similarly, a CRF model with spelling characteristics of the target word and words in the context as features is used by Xia [2016]. On similar lines, character, word and context features have been used by Jhamtani et al. [2017] and Barman et al. [2014]. Neural network based models that capture character, word and context features have also been explored in literature. Jaech et al. [2016] use two CNN layers to obtain word embeddings that capture the character information and a BiLSTM layer to capture the contextual information. Similarly, Mandal and Singh [2018] use a CNN-LSTM layer for capturing character information followed by a Bi-LSTM-CRF layer for capturing contextual information. Similarly, Samih et al. [2016] uses LSTM-CRF for word language identification. In a slightly different set-up, a feed forward model is used to classify each word independent of the context which is then followed by a decoder that predicts the labels for a word sequence taking the entire sentence into consideration. In contrast to the above approaches that uses word-level annotated data as training data, unsupervised and weakly supervised techniques using monolingual corpora in individual languages are used by King and Abney [2013] and Rijhwani et al. [2017]. As obvious, the objective of the unsupervised or weakly supervised approaches is to reduce the manual audit requirement.

Despite significant performances obtained using the existing approaches, they do not address the two limitations discussed in the previous section. While the supervised approaches make use of word-level annotated data that requires manual audit, the unsupervised/weakly supervised techniques are not applicable over transliter-

ated text due to unavailability of clean monolingual corpora in individual languages. Additionally, the above approaches also exhibit limitations in terms of their ability to prioritize and deprioritize contextual information in order to efficiently obtain language labels for words that are naturally embedded into sentences of other languages as well as words that are valid across multiple languages. As clearly evident from the discussion above and also, as pointed out in the introduction, the existing approaches use a mix of word and context features for differentiating between the two categories of words discussed above. However, considering variations in word forms and context in social media text, in absence of large-scale annotated data, it may not be possible to capture word and language variations in a highly multilingual environment. Therefore, in this chapter, we propose a dynamic switching based language classification framework that leverages the individual classifiers built in Chapter 6 using sentence level annotation to obtain accurate language labels using minimal annotated resources.

7.3 Proposed Methodology

This section describes the proposed model to address word-level language identification over code-mixed social media text in a multilingual environment. The objective is to build a dynamic switching mechanism between two classifiers which consider word in isolation (described as *global semantic similarity*) and word with its contextual information (described as *local contextual similarity*) proposed in Chapter 6 . Like in Chapter 6, to build the classifier considering word in isolation and word with contextual information, this study considers the same sentence level annotated dataset. While using sentence level annotations, the vocabulary set is divided into two disjoint subsets *resolved set* $\mathcal{R}$ and *unresolved set* $\mathcal{U}$. The *resolved set* $\mathcal{R}$ are those words that occur only in sentences of a particular language. Therefore, they can be assumed to belong to the same language as the language of the sentences in which they are occurring. The *unresolved set* on the other hand are those words that occur in sentences of multiple languages. Therefore, the language of these words cannot be confirmed using the sentence level language information alone. So, the task of language identification can be considered as the task of identifying the

underlying languages of the words in the *unresolved set* $\mathcal{U}$ using the words in the *resolved set*. The words in the *resolved set* $\mathcal{R}$ are used as training samples in building classifiers. The classifiers proposed in Chapter 6, i.e., considering word in isolation and considering contextual information of words are considered as the baseline classifiers in this study. Details of these classifiers have been discussed in Chapter 6 . They are briefly revisited here to help the reader understand the proposed methods with ease.

7.3.1 Word-level language classification considering word in isolation (global semantic similarity)

This approach assumes that taking into consideration information about the target word alone is sufficient to identify the language of a word. Since this approach is based on the target word only irrespective of the context in which the target word appears, therefore all occurrences of a given word form are identified as belonging to the same language. In this study, the target word is represented by a word embedding vector that is obtained by training word embedding models over a large corpus. Word embedding models like word2vec (Mikolov et al. [2013]) capture the syntactic and semantic similarities of words in a corpus by making use of the contextual information. The vector representations of words generated by these models are such that similar words lie close together in the projected vector space. Therefore, when trained over a large corpus of code-mixed text, the intuition is that the vector representations obtained are able to capture the language characteristics such that words in the same language lie closer in the projected vector space than words in different languages. In such cases, the language of a target word can be identified by finding it's similarity to other words whose languages have already been identified. Therefore, with words in the *resolved set* $\mathcal{R}$ as training samples and their vector representations as feature vectors, a classifier is trained. The trained classifier is then used to identify words in the *unresolved set* $\mathcal{U}$.

7.3.2 Word-level language classification using contextual information (local contextual similarity)

In a multilingual environment, context (preceding words and following words) often helps in identifying language of a target word i.e., language of the context and target words may often be same. Contextual information is particularly important when the languages under consideration contain shared vocabulary whether due to inherent similarities between languages or due to other phenomena such as phonetic typing or creative writing. Therefore, in the local context similarity based classifier, context information is also considered while classifying the language of the target words. The target word is represented by a vector which is the concatenation of the vector representations of the target word and words in the context in the order of their occurrences. Like in global similarity, vector representations are obtained by training word emebedding models over a large corpus of code-mixed text.

Both the above classification approaches are associated with the respective advantages and disadvantages. The global semantic similarity based approach being based only on the embedding vector of the target word, all occurrences of the same word are assigned the same language label. While, this approach works well when words are borrowed or embedded into sentences of other languages, this approach fails when the word is valid in multiple languages. On the other hand, considering the contextual information helps in correctly identifying languages of words that are valid in multiple languages. In a multilingual noisy environment, it is desirable that the model learns to prioritize between the words in isolation and the contextual information depending upon a given input. The proposed model discussed below attempts to achieve this by using a feed-forward neural network combined with the output from the individual classifiers which we refer to as the baseline classifiers with respect to the proposed framework.

7.3.3 Proposed Classification Framework

The proposed approach uses a feed-forward network to dynamically switch decision between the *global semantic similarity* and *local contextual similarity* for any given input. Figure 7.1 shows the proposed model architecture. Let $\mathcal{C}1$ represents the

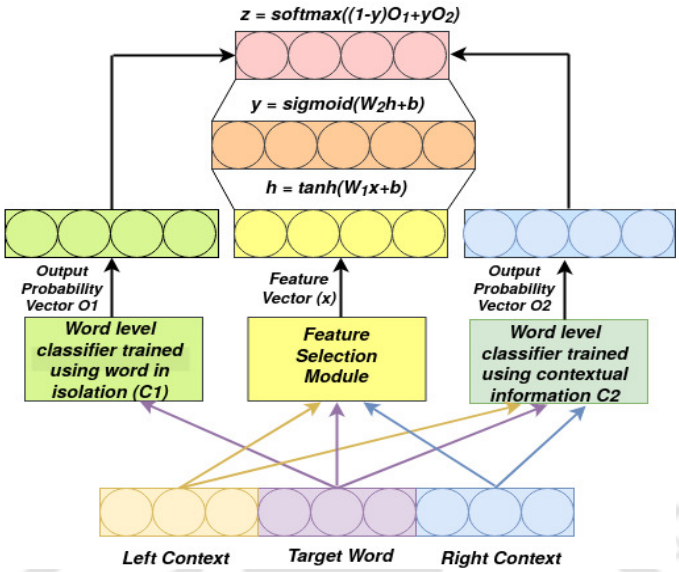


Figure 7.1: Proposed word-level classifier combining output word-level classifiers built using word in isolation (C1) and using contextual information (C2)

global semantic similarity based classifier and $\mathcal{C}2$ represents the *local contextual similarity* based classifier. The output from $\mathcal{C}1$ and $\mathcal{C}2$ are vectors representing the probability distribution over the languages for the target word. Let these probability distributions be represented by $\mathbf{o}_1$ and $\mathbf{o}_2$ respectively. Let $\mathbf{x}$ be the feature vector representing the target word. This $\mathbf{x}$ could be the textual content of the input. It can also be other non-textual based features representing the target word and its context. The different representations of $\mathbf{x}$ are discussed in detail in Section 7.3.4. The input vector $\mathbf{x}$ is passed to a fully connected feed forward neural network with one hidden layer. Let W_1 be the parameter matrix between input layer and the hidden layer and W_2 be the the parameter matrix between the hidden layer and the switch layer. The output vector of the switch layer $\mathbf{y}$ is normalized with a sigmoid operation. The expressions for the forward pass are shown below.

$$\mathbf{h} = tanh(W_1^T \mathbf{x} + \mathbf{b}_1) \tag{7.1}$$

$$\mathbf{y} = \sigma(W_2^T \mathbf{h} + \mathbf{b}_2) \tag{7.2}$$

The vector $\mathbf{y}$ in equation 7.2 acts as the switch the purpose of which is to switch the output of the classifier between $\mathbf{o}_1$ and $\mathbf{o}_2$ based on the given input $\mathbf{x}$ as shown in equation 7.3 where $\odot$ denotes element wise multiplication.

