Towards Normalising Konkani-English Code-Mixed Social Media Text

Akshata Phadte
DCST Goa University
Goa, India.
akshataph07@gmail.com

Gaurish Thakkar
DCST Goa University
Goa, India.
thak123@gmail.com

Abstract

In this paper, we present an empirical study on problem of word-level language identification and text normalization for Konkani-English Code-Mixed Social Media Text (CMST). we describe a new dataset which contains of more than thousands posts from Facebook posts that exhibit code mixing between Konkani-English. To the best of our knowledge, our work is the first attempt at the creation of a linguistic resource for this language pair which will be made public and developed a language identification and Normalisation System for Konkani-Englsih language pair.

We also present word-level language identification experiments are performed using this dataset. Different techniques are employed, including a simple unsupervised dictionary-based approach, supervised word-level Language identification using sequence labelling using Conditional Random Fields based models, SVM, Random Forest. The targeted research problem also entails solving another problem, that to correct English spelling errors in code-mixed social media text that contains English words as well as Romanized transliteration of words from another language, in this case Konkani.

1 Introduction

Social media in todays world possess enormous amount of data. But the problem starts in Multilingual speakers tend to exhibit code-mixing and code-switching in their use of language on social media platforms. Now Automatic understanding of Social Media text is unravelling a whole new field of study. English is still found the most popular language in Social Media Text, its dominance is receding. Code mixing occurs due to various reasons. According to a work by (Hidayat, 2012), There are the following major reasons for Code-Mixing:-

- **45%: Real lexical needs :** For instance someone is thinking of some object but is not able to recall the word in the language, then he/she will tend to switch to a language where he knows the appropriate word.
- **40%: Talking about a particular topic people :** tend to talk about some topics in their mother tongue (like food) and generally while discussing science people tend to switch to English.
- **5%: for content clarification :** while explaining one topic, for better clarification of the audience, to make the audience more clear about the topic, code switching is used.

Konkani-English bilingual speakers produce huge amounts of code-mixed social media text (CSMT). (Vyas et al., 2014) noted that the complexity in analyzing code-mixed social media text (CSMT) stems from nonadherence to a formal grammar, spelling variations, lack of annotated data, inherent conversational nature of the text and ofcourse, code-mixing. Therefore, there is a need to create datasets and Natural Language Processing (NLP) tools for code-mixed social media text (CSMT) as traditional tools are ill-equipped for it. Taking a step in this direction, we describe the Word Level Language Identification system for Konkani-English language pair that we will be building in this study. The salient contributions of this work are in formalizing the problem and related challenges for processing of Konkani-English social media data, creation of an annotated

S Bandyopadhyay, D S Sharma and R Sangal. Proc. of the 14th Intl. Conference on Natural Language Processing, pages 85–94,
Kolkata, India. December 2017.

dataset and initial experiments for language identification of this data.

2 Related Work

A lot of work has been done on social media data and code-mixed data over the past decades. Code-mixing being a relatively newer phenomena has gained attention of researchers only in the past two decades. On the other hand, Language Identification has been considered to be a solved problem by (McNamee, 2005), but new complications were added to this task in the context of code-mixed social media data. Similarly, Word Normalisation has been extensively studied, but there is little work done on the Konkani-English , Hindi-English language pair.

2.1 Code-Mixed Data

One of the earliest works on code-Mixing for Facebook data was done by (Hidayat, 2012) and showed that Facebook users tend to mainly use inter-sentential switching over intra-sentential, and report that 45% of the switching was instigated by real lexical needs, 40% was used for talking about a particular topic, and 5% for content clarification .

(Dey and Fung, 2014) also investigated the rules for code-switching in Hindi-English data by interviewing bilingual students and transcribing their utterances . They found that on average, roughly 67% of each sentence were made up of Hindi words and 33% English words.