$$\mathbf{z} = softmax((1 - \mathbf{y}) \odot \mathbf{o}_1 + \mathbf{y} \odot o_2) \tag{7.3}$$

If $|\mathbf{y}|$ is high i.e., the values in $\mathbf{y}$ are close to 1, the first component of the sum in equation 7.3 reduces to 0 and the final output of the classifier is biased towards $\mathbf{o}_2$. Similarly, if $|\mathbf{y}|$ is low i.e., the values in $\mathbf{y}$ are close to 0, the final output of the classifier is biased towards $\mathbf{o}_1$. Therefore, the objective of the proposed framework is to train the network in such a manner that the values in $\mathbf{y}$ range in between 0 and 1 according to whether the final desired output should be biased towards $\mathbf{o}_1$ or $\mathbf{o}_2$. The proposed model comprises of three components - the two baseline components and the feed-forward network that acts as the switching mechanism between the baseline components. Instead of training the three components end-to-end, each of the components are trained separately and the output from the pre-trained baseline models are selected based on the output of the proposed switch network i.e., $\mathbf{y}$. This enables the proposed framework to be used with any baseline component. For learning the parameters of switch network, the representation vector of the input text $\mathbf{x}$ is considered as the only input to the network. The output vectors $\mathbf{o}_1$ and $\mathbf{o}_2$ of the component classifiers are used while estimating the model output $\mathbf{z}$ as defined in equation 7.3. Let $\tilde{\mathbf{z}}$ is the ground truth vector (one hot vector) associated with the input $\mathbf{x}$. Cross entropy between $\tilde{\mathbf{z}}$ and $\mathbf{z}$ has been used as the loss function for learning the model parameters W_1 and W_2 as defined below.

$$E = -\sum_{i=1}^{C} \tilde{z}_i log(z_i) \tag{7.4}$$

where C is the number of classes i.e., the number of languages. The training procedure for the component classifiers are discussed in Section 7.4.

7.3.4 Input representation

The goal of the proposed framework is to dynamically switch between the baseline classification outcomes depending on the given input. This study proposes different features that can be used to represent a given input text to train the proposed framework to make an effective switching decision. These features are discussed below.

Content features

The objective in the content based features is to use the textual information to identify the language of a word. The textual information considered in this case comprises of the target word and the surrounding words. Therefore, the input vector **x** while considering content features is same as that of the context based component classifier, i.e., the target word along with its preceding and following context is fed as input to the proposed classifier. The intuition is that the word along with it's context will be able to capture switching characteristics between the two classifiers. Pre-trained word embeddings obtained by training a word embedding (skipgram) model over a large code-mixed corpus are used to represent the input. While feeding as input to the classifier, the word embeddings of the target word and the words in the context are concatenated in the order of their occurrence.

Neighbourhood based features

The content based features make use of the textual information directly for identifying language of a target word. However, the noise present in the social media data like irregular and ambiguous word forms leads to limitations in the performance of the text based features. Therefore, the goal in the neighbourhood based features is to extract non textual information from the data that can overcome the limitations of the text based features.

The neighbourhood based features proposed in this study attempts to encode the semantic relation of the words with other words in the corpus into a fixed sized vector. Therefore, the neighbourhood vector is a fixed sized vector representing structured information about the input in contrast to the content features that are

represented by the word embeddings that accomodate detail semantic information about the word and its context.

This study proposes two neighbourhood based features as discussed below. The intuition is that the neighbourhood characteristics of a word can be an important indicator of whether the word is a borrowed word or is a word that is valid in multiple languages. These approaches are discussed in detail below:

- **Local Neighbourhood** : The Local Neighbourhood based information tries to encode the semantic information between the target word and its local context (preceding and following words). The intuition is that the semantic distance of the target word with its context can be a good indicator of whether the target word and the words in the context belong to the same language. In other words, if a target word belongs to the same language as its context, its distance from other words in the context will be lesser compared to when the target word belongs to a different language and is embedded or borrowed into the current context. Accordingly, the classification output can be biased towards the prediction obtained considering the contextual information or prediction obtained considering the word in isolation. This approach is therefore called the local neighbourhood based approach because it considers the local neighbourhood of the target word.

  The local neighbourhood is represented by a vector comprising of the minimum, maximum and average distance of the target word from the words in the context. The distance is given by the cosine similarity between the embedding vectors of the words. The intuition is that a word will be semantically more similar to words in the same language than words in other languages. Therefore, when a word is borrowed or embedded into a sentence of another language, its distance from the context would be higher compared to instances when the target word and the context words are in the same languages.

  Let $\mathbf{t}_i$ be the embedding vector of the i^{th} word in a sentence. Then the t_{i-j} and t_{i+j} denote its j^{th} left and right neighbours, respectively. The input vector

$\mathbf{x}$ to the switch network is defined by a vector with three elements as follows :

$$x_0 = \min(\{s|s = cosine(\mathbf{t}_i, \mathbf{t}_{i+j}))\ \forall_j, -k \leq j \leq k, j \neq i\}) \tag{7.5}$$

$$x_1 = \max(\{s|s = cosine(\mathbf{t}_i, \mathbf{t}_{i+j}))\ \forall_j, -k \leq j \leq k, j \neq i\}) \tag{7.6}$$

$$x_2 = \frac{1}{2k} \sum_{\substack{j=-k \\ j \neq i}}^{k} (cosine(\mathbf{t}_i, \mathbf{t}_{i+j})) \tag{7.7}$$

where k denotes the number of words in the left and right neighbourhood.

- **Global neighbourhood** : The global neighbourhood tries to encode the semantic information of the target word with respect to its neighbours in the global embedding space. The global neighbourhood of the target word is represented by a vector that represents the k nearest neighbours of the target word in the global embedding space. The intuition is that if a word is valid across multiple languages, the neighbourhood of this word in the global embedding space will consist of words of different languages. On the other hand, if a word is valid only in a single language and its occurrences in sentences of other languages are result of embedding/borrowing, the global embedding space of such words will comprise all words of the same language. Therefore, while determining the language of a target word, if the neighbourhood of the target word is dominated by words in a single language, the output from the classifier considering the word in isolation should be given preference. On the other hand, if the neighbourhood is comprised of words from different languages, then while determining the language of the word, the output from the classifier considering contextual information should be given priority. Global neighbourhood vector $\mathbf{x}$ of a word t is the distribution of the languages in it's neighbourhood. Let $\mathcal{V}_k$ be the set of top k most similar words to the target word t and let $\mathcal{L}$ denote the set of languages under consideration. Then the elements of the feature vector $\mathbf{x}$ that denotes the global neighbourhood of the target word t consists of the number of words in each language in the set $\mathcal{V}_k$.

For example if $|\mathcal{L}|$=4, then the feature vector representing the global neighbourhood of the term t is given by $\mathbf{x}$ =(x_1,x_2,x_3,x_4) where $x_i, i = 1...4$, is the number of words in $\mathcal{V}_k$ that belong to language L_i.The feature vector $\mathbf{x}$ is then used as input to the switch network. As discussed in Section 7.4, the switch network is built considering only words in the resolved set $\mathcal{R}$.

Combination of content and neighbourhood based features

The aim is to represent an input by a combination of its content features and neighbourhood features. The idea is to explore the advantages of combining the detailed semantic information represented by the content features with the fixed sized structural information represented by the local and global neighbourhood. We explore two ways of combining the features. In the first approach, all three sets of features, i.e., the text features, the local neighbourhood features and the global neighbourhood features are simply merged in the hidden layer. In the second approach, the above sets of features are merged in the hidden layer following a selective feature dropout mechanism proposed in (Zhang et al. [2018]). In the original paper (Zhang et al. [2018]), selective feature dropout is implemented by randomly setting the feature group values to zero with 50% probability. However, in the current study, the feature dropout strategy is implemented by alternatively setting a very high dropout (90%) between each feature group and the hidden layer during the training iterations.

7.4 Datasets and Experimental Set-up

This section discusses in detail the datasets and the experimental set-ups used to carry out the experimental investigations pertaining to the language identification techniques discussed in this study.

7.4.1 Experimental Dataset

The proposed language identification framework works in three phases. First, using the sentence level dataset, as discussed in Section 7.3.3, *Resolved* word set and *Unresolved* word set are created. This study also uses the SLI-MI dataset.

Table 7.1: Experimental dataset used to train and evaluate the proposed switch network

Language	Training (#Words)	Evaluation (#Words)
Assamese (as)	239	10,132
Bengali (bn)	167	4,720
Hindi (hn)	105	6,464
English (en)	780	12,859

To train the proposed switch network, a small number of words from the resolved set $\mathcal{R}$ are chosen for manual annotation. The goal for manual annotation is to correct any wrong labels in $\mathcal{R}$ that may have occurred while propagating the language information from sentence level to word-level. Furthermore, if both the baseline components result in the same predicted label, then the decision of the switch network is not crucial. The prediction from any one of the baseline components may be chosen as the final prediction. However, if the predictions from both the baseline components are different, then the switch network must learn to make the correct choice between the baseline components based on the input characteristics. Therefore, while choosing training samples for the switch network, samples that obtain different predictions from the baseline components are chosen. Further, to evaluate the performance of the proposed framework, we generate a manually annotated evaluation set by randomly sampling words from the unresolved set. The training and evaluation set is shown in Table 7.1.

7.4.2 Experimental Set-Up

The experimental set-up for the baseline classifiers is same as that described in Chapter 6. The words in the resolved set $\mathcal{R}$ are used as training samples. Four popularly used classification frameworks - Convolutional Neural Network (CNN), Bidirectional Long Short Term Memory (BiLSTM), Logistic Regression (LR) and Support Vector Machines (SVM) are used. The CNN classifier is made up of a single convolution layer followed by a max pooling layer and a fully connected output layer. 50 filters each of size 2, 3 and 4 are used for training the classifiers using contextual information and 50 filters of size 1 are used for training the classifiers

using words in isolation. The BiLSTM classifier also consists of a single BiLSTM layer with 100 units followed by a fully connected output layer. Both the CNN and BiLSTM classifiers are trained using Adam optimizer and training is monitored through validation loss. 20% of the training data is set aside as validation data. In case of the LR and SVM classifiers, the pre-trained embeddings are used as features. The pre-trained word embeddings are the same as those described in Section 6.2. For training the classifier using word in isolation, only the embedding of the target word is fed as input while for training the classifier using contextual information, the embedding vectors of the words in the context and the embedding vector of the target word are concatenated in order to be used as features. All hyper-parameters are tuned using the validation data described above. The SVM classifier uses a rbf kernel.

The switch network consists of a hidden layer with 100 units and an output layer with number of units corresponding to the number of languages (i.e., 4). The feature vectors generated using strategies discussed in Section **??** are used as input. The text features are constructed considering window size of 3 on either side of the target word. The local neighbourhood feature is constructed considering a window size of 3 on either side of the target word. The global neighbourhood feature is constructed considering the top 10 neighbours from the resolved set $\mathcal{R}$[1].

7.5 Results and Observations

This section discusses the results and observations obtained using different experimental setups. First, the performance of the base-line classification set-ups is discussed followed by the performance of the proposed framework using different features.

7.5.1 Performance obtained using the baseline classifiers

The aim of the proposed framework is to combine the advantages of the classifiers using *word in isolation* and using *contextual information*. With respect to the

[1]Since there is not enough word-level annotated data to use as validation data, for the content feature and the local neighbourhood feature the value of k has been set to 3 as in the baseline set-up and the value of k for the global neighbourhood has been set to 10 as the performance started converging around k = 10

Table 7.2: Language wise F-scores and average macro-Fscore (MacF) and micro-Fscore (MicF) obtained by the baseline classifiers

Classifier	Baseline (Word in isolation)						Baseline (Word with context)					
	as	bn	en	hn	MacF	MicF	as	bn	en	hn	MacF	MicF
CNN	78.72	55.55	89.41	70.40	75.40	78.73	86.27	87.04	84.58	89.52	**87.48**	**86.61**
BiLSTM	78.33	55.18	89.55	68.67	75.17	78.44	85.63	86.67	84.63	87.48	86.50	85.70
SVM	80.30	57.07	90.90	77.38	**76.56**	**80.05**	86.22	86.54	85.21	89.63	87.25	86.50
LR	79.14	55.54	90.96	74.55	75.42	78.91	85.55	85.29	85.11	86.95	86.23	85.58

proposed framework, we refer to these classifiers as baseline classifiers. In this section, we analyse in details the performance of the baseline classifiers followed by the performance of the proposed framework in the following sections.

The performance of the baseline classifiers have also been discussed in chapter 6. Considering word in isolation, best macro and micro-average F-scores of 76.56% and 80.05% are obtained using SVM classifier. The considerably lower F-scores can be attributed to the nature of the classification set-up where all occurrences of the same word are represented by the same feature vector and hence predicted as belonging to the same language. Therefore, this approach works well when the text consists of large number of embedded or borrowed words but less number of words that are valid across multiple languages. Considering contextual information, best macro and micro average F-scores of 87.48% and 86.61% is obtained using CNN classifier. It is also observed that using classifier built considering words in isolation, the percentage of embedded or borrowed words correctly classified is 96.17% while the percentage of legitimate words correctly classified is 76.59%. On the other hand, using the classifier built using contextual information, the percentage of embedded words correctly classified is 86.98% while the percentage of legitimate words correctly classified is 91.22%. Therefore although considering contextual information helps in improving the overall classification performance, there is a major drop in performance in terms of language identification of the embedded or borrowed words. This observation serves as a major motivation for the current study which attempts at combining the advantages of both the baseline classifiers.

Table 7.3: Max achievable MicroF by combining baselines

Classifier	Max
CNN	**95.49**
BiLSTM	94.25
SVM	94.76
LR	93.87

Table 7.4: Performance obtained by neural ensembling of the baseline classifiers (using word in isolation and using contextual information)

Classifier	as	bn	en	hn	MacF	MicF
CNN	86.39	86.24	84.94	90.28	87.33	86.56
BiLSTM	86.09	85.89	84.55	86.26	86.17	85.61
SVM	86.46	86.74	89.94	90.04	**87.53**	**86.59**
LR	85.20	83.95	85.21	87.27	85.73	85.46

Error analysis of the baseline classifiers

With the motivation stated above, we attempt to estimate the best performance that can be achieved if a correct choice is made between the outputs of the baseline classifiers for each given input. In general, if n_1 and n_2 are the number of correctly classified samples by the two baseline components and n_{12} is the number of test samples that are correctly classified by both the baseline components, then the maximum achievable Micro-average F-score by the proposed framework will be given by $\frac{n_1+n_2-n_{12}}{n}$ where n is the total number of test samples. We empirically make an estimate of the maximum achievable performance by choosing the best predictions from each of the classifiers which is shown in Table 7.3. We compare this to the performance obtained by the proposed framework reported in Table 7.5. Observations show that there are still rooms for improvement so that the proposed framework achieves the maximum achievable performance.

Performance obtained using ensemble classification

Ensemble frameworks are commonly used frameworks to integrate information from different classification frameworks. This study therefore, also uses a neural ensemble

Table 7.5: Language wise and macro (MacF) and micro (MicF) average F-scores obtained using the proposed framework under different classification-setups and the relative performance with respect to the baseline components

Classifier	Features	as	bn	hn	en	MacF	MicF
CNN	Text	91.49	87.54	91.27	91.21	90.20(+3.10%)	90.49(+4.47%)
	Local Neighbourhood Distance	86.96	86.36	86.21	91.52	88.13(+0.74%)	87.44(+0.95%)
	Global Neighbourhood	90.45	88.80	90.85	91.04	90.37(+3.48%)	90.47(+4.51%)
	Combination of all Features	90.89	87.94	91.93	91.55	90.63(+3.60%)	90.98(+5.04%)
	Combination with Feature Dropout	91.09	89.05	91.72	90.97	**90.79**(+3.96%)	**91.03**(+5.16%)
BiLSTM	Text	89.68	86.98	89.43	87.02	88.48(+2.28%)	88.74(+3.54%)
	Local Neighbourhood Distance	86.19	85.93	84.17	86.91	86.36(-0.16%)	85.75(+0.05%)
	Global Neighbourhood	88.41	83.61	90.42	86.73	87.61(+1.28%)	88.24(+2.96%)
	Combination of all Features	90.00	87.12	89.76	87.05	88.69(+2.53%)	89.00(+3.79%)
	Combination with Feature Dropout	90.17	87.14	89.98	86.85	**88.74**(+2.58%)	**89.10**(+3.90%)
SVM	Text	89.16	84.16	89.89	90.22	88.44(+1.36%)	88.93(+2.80%)
	Local Neighbourhood Distance	86.28	84.83	85.05	90.30	86.84(-0.46%)	86.38(-0.13%)
	Global Neighbourhood	88.57	83.88	89.63	90.08	88.20(+1.08%)	88.62(+2.45%)
	Combination of all Features	88.91	83.93	90.12	89.15	88.22(+1.11%)	88.77(+2.62%)
	Combination with Feature Dropout	89.67	84.39	90.25	90.39	**88.76**(+1.73%)	**89.27**(+3.20%)
LR	Text	88.43	83.52	89.68	87.27	87.37(+1.32%)	88.00(+2.82%)
	Local Neighbourhood Distance	85.51	84.23	85.32	87.27	85.85(-0.44%)	85.58(+0.00%)
	Global Neighbourhood	87.50	83.07	89.44	87.25	87.04(+0.93%)	87.60(+2.36%)
	Combinationof all Features	88.78	84.16	89.78	87.91	87.80(+1.82%)	**88.35**(+3.23%)
	Combination with Feature Dropout	89.13	84.38	89.26	87.94	**87.81**(+1.83%)	88.29(+3.16%)

framework to combine the output of the baseline classifiers. Classification outputs (output probability classifiers) from the baseline classifiers are used as feature vectors which are fed to a fully connected hidden layer followed by a fully connected output layer. This performance is reported in Table 7.4. It is observed that the performance of the ensemble framework is very close to that obtained using classifier considering contextual information with very minimal improvement.