2.2 Language Identification

Previous work on text has mainly been on identifying a language from documents of several languages, such that even when evidence is collected at word level, evaluation is at document level (Prager, 1999); (Singh and Gorla, 2007); (Yamaguchi and Tanaka-Ishii, 2012). (Carter et al., 2013) collected tweets in five different European languages and analysed multi-lingual microblogs for understanding the dominant language in any specific tweet . He then performed post-level language identification, experimenting with a range of different models and a character n-gram distance metric, reporting a best overall classification accuracy of 92.4%. (Tratz et al., 2013) on the other hand worked on highly code mixed tweets, with 20.2% of their test and development sets consisting of tweets in more than one language. They aimed to separate Romanised Moroccan, Arabic (Darija), English and French tweets using a Maximum Entropy classifier, achieving F-scores of 0.928 and 0.892 for English and French, but only 0.846 for Darija due to low precision.

(Nguyen and Dogruoz, 2013) worked on language identification at the word level on randomly sampled Turkish-Dutch posts from an online chat forum . They compared dictionary based methods to statistical ones. Their best system reached an accuracy of 97.6%, but with a substantially lower accuracy on post level (89.5%), even though 83% of the posts actually were monolingual. They report on language identification experiments performed on Turkish and Dutch forum data. Experiments have been carried out using language models, dictionaries, logistic regression classification and Conditional Random Fields. They find that language models are more robust than dictionaries and that contextual information is helpful for the task.

Furthermore, (Barman et al., 2014) investigated language identification at word level on Bengali-Hindi-English code-mixed social media text . They annotated a corpus with more than 180,000 tokens and achieved an accuracy of 95.76% using statistical models with monolingual dictionaries.

3 Normalisation

Owing to massive growth of SMS and social media content, text normalisation systems have gained attention where the focus is on conversion of these tokens into standard dictionary words.The first Chinese monolingual chat corpus was released by (Wong and Xia, 2008). They also introduced a word normalisation model, which was a hybrid of the Source Channel Model and phonetic mapping model.

(Wang et al., 2009) work with abbreviations for spoken Chinese rather than for English text messages. They first perform an abbreviation generation task for words and then reverse the mapping in a look-up table. They use conditional random fields as a binary classifier to determine the probability of removing a Chinese character to form an abbreviation. They rerank the resulting abbreviations by using a length prior modeled from their training data and co-occurrence of the original word and generated abbreviation using web search.

A commonly accepted research methodology is

treating normalisation as a noisy channel problem. (Choudhury et al., 2010) explain a supervised noisy channel framework using HMMs for SMS normalisation. This work was then extended by (Cook and Stevenson, 2009) to create an unsupervised noisy channel approach using probabilistic models for common abbreviation types and choosing the English word with the highest probability after combining the models. (Beaufort et al., 2010) combine a noisy channel model with a rule-based finite-state transducer and got reasonable results on French SMS, but did not test their method on English text. (Xue et al., 2011) adopted the noisy-channel framework for normalisation of microtext and proved that it is an effective method for performing normalisation.

(Vyas et al., 2014) worked on POS tagging for Hindi-English data . For Hindi normalisation, they used the system built by (Gella et al., 2013) but they did not normalise English text as they used the (Owoputi et al., 2013) Twitter POS Tagger in the next step, which does not require normalised data.

4 Data Preparation

For performing Language identification for Konkani-English language we don`t have sufficient annotated datasets and other resources. As a part of this research work we developed the following resources.

we collected data from Facebook public pages of Konkani group. All these pages are very popular with 9800 likes. A total of 4983 posts were scrapped from Konkani group pages, which were published between 6 may 2014 to 28th September 2016 and preference was given to posts having a long thread of posts. The corpus thus generated has 4,983 posts and 1,13,578 words. Due to the usage of Facebook as the underlying crowd sourcing engine, the data generated was highly conversational and had reasonable amount of social-media lingo.

Facebook posts were broken down into sentences using sentence Tokenize and 5088 of those code-mixed sentences were randomly selected for manual annotation. The data was semi-automatically cleaned and formatted, removing user names for privacy. The names of public figures in the posts were retained.

4.1 Data Statistics

The size of the original data was 34036 sentences of facebook post. 5088 (14.94%) of those code-mixed sentences were randomly selected, containing a total of 60,118 tokens. Table 1 show the distribution of the dataset at token level respectively. Of these tokens, 34,118 (56.75%) are Konkani words which are in Roman script, 17,764 (29.54%) are English words. 8,236 (13.69%) are acronym, slag words, hindi words etc which are marked as ‘ Rest ’ .