7.5.2 Performance of the Proposed Framework

Table 7.5 shows the performance obtained using the proposed framework. Among the different baseline classification set-ups, using CNN as the baseline classifier shows the best performance.On comparison with the baseline classifiers, we find that the proposed framework exhibits improvement compared to the performance of the baseline classifiers. Further, using the proposed framework, it is seen that among the different features used to represent the input, the content based features show the

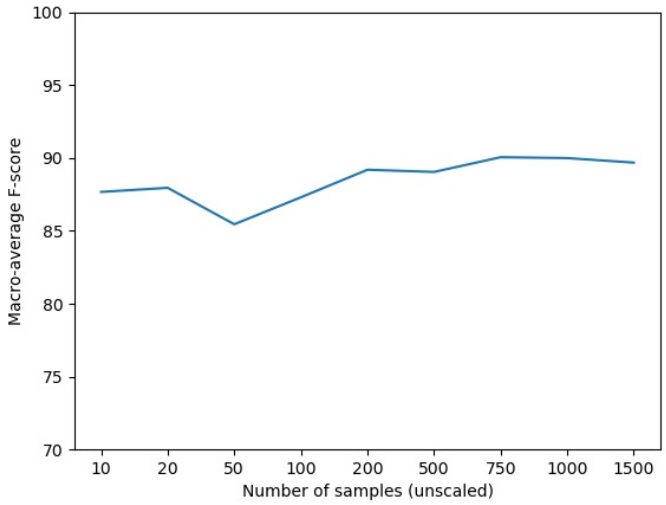


Figure 7.2: Performance obtained using the proposed model (content features) with varying number of training samples

best performance with combination of all features performing marginally better than the text based features alone. In the following discussion, we explore the performance of the proposed model using varying number of training samples and using different features.

Performance obtained with varying number of training samples

Figure 7.2 shows the performance of the proposed framework with varying number of training samples. Interestingly, the switch network is able to converge with a small dataset (87.5% for 10 number of samples to 90.1% for 750 number of samples). As observed in Figure 7.2, no significant changes in performance is obtained on increasing the number of training samples. An analysis shows that the misclassifications are due to noisy word forms and noisy context. Therefore, it is assumed that further improvement in performance can be obtained only with the help of supervised information through annotated resources and knowledge bases.

Performance obtained using different features

Section 7.3.4 discusses the different ways used to represent the input before feeding it to the proposed model. This discussion highlights the major observations.

1. **Text based features show good performance across all setups**. The text features in the current experimental set-up are represented by the corresponding word embeddings. Therefore, the good performance of the text features can be attributed to the pre-trained word embeddings. When trained over a large corpus, the word embeddings are expected to capture the inherent similarities between words in the same language as well as dis-similarities between words in different languages. The word embedding of the target word combined with that of the context serves as a representation of the word in a particular context. The performance obtained using words in $\mathcal{R}$ as training samples shows that the word representations thus obtained are distinctive in nature in terms of language identification.

   However, the word embeddings used have a few disadvantages. First, embedding of the same word in different senses are conflated into a single embedding. Therefore, occurrences of the same word in different languages is combined into a single embedding. Second, due to large number of spelling variations in social media data, word embeddings of infrequent word forms may not be reflective of their actual language characteristics.

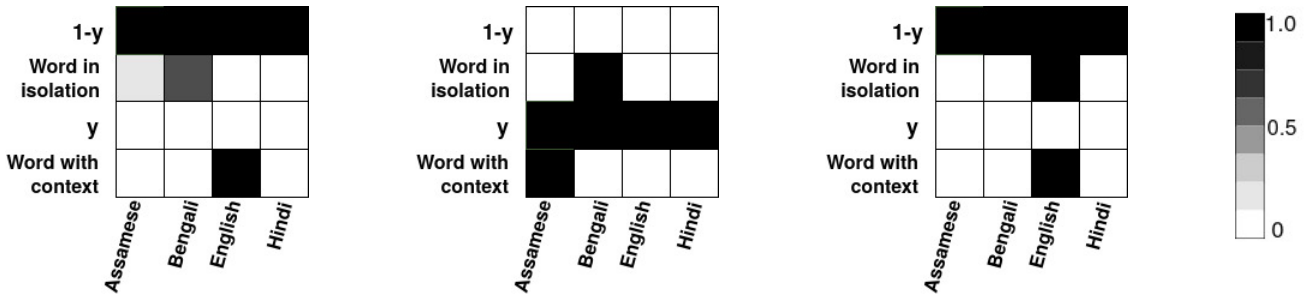
2. **Neighbourhood based features do not excel but exhibit performance that is comparable to the text based features**. The neighbourhood based features (the local neighbourhood and the global neighbourhood) represent finite structural information about the target input and its context in contrary to the text based features that represent detail semantic information. Although the neighbourhood based features do no excel the text based features in all set-ups, the performance is comparable to the text based features. These observations convey two important advantages of the proposed neighbourhood based features. First, the neighbourhood based information can be used complementary to the text based information to counter the ambiguities resulting from text based features. For example, when a word is embedded into a sentence of another language, the global neighbourhood based features can be correct indicator of the language of the word. Second, the finite sized neighbourhood vectors are able to capture the same amount of information as

the comparatively larger word embedding vectors thus leading to a reduction in the amount of computational resources required.

Further, considering that the neighbourhood features are computed based on the embeddings that are themselves generated from noisy user generated text, it can be expected that with cleaner neighbourhood representations, there will be further boost in performance.

3. **Combination of all features** : Although the neighbourhood based features (local neighbourhood and global neighbourhood) individually do not outperform the text based features across all set-ups, on combination with the text based features, they add to improvement compared to the performance obtained using text based features only. However, the improvement is only minimal. This minimal improvement can be explained from the experimental observations where it is observed that the results obtained from the text based features and the neighbourhood based features are highly co-related. This is also true conceptually because the neighbourhood based features are a structured representation of the text based features. The structured information resolve some of the ambiguities that may be inflicted by the text based features. Therefore, some of the mis-classifications resulting from the text based features are correctly classified by the neighbourhood based features. However, in the attempt to extract structured features from the unstructured text based features into a fixed size neighbourhood vector, there is also loss of information. This is why, performance of the neighbourhood based features individually is comparatively lower than the text based features in certain set-ups.

   However, as mentioned above, since the neighbourhood vectors used in this study are themselves obtained from word embedding vectors that are generated from noisy text, it can be expected that with cleaner representations, they should be able to excel the textual features.



(a) Heatmap for target word *request* in the text *eta [bn] amar [bn] request [en] apnar [bn] proti [bn]*

(b)Heatmap for the target word *nischoy* in the text *nischoy [as] kiba [as] eta [as] hoise [as]*

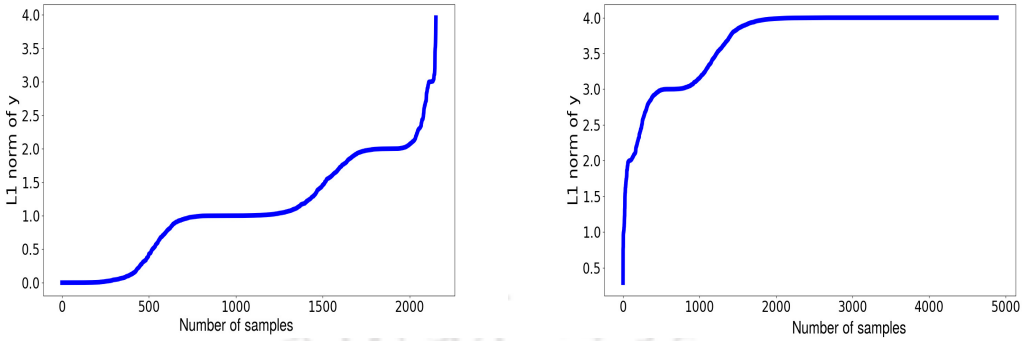
(c) Heatmap for the target word *good* in the text *Have [en] a [en] goood [en] day [en]*

Figure 7.3: Figure showing the value of switch vector **y** for different input text

7.5.3 Analysis of the switch layer

The objective of the proposed framework is to make an optimal choice between the baseline classifiers using a dynamic switching mechanism. The vector **y** discussed in Section 7.3.3 acts as the switch. To validate whether **y** actually acts as a switch, we analyze the values of **y**. Fig 7.3 shows a visualization of the value of **y** for three different examples using a heatmap where the color black corresponds to 1 and white corresponds to 0 and the intermediate values correspond to the intermediate shades.

Fig 7.3(a) is an example where the target word *request* which is an English word is embedded into a transliterated Bengali sentence *eta amar request apnar proti [this is my request to you]*. Therefore the class predicted using word in isolation should be given more priority over that obtain using the context. We see that the weight associated with the output from the classifier using the contextual information i.e., the values in **y** approximate to 0 making the values in **1y** approximately one satisfying our objective. Similarly, Fig 7.3(b) is an example where both the target word *nischoy [sure]* and the context *nischoy kiba eta hoise [definitely something must have happened]* are in the same language and therefore the class label predicted by the classifier using the contextual information should be given more priority which is also validated from the values of **y** and **1** **y** in the Figure. Figure 7.3(c) is an example where the class labels predicted using both the baseline



(a)L1 norm of $\mathbf{y}$ for test examples where prediction obtained using word in isolation is correct

(b) L1 norm of $\mathbf{y}$ for test examples where prediction obtained using contextual information is correct

Figure 7.4: L1 norm of the switch vector $\mathbf{y}$

classifiers are the same. Therefore, the switch vector $\mathbf{y}$ may be biased towards either prediction (word in isolation in this case).

The above illustration is for a few particular examples. In order to visualize the characteristic over a larger set, we analyse the L1 norm of $\mathbf{y}$ for the two possible cases : first, when the output should be guided by the word in isolation or *global semantic similarity* and second, when the output should be guided by the contextual information or *local contextual similarity*. This is shown in Figure 7.4 and is in coherence with our assumption that when the switch should be towards the *global semantic similarity*, the L1 norm of $\mathbf{y}$ should be low and high when the switch should be towards the *local contextual similarity*. Thus, Figure 7.4 confirms our intuition.

7.5.4 Handling unseen and shared vocabulary

As discussed in Section 7.3.3, the vocabulary in the dataset is divided into two disjoint sets : the resolved set $\mathcal{R}$ and the unresolved set $\mathcal{U}$. The training set in the proposed word-level identification frameworks are made up of words from the resolved set and the test set consists of words all from the unresolved set $\mathcal{U}$. Since $\mathcal{R}$ and $\mathcal{U}$ are disjoint, it implies that the test set is made up of samples none of which are seen during the training. In other words, the test set is made up of all unseen/out-of-vocabulary words.

Further, it has also been discussed in Section 7.3.3 that the words in $\mathcal{U}$ are words

that occur across sentences in multiple languages. These occurrences could be either due to the embedding/borrowing of words from a different language into sentences of native language. These occurrences could also be due to inherent similarities between languages or could be the result of irregular transliteration. Since the test set is made of all words from $\mathcal{U}$, this also implies that the test set also consists of shared vocabulary.

Therefore, the macro and micro-average F-scores of 90.79% and 91.03% obtained using the proposed framework demonstrate's the model's ability to address out-of-vocabulary and shared vocabulary issue in language identification.

7.6 Summary

This study investigates the problem of word-level language identification for code-mixed social media text in a multilingual environment. The objective is to boost language identification performance by combining outputs from different baseline classifiers each of which capture information that are non-complementary in nature. Results obtained indicate that the proposed framework is able to make correct choice between the outputs of the baseline classifiers when one of the outputs is incorrect. The results obtained also indicate the effectiveness of the proposed model in addressing shared vocabulary and out-of-vocabulary issue in language identification in code-mixed social media data. Further, the proposed model uses minimum annotated resources and no external resources which also makes it a suitable framework for language identification in low resource scenarios.

Even though the proposed approach shows improved performance compared to the baseline components, the performance is still significantly less than the best performance that can be obtained by selecting the best predictions from each of the baseline components. Therefore, in future, our efforts will be directed towards boosting the performance of the proposed framework further so that performance close to the maximum attainable performance is achieved. An important observation from this study is that non-textual features like neighbourhood based features are as good as text based features in capturing information necessary for language identification. One of the limitations of the neighbourhood based features used in

this study is that they are derived from the word embeddings that are noisy in nature. Therefore, in future, we would like to explore better non-textual features (neighbourhood based/distance based features) that can excel the performance of the text based features. Another area of exploration would be to boost word-level language identification assuming availability of sufficient resources like annotated corpus, dictionaries, normalization tools etc.

8

Conclusion and Future Work

This thesis work explores the problem of automatic language identification over code-mixed social media text in a highly multilingual environment. While the techniques explored are generic in nature and not specific to any languages, the principal datasets considered in this study comprise languages commonly spoken in Assam, a state in the North-Eastern part of India. Messages in languages other than English are phonetically typed. This study specifically considers phonetically typed social media messages such that all languages share a common script (Roman script in this case). The study particularly focuses on minimum manual audit and minimum usage of external resources so that the proposed techniques can be easily extended to newer languages with fewer resources. In this chapter, we summarize the contributions made in this thesis work in automatic language identification. We further discuss the limitations and possible future directions to explore.

8.1 Summary of Contributions

The contributions in this thesis work can broadly be divided into two categories. First, in terms of resources, as a part of this thesis work, we generate three manually annotated and three automatically annotated language identification datasets. Second, in terms of automatic language identification, this thesis work makes four major contributions which are summarized below.

First, the language characteristics of user conversations in a multilingual social media environment is studied. Although user language dynamics in social networks have been studied previously, with respect to Indian languages, most studies are centered around a few languages like Hindi, Tamil, Kannada etc. None of the existing studies have so far considered exploring languages in Assam. This study considers Assam as the target location and considers a multilingual environment comprising six languages - Assamese, Bengali, Hindi, English, Karbi and Boro. It is observed that while 50.91% of all users are multilingual, majority section of the multilingual users use two languages with a very small section using three or more languages. It is also observed that most multilingual users tend to use a local language and a global languages (English in the case of this study) than using multiple local languages. Further, it is observed that the language choices of users in a multilingual environment is largely influenced by fellow participants. However, it is also likely the language choices are also influenced by other factors such as underlying emotions, target audience etc. These factors have not been explored in this study and is left to be explored in future. These observations can be useful for different computational and linguistic analyses and can also pave the way for advanced analyses such as information diffusion across language communities, cross-lingual dynamics etc.

Second, we propose a sentence-level language identification technique taking advantages of the social and conversational features in user conversations. Unlike previous studies that use social conversational features as language priors, the proposed techniques use social and conversational features to refine a noisy code-mixed text and generate a new language bias text. From various experimental observations, it is observed that social and conversational features help in enhancing language identification performance. We obtain an improvement of 5.5%, 6.8% and 2.3% over Facebook, YouTube and Twitter datasets respectively over the best performing baseline counterparts. However, the proposed methods are dependent on social and conversational characteristics. Therefore, if there are not enough social conversational information available, the proposed techniques may show a drop in performance .

Third, we propose different word-level language identification techniques using

sentence-level annotations using global semantic similarity and local contextual similarity. The proposed methods are inexpensive in terms of the resources required. The minimal requirement of resources and the significantly encouraging performance (85.1% max F-score) makes the proposed framework an acceptable method for word level language identification for low resource languages in a highly multilingual environment. Further, it is observed that while global semantic similarity is capable of identifying borrowed or embedded words, local contextual similarity is capable of resolving languages of words that are valid in multiple languages. Therefore exploring methods that can combine advantages of both global semantic similarity and local contextual similarity are directions to explore in future.

Fourth, we propose a word-level identification framework that makes use of a dynamic switching mechanism to make the correct choice between the output of two different classifiers when one of the outputs may be incorrect. The method is particularly useful when the constituent classifiers are trained over a set of non-complementary features. Results obtained show that the proposed technique is able to make the correct choice between different classification outcomes when one of the classification outcomes is incorrect. The proposed technique shows encouraging performance achieving a macro-average F-score of 90.79% and outperforming the baseline counterparts by 3.96%.

Overall, this study is driven by the objective of accurately identifying languages in code-mixed social media text in a highly multilingual environment while using minimal manual audit and minimal external resources. The proposed sentence and word-level language identification techniques show encouraging performances in a multilingual environment. However, there are certain aspects pertaining to multilingual user characteristics and language identification that have not been addressed in this study. These are discussed in the following section.

8.2 Limitations and Future Works

This section discusses the limitations associated with the current study and some of the potential directions to explore in future.

Datasets : Large sized annotated datasets are very useful for various forms of computational and linguistic analyses. The larger dataset comprising of 583,486 Facebook comments and 6,221 YouTube comments is obtained using a classifier trained over a fixed size manually annotated dataset and hence is subject to classification error. In future, efforts may be directed towards semi-supervised learning or active learning techniques to incrementally annotate the larger dataset with more accurate class labels.

User Language Characteristics in Multilingual Social Media Conversations: While the investigation of multilingual characteristics of user conversations leads to interesting observations, there are certain limitations that we plan to address in future. First, the influence of factors like underlying emotion, target audience etc. on the language choice of a user in a multilingual conversation have not been explored in this study. Second, we have also observed in this study that a large percentage of native language comments consist of naturally embedded borrowed words. Systematic patterns, if any, in the nature of the embedded words have not been explored in this study. Analysing these two aspects will give more insights with respect to user language characteristics in a multilingual environment and may also be useful for different computational as well as linguistic and sociological analyses. Further, cross-platform analyses of data is another direction to explore in future. This will help in understanding the feasibility of using models trained over data from one platform to classify data collected from other platforms.

Sentence-Level Language Identification : The sentence-level language identification techniques proposed in this study help in enhancing language identification performance and show significant improvement in comparison to the baseline counterparts. However, the proposed techniques are dependent on social conversational features. Therefore, in situations where enough conversational information surrounding a user comment is not available, there may be a drop in performance. Therefore, further investigation to obtain improved performance even in absence of conversational information is required. One possible direction to explore is to incorporate information from monolingual language models for sentence language

identification. However, since, we are studying language identification over transliterated text and it is difficult to obtain monolingual language models in transliterated text, incorporating information from monolingual language models would require an intermediate transliteration module that either back-transliterate the text from Roman scripts into native scripts or forward-transliterate the text in native script to Roman script.

Word-Level Language Identification : The word-level language identification performances achieved in this study is satisfactory and significant considering the less amount of resources used. However, we believe that there still exists room for improvement. An analysis of the incorrect predictions shows that incorrect predictions are primarily due to two reasons (1) noisy and infrequent word forms and (2) noisy context. As such, to further improve ALI performance, it is assumed that we need to incorporate external information from resources such as dictionaries, monolingual corpora etc. Again, as this study considers the phonetically typed or transliterated messages, to be able to use the resources in the native scripts, there needs to be an efficient transliteration system for back transliterating the noisy social media text into the native forms. Back transliterating the noisy social media text and incorporating the obtained information into language identification frameworks to obtain improved performance is an important area to explore in future.

Bibliography

P. Agarwal, A. Sharma, J. Grover, M. Sikka, K. Rudra, and M. Choudhury. I may talk in english but gaali toh hindi mein hi denge: A study of english-hindi code-switching and swearing pattern on social networks. In *Proceedings of the International Conference on Communication Systems and Networks*, pages 554–557, 2017.

P. Agarwal, K. Garimella, S. Joglekar, N. Sastry, and G. Tyson. Characterising user content on a multi-lingual social network. In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 2–11, Atlanta, USA, 2020.

M. Al-Badrashiny, H. Elfardy, and M. 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, 2015.

A. Ali, N. Dehak, P. Cardinal, S. Khurana, S. H. Yella, J. Glass, P. Bell, and S. Renals. Automatic dialect detection in arabic broadcast speech. In *Proceedings of Annual Conference of the International Speech Communication Association*, pages 2934–2938, San Francisco, USA, 2016.

A. Amine, Z. Elberrichi, and M. Simonet. Automatic language identification: An alternative unsupervised approach using a new hybrid algorithm. *International Journal of Computer Science and Applications*, 7(1):94–107, 2010.

T. Baldwin and M. Lui. Language identification: The long and the short of the matter. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 229–237, Los Angeles, CA, 2010.

T. Baldwin, P. Cook, M. Lui, A. MacKinlay, and L. Wang. How noisy social media text, how diffrnt social media sources? In *Proceedings of the 6th International*

Joint Conference on Natural Language Processing, pages 356–364, Nagoya, Japan, 2013.

S. Banerjee, A. Kuila, A. Roy, S. K. Naskar, P. Rosso, and S. Bandyopadhyay. A hybrid approach for transliterated word-level language identification: Crf with post-processing heuristics. In *Proceedings of the 2014 Forum for Information Retrieval Evaluation*, pages 54–59, Bangalore, India, 2014.

U. Barman, A. Das, J. Wagner, and J. 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, 2014.

P. Barros, S. Magg, C. Weber, and S. Wermter. A multichannel convolutional neural network for hand posture recognition. In *Proceedings of the International Conference on Artificial Neural Networks*, pages 403–410, Hamburg, Germany, 2014.

A. Bawa, M. Choudhury, and K. Bali. Accommodation of conversational code-choice. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 82–91, Melbourne, Australia, 2018.

S. Bergsma, P. McNamee, M. Bagdouri, C. Fink, and T. Wilson. Language identification for creating language-specific twitter collections. In *Proceedings of the Second workshop on Language in Social Media*, pages 65–74, Montreal, Canada, 2012.

Y. Bestgen. Improving the character ngram model for the dsl task with bm25 weighting and less frequently used feature sets. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 115–123, Valencia, Spain, 2017.

C. Biemann and S. Teresniak. Disentangling from babylonian confusion–unsupervised language identification. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 773–784, Mexico City, Mexico, 2005.

A. Bohra, D. Vijay, V. Singh, S. S. Akhtar, and M. Shrivastava. A dataset of hindi-english code-mixed social media text for hate speech detection. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*, pages 36–41, New Orleans, Louisiana, 2018.

R. D. Brown. Finding and identifying text in 900+ languages. *Digital Investigation*, 9:S34–S43, 2012.

R. D. Brown. Selecting and weighting n-grams to identify 1100 languages. In *Proceedings of the International Conference on Text, Speech and Dialogue*, pages 475–483, Pilsen, Czech Republic, 2013.

R. D. Brown. Non-linear mapping for improved identification of 1300+ languages. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 627–632, Doha, Qatar, 2014.

B. Bullock, W. Guzmán, J. Serigos, V. Sharath, and A. J. Toribio. Predicting the presence of a matrix language in code-switching. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 68–75, Melbourne, Australia, 2018.

S. Carter, W. Weerkamp, and M. Tsagkias. Microblog language identification: Overcoming the limitations of short, unedited and idiomatic text. *Language Resources and Evaluation Journal*, 47(1):195–215, 2013.

W. B. Cavnar and J. M. Trenkle. N-gram-based text categorization. In *Proceedings of Third Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, Las Vegas, USA, 1994.

H. Ceylan and Y. Kim. Language identification of search engine queries. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1066–1074, Suntec, Singapore, 2009.

K. Chandu, T. Manzini, S. Singh, and A. W. Black. Language informed modeling of code-switched text. In *Proceedings of the Third Workshop on Computational*

Approaches to Linguistic Code-Switching, pages 92–97, New Orleans, Louisiana, 2018.

G. Chittaranjan, Y. Vyas, K. Bali, and M. Choudhury. Word-level language identification using crf: Code-switching shared task report of msr india system. In *Proceedings of The First Workshop on Computational Approaches to Code Switching*, pages 73–79, Doha, Qatar, 2014.

A. M. Ciobanu and L. P. Dinu. A computational perspective on the romanian dialects. In *Proceedings of the International Conference on Language Resources and Evaluation*, pages 3281–3285, Portoroz, Slovenia, 2016.

S. Clematide and P. Makarov. Cluzh at vardial gdi 2017: Testing a variety of machine learning tools for the classification of swiss german dialects. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 170–177, Valencia, Spain, 2017.

S. Coats. European language ecology and bilingualism with english on twitter. In *Proceedings of the Conference on CMC and Social Media Corpora for the Humanities*, pages 35–38, Bolzano, Italy, 2017.

J. Cowie, Y. Ludovic, and R. Zacharski. Language recognition for mono-and multilingual documents. In *Proceedings of the VexTal Conference*, pages 209–214, Venice, Italy, 1999.

A. Das and B. Gamback. Identifying languages at the word level in code-mixed indian social media text. In *Proceedings of the International Conference on Natural Language Processing*, pages 378–387, Goa, India, 2014.

S. D. Das, S. Mandal, and D. Das. Language identification of bengali-english code-mixed data using character & phonetic based lstm models. In *Proceedings of the Forum for Information Retrieval Evaluation*, pages 60–64, Kolkata, India, 2019.

N. Dongen. Analysis and prediction of dutch-english code-switching in dutch social media messages. *Master's thesis, Universiteit van Amsterdam, Amsterdam, Netherlands*, 2017.

I. Eleta and J. Golbeck. Bridging languages in social networks: How multilingual users of twitter connect language communities? *American Society for Information Science and Technology*, 49(1):1–4, 2012.

I. Eleta and J. Golbeck. Multilingual use of twitter: Social networks at the language frontier. *Computers in Human Behavior*, 41(1):424–432, 2014.

R. Eskander, M. Al-Badrashiny, N. Habash, and O. 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, 2014.

R. O. G. Gavilanes, D. Gomez, D. Parra Santander, C. Trattner, A. Kaltenbrunner, and E. Graells. Language, twitter and academic conferences. In *Proceedings of the ACM Conference on hypertext & social media*, pages 159–163, Guzelyurt, Northern Cyprus, 2015.

S. Gella, K. Bali, and M. Choudhury. "ye word kis lang ka hai bhai?" testing the limits of word level language identification. In *Proceedings of the International Conference on Natural Language Processing*, pages 368–377, Goa, India, 2014.

O. Giwa and M. H. Davel. N-gram based language identification of individual words. In *Proceedings of the Annual Symposium of the Pattern Recognition Association of South Africa*, pages 15–22, Johannesburg, South Africa, 2013.

M. Goldszmidt, M. Najork, and S. Paparizos. Boot-strapping language identifiers for short colloquial postings. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 95–111, Prague, Czech Republic, 2013.

G. Grefenstette. Comparing two language identification schemes. In *Proceedings of the International Conference on Statistical Analysis of Textual Data*, pages 263–268, Rome, Italy, 1995.

S. Gundapu, I. KCIS, and R. Mamidi. Word level language identification in english telugu code mixed data. In *Proceedings of the Pacific Asia Conference on Language, Information and Computation*, pages 180–186, Hong Kong, 2018.

J. Hakkinen and J. Tian. N-gram and decision tree based language identification for written words. In *IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 335–338, San Juan, Puerto Rico, 2001.

S. A. Hale. Global connectivity and multilinguals in the twitter network. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 833–842, Ontario, Canada, 2014.

A. Hategan, B. Barliga, and I. Tabus. Language identification of individualwords in a multilingual automatic speech recognition system. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4357–4360, Tapei, Taiwan, 2009.

P. Henrich. Language identification for the automatic grapheme-to-phoneme conversion of foreign words in a german text-to-speech system. In *Proceedings of the European Conference on Speech Communication and Technology*, Paris, France, 1989.

J. Hochberg, K. Bowers, M. Cannon, and P. Kelly. Script and language identification for handwritten document images. *International Journal on Document Analysis and Recognition*, 2(2-3):45–52, 1999.

N. Hollenstein and N. Aepli. A resource for natural language processing of swiss german dialects. In *Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology*, Essen, Germany, 2015.

A. S. House and E. 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.

C.-R. Huang and L.-H. Lee. Contrastive approach towards text source classification based on top-bag-of-word similarity. In *Proceedings of the Pacific Asia Conference on Language, Information and Computation*, pages 404–410, Bali, Indonesia, 2008.

B. Hughes, T. Baldwin, S. Bird, J. Nicholson, and A. MacKinlay. Reconsidering language identification for written language resources. In *Proceedings of the International Conference on Language Resources and Evaluation*, pages 485,488, Genoa, Italy, 2006.

A. Jaech, G. Mulcaire, S. Hathi, M. Ostendorf, and N. 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, 2016.

T. Jauhiainen, K. Lindén, and H. Jauhiainen. Language set identification in noisy synthetic multilingual documents. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 633–643. Springer, 2015.

T. Jauhiainen, K. Lindén, and H. Jauhiainen. Evaluation of language identification methods using 285 languages. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 183–191, Gothenburg, Sweden, 2017.

H. Jhamtani, S. K. Bhogi, and V. Raychoudhury. Word-level language identification in bi-lingual code-switched texts. In *Proceedings of the Pacific Asia Conference on Language, Information and Computing*, pages 348–357, Phuket, Thailand, 2017.

H. Jin. Detection and characterization of influential cross-lingual information diffusion on social networks. In *Proceedings of the International Conference on World Wide Web Companion*, pages 741–745, Perth, Australia, 2017.

D. Jurgens, Y. Tsvetkov, and D. Jurafsky. Incorporating dialectal variability for socially equitable language identification. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 51–57, Vancouver, Canada, 2017.

R. Kapoor, Y. Kumar, K. Rajput, R. R. Shah, P. Kumaraguru, and R. Zimmermann. Mind your language: Abuse and offense detection for code-switched languages. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 9951–9952, Hawaii, USA, 2019.

S. Kim, I. Weber, L. Wei, and A. Oh. Sociolinguistic analysis of twitter in multilingual societies. In *Proceedings of the ACM Conference on Hypertext and Social Media*, pages 243–248, Santiago, Chile, 2014.

Y. Kim. Convolutional neural networks for sentence classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751, Doha, Qatar, 2014.

B. King and S. Abney. Labeling the languages of words in mixed-language documents using weakly supervised methods. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1110–1119, Atlanta, 2013.

B. King, D. Radev, and S. Abney. Experiments in sentence language identification with groups of similar languages. In *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, pages 146–154, Dublin, Ireland, 2014.

T. Kocmi and O. Bojar. Lanidenn: Multilingual language identification on character window. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*, pages 927–936, Valencia, Spain, 2017.

S. Konstantopoulos. What's in a name? In *Proceedings of the Conference on Recent Advances in Natural Language Processing*, Borovets, Bulgaria, 2007.

A. Lantz-Andersson. Language play in a second language: Social media as contexts for emerging sociopragmatic competence. *Education and Information Technologies*, 23(2):705–724, 2018.

N. Ljubesic, N. Mikelic, and D. Boras. Language indentification: How to distinguish similar languages? In *Proceedings of the International Conference on Information Technology Interfaces.*, pages 541–546, Cavtat, Croatia.

M. Lui and P. Cook. Classifying english documents by national dialect. In *Proceedings of the Australasian Language Technology Association Workshop 2013*, pages 5–15, Brisbane, Australia, 2013.

M. Lui, J. H. Lau, and T. Baldwin. Automatic detection and language identification of multilingual documents. *Transactions of the Association for Computational Linguistics*, 2:27–40, 2014.

S. MacNamara, P. Cunningham, and J. Byrne. Neural networks for language identification: a comparative study. *Information processing & management*, 34(4): 395–403, 1998.

M. Mager, O. Cetinoglu, and K. Kann. Subword-level language identification for intra-word code-switching. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2005–2011, Minneapolis, Minnesota, 2019.

S. Mandal and A. K. Singh. Language identification in code-mixed data using multichannel neural networks and context capture. In *Proceedings of The Fourth Workshop on Noisy User-generated Text*, pages 116–120, Brussels, Belgium, 2018.

S. Mandal, S. Banerjee, S. K. Naskar, P. Rosso, and S. Bandyopadhyay. Adaptive voting in multiple classifier systems for word level language identification. In *FIRE workshops*, pages 47–50, 2015.

T. Mandl, M. Shramko, O. Tartakovski, and C. Womser-Hacker. Language identification in multi-lingual web-documents. In *Proceedings of the International Conference on Application of Natural Language to Information Systems*, pages 153–163, Klagenfurt, Austria, 2006.

B. Martins and M. J. Silva. Language identification in web pages. In *Proceedings of the ACM symposium on Applied computing*, pages 764–768, Santa Fe, USA, 2005.

L. A. Mather. A linear algebra approach to language identification. In *International Workshop on Principles of Digital Document Processing*, pages 92–103, Saint Malo, France, 1998.

D. Mave, S. Maharjan, and T. Solorio. Language identification and analysis of code-switched social media text. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, Melbourne, Australia, 2018.

P. McNamee. Language identification: a solved problem suitable for undergraduate instruction. *Journal of computing sciences in colleges*, 20(3):94–101, 2005.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013.

Y. Miyamoto and K. Cho. Gated word-character recurrent language model. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1992–1997, Texas, USA, 2016.

G. Molina, N. Rey-Villamizar, T. Solorio, F. AlGhamdi, M. Ghoneim, A. Hawwari, and M. Diab. Overview for the second shared task on language identification in code-switched data. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 40–49, Texas, USA, 2016.

Y. Nakamura. Identification of languages with short sample texts : A linguometric study. *Library and Information Science*, 9:459–481, 1971.

I. Ndubuisi-Obi, S. Ghosh, and D. Jurgens. Wetin dey with these comments? modeling sociolinguistic factors affecting code-switching behavior in nigerian online discussions. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 6204–6214, Florence, Italy, 2019.

D. Nguyen and L. Cornips. Automatic detection of intra-word code-switching. In *Proceedings of the Fourteenth SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 82–86, Berlin, Germany, 2016.

D. Nguyen and A. S. Doğruöz. Word level language identification in online multilingual communication. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 857–862, Washington, USA, 2013.

D. Nguyen, D. Trieschnigg, and L. Cornips. Audience and the use of minority languages on twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 666–669, Oxford, UK, 2015.

E. Papalexakis, D. Nguyen, and A. S. Doğruöz. Predicting code-switching in multilingual communication for immigrant communities. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 42–50, Doha, Qatar, 2014.

J. Patro, B. Samanta, S. Singh, A. Basu, P. Mukherjee, M. Choudhury, and A. Mukherjee. All that is English may be Hindi: Enhancing language identification through automatic ranking of the likeliness of word borrowing in social media. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 2264–2274, Copenhagen, Denmark, 2017.

F. Peng and D. Schuurmans. Combining naive bayes and n-gram language models for text classification. In *Proceedings of the European Conference on Information Retrieval*, pages 335–350, Pisa, Italy.

J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543, Doha, Qatar, 2014.

G. Pethö and E. Mózes. An n-gram-based language identification algorithm for variable-length and variable-language texts. *Argumentum*, 10:56–82, 2014.

S. Phani, S. Lahiri, and A. Biswas. Sentiment analysis of tweets in three indian languages. In *Proceedings of the Sixth Workshop on South and Southeast Asian Natural Language Processing*, pages 93–102, Osaka, Japan, 2016.

M. Piergallini, R. Shirvani, G. S. Gautam, and M. Chouikha. Word-level language identification and predicting codeswitching points in swahili-english language data. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 21–29, Texas, USA, 2016.

J. M. Prager. Linguini: Language identification for multilingual documents. *Journal of Management Information Systems*, 16(3):71–101, 1999.

B. Ranaivo-Malançon. Automatic identification of close languages-case study: Malay and indonesian. *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, 2(2):126–134, 2006.

F. Rangel, M. Franco-Salvador, and P. Rosso. A low dimensionality representation for language variety identification. In *Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics*, pages 156–169, Turkey, 2016.

M. Rei, G. Crichton, and S. Pyysalo. Attending to characters in neural sequence labeling models. In *Proceedings of the 26th International Conference on Computational Linguistics*, pages 309–318, Osaka, Japan, 2016.

S. Rijhwani, R. Sequiera, M. Choudhury, K. Bali, and C. S. Maddila. Estimating code-switching on twitter with a novel generalized word-level language detection technique. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1971–1982, Vancouver, Canada, 2017.

K. Rudra, S. Rijhwani, R. Begum, K. Bali, M. Choudhury, and N. Ganguly. Understanding language preference for expression of opinion and sentiment: What do hindi-english speakers do on twitter? In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1131–1141, Texas, USA, 2016.

K. Rudra, A. Sharma, K. Bali, M. Choudhury, and N. Ganguly. Identifying and analyzing different aspects of english-hindi code-switching in twitter. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 18(3), 2019.

Y. Samih, S. Maharjan, M. Attia, L. Kallmeyer, and T. Solorio. Multilingual code-switching identification via lstm recurrent neural networks. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 50–59, Texas, USA, 2016.

A. Selamat and N. C. Ching. Arabic script documents language identifications using fuzzy art. In *Proceedings of the Asia International Conference on Modelling & Simulation*, pages 528–533, Kuala Lumpur, Malaysia, 2008.

H. Shi, T. Ushio, M. Endo, K. Yamagami, and N. Horii. A multichannel convolutional neural network for cross-language dialog state tracking. In *Proceedings of*

the Sixth IEEE Workshop on Spoken Language Technology, pages 559–564, San Diego, California, 2016.

K. Shiells and P. Pham. Unsupervised clustering for language identification. 2010.

P. Shoemark, D. Sur, L. Shrimpton, I. Murray, and S. Goldwater. Aye or naw, whit dae ye hink? scottish independence and linguistic identity on social media. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*, pages 1239–1248, Kyiv, Ukraine, 2017.

U. K. Sikdar and B. 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, Texas, USA, 2016.

K. Singh, I. Sen, and P. Kumaraguru. Language identification and named entity recognition in hinglish code mixed tweets. In *Proceedings of ACL Student Research Workshop*, pages 52–58, Melbourne, Australia, 2018a.

K. Singh, I. Sen, and P. Kumaraguru. A twitter corpus for hindi-english code mixed pos tagging. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pages 12–17, Melbourne, Australia, 2018b.

T. Solorio, E. Blair, S. Maharjan, S. Bethard, M. Diab, M. Ghoneim, A. Hawwari, F. AlGhamdi, J. Hirschberg, and A. Chang. Overview for the first shared task on language identification in code-switched data. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 62–72, Doha, Qatar, 2014.

S. Stefanich, J. Cabrelli Amaro, D. Hilderman, and J. A. Archibald. The morphophonology of intraword codeswitching: Representation and processing. *Frontiers in Communication*, 4:54, 2019.

I. Stewart, Y. Pinter, and J. Eisenstein. Si o no, que penses? catalonian independence and linguistic identity on social media. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 136–141, New Orleans, Louisiana, 2018.

L. Tan, M. Zampieri, N. Ljubešic, and J. Tiedemann. Merging comparable data sources for the discrimination of similar languages: The dsl corpus collection. In *Proceedings of the Seventh Workshop on Building and Using Comparable Corpora*, pages 11–15, Reykjavik, Iceland, 2014.

J. Tiedemann and N. Ljubešić. Efficient discrimination between closely related languages. In *Proceedings of the International Conference on Computational Linguistics*, pages 2619–2634, Mumbai, India, 2012.

E. Tromp and M. Pechenizkiy. Graph-based n-gram language identification on short texts. In *Proceedings of the Machine Learning conference of Belgium and The Netherlands*, pages 27–34, The Hague, Netherlands, 2011.

C. van der Lee and A. 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*, pages 190–199, Valencia, Spain, 2017.

T. Vatanen, J. J. Väyrynen, and S. Virpioja. Language identification of short text segments with n-gram models. In *Proceedings of the International Conference on Language Resources and Evaluation*, pages 3423–3430, Valletta, Malta, 2010.

J. Vogel and D. Tresner-Kirsch. Robust language identification in short, noisy texts: Improvements to liga. In *Proceedings of the International Workshop on Mining Ubiquitous and Social Environments*, pages 43–50, Bristol, UK, 2012.

S. Volkova, S. Ranshous, and L. Phillips. Predicting foreign language usage from english-only social media posts. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 608–614, New Orleans, Louisiana, 2018.

Y. Vyas, S. Gella, J. Sharma, K. Bali, and M. Choudhury. Pos tagging of english-hindi code-mixed social media content. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 974–979, Doha, Qatar, 2014.

P. Wang, N. Bojja, and S. Kannan. A language detection system for short chats in mobile games. In *Proceedings of the International Workshop on Natural Language Processing for Social Media*, pages 20–28, Denver, Colorado, 2015.

M. X. Xia. Codeswitching language identification using subword information enriched word vectors. In *Proceedings of The Second Workshop on Computational Approaches to Code Switching*, pages 132–136, Texas, USA, 2016.

X. Yang and W. Liang. An n-gram-and-wikipedia joint approach to natural language identification. In *Proceedings of the International Universal Communication Symposium*, pages 332–339, Beijing, China., 2010.

Z. Yirmibeşoğlu and G. Eryiğit. Detecting code-switching between turkish-english language pair. In *Proceedings of The Fourth Workshop on Noisy User-generated Text*, pages 110–115, Brussels, Belgium, 2018.

M. Zampieri and B. G. Gebre. Automatic identification of language varieties: The case of portuguese. In *Proceedings of The Conference on Natural Language Processing*, pages 233–237, Vienna, Austria, 2012.

M. Zampieri, B. G. Gebre, and S. 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, 2013.

M. Zampieri, S. Malmasi, O.-M. Sulea, and L. P. Dinu. A computational approach to the study of portuguese newspapers published in macau. In *Proceedings of Workshop on Natural Language Processing Meets Journalism*, pages 47–51, New York, USA, 2016.

A. Zečević and S. Vujicic-Stankovic. The mysterious letter j. In *Proceedings of the Workshop on Adaptation of Language Resources and Tools for Closely Related Languages and Language Variants*, pages 40–44, Hissar, Bulgaria, 2013.

Y. Zhang, J. Riesa, D. Gillick, A. Bakalov, J. Baldridge, and D. Weiss. A fast, compact, accurate model for language identification of codemixed text. In *Proceedings*

of the Conference on Empirical Methods in Natural Language Processing, pages 328–337, Brussels, Belgium, 2018.

G. Zhu, X. Yu, Y. Li, and D. Doermann. Language identification for handwritten document images using a shape codebook. *pattern recognition*, 42(12):3184–3191, 2009.

A. Zubiaga, I. San Vicente, P. Gamallo, J. R. Pichel, I. Alegria, N. Aranberri, A. Ezeiza, and V. Fresno. Tweetlid: a benchmark for tweet language identification. *Language Resources and Evaluation Journal*, 50(4):729–766, 2016.

Publications

Journals

- **Neelakshi Sarma**, Sanasam Ranbir Singh and Diganta Goswami,"Influence of Social Conversational Features on Language Identification in Highly Multilingual Online Conversations", *Information Processing and Management, Elsevier*, Volume 56, Issue 1, January 2019, Pages 151-166

- **Neelakshi Sarma**, Sanasam Ranbir Singh and Diganta Goswami. "Identifying Languages at the Word Level in Assamese-Bengali-Hindi-English Code-Mixed Social Media Text", *International Journal of Asian Language Processing, COLIPS,* Volume 29, Issue 1, November 2019, Pages 35-48

- **Neelakshi Sarma**, Sanasam Ranbir Singh and Diganta Goswami, "Switch-Net : Learning to Switch for Word Level Language Identification in Code-Mixed Social Media Text", *Natural Language Engineering, Cambridge University Press* [Accepted]

- **Neelakshi Sarma**, Sanasam Ranbir Singh and Diganta Goswami, "User Language Characteristics in Highly Multilingual Conversations in Social Media Data",*Computers in Human Behaviour Reports, Elsevier* [Under Review]

Conference

- **Neelakshi Sarma**, Sanasam Ranbir Singh and Diganta Goswami, "Word Level Language Identification in Assamese-Bengali-Hindi-English Code-Mixed Social Media Text". In *Proceedings of the 22nd International Conference on Asian Language Processing (IALP)* , 2018, Bandung, Indonesia. pp. 261-266.