Language	All Sentences
Konkani	34,118 (56.75%)
English	17,764 (29.54%)
Rest	8236 (13.69%)
Total	**60,118**

Table 1: Data distribution at token level

4.2 Dataset examples

1. ***Interviewer:*** *Tuka British Accent’n ulopak kalta? thn plz speeak.. pleeeaase! thn i cn say ur genuis*
   Interviewer: *Bare ulon dakhoi*

The dataset is comprised of sentences similar to *Example 1* . *Example 1* shows Code-Mixing as some English words are embedded in a Konkani utterance. Spelling variations (ur - your), ambiguous words (To - So in Konkani or To in English) and non-adherence to a formal grammar (out of place ellipsis.., no or misplaced punctuation) are some of the challenges evident in analyzing the examples above.

5 Annotation Guidelines

The creation of this linguistic resource involved Language identification layer. In the following paragraphs, we describe the annotation guidelines for these tasks in detail. Manual Annotation was done on the following layer:

5.1 Language Identification

Similar to (Barman et al., 2014), we will be treating language identification as a three class (‘ kn’, ‘ en’, ‘ rest’) classification problem. Every word was given a tag out of three - en, kn and rest to mark its language. Words that a bilingual speaker could identify as belonging to either Konkani or English were marked as ‘kn’ or ‘en’, respectively.

The label ‘rest’ was given to symbols, emoticons, punctuation, named entities, acronyms, foreign words.

The label ‘rest’ was created in order to accommodate words that did not strictly belong to any language, described below:

1 Symbols, emoticons and punctuation

2 **Named Entities** : Named Entities are language independent in most cases. For instance, ‘Jack’ would be represented by equivalent characters in Konkani and English.

3 **Acronyms**: This includes SMS acronyms such as ‘LOL’, and established contractions such as ‘USA’. Acronyms are very interesting linguistic units, and play an important role in social media text. They represent not just entities but also phrases and reactions. We wanted to keep their analysis separate from the rest of the language; and hence they were categorised as ‘rest’ in our dataset.

4 **foreign words** : A word borrowed from a language except Konkani and English has been treated as ‘rest’ as well. This does not include commonly borrowed Hindi words in Konkani; they are treated as a part of Konkani language.

5 **Sub-lexical code-mixing** : Any word with word-level code-mixing has been classified as ‘rest’, since it represents a more complex morphology.

5.2 Normalisation

Words with language tag ‘ kn’ in Roman script were labeled with their standard form in the native script of Konkani Devanagari, i.e. a back-transliteration will be perform. Words with language tag ‘ en’ were labeled with their standard spelling. Words with language tag ‘ rest’ were kept as they are.
Following are some case-specific guidelines.

1 In case a token consists of two words (due to an error in typing the space), the tokens are separated and written in their original script. For instance, ‘ whatis’ would be normalised to ‘ what is’, with the language ID as English.

2 In cases where multiple spellings of a word are considered acceptable, we have allowed both spelling variations to exist as the standard spellings. For instance, in ‘ color’ and ‘ colour’, ‘ dialogue’ and ‘ dialog’, both spellings are valid.

3 Contractions such as ‘ don’t’ and ‘ who’s’ have been left undisturbed. The dataset thus contains both variations - ‘ don’t’ and ‘ do not’, depending on the original chat text.

4 Konkani has evolved through the past decades, and often we see variations in spelling of a single word. We observed the variation patterns and choose the standard spellings.

The overall annotation process was not a very ambiguous task and annotation instruction was straight-forward. Three Konkani-English bilingual speaker annotated whole dataset. They were not Linguist! Two other annotators reviewed and cleaned it. To measure inter-annotator agreement, another annotator read the guidelines and annotated 125 sentences from scratch. The inter-annotator agreement calculated by third annotator using Cohens Kappa (Cohen, 1960) came out to be 0.78 for language identification.

6 Tools and Resources

We have used the following resources and tools in our experiment. Our English dictionaries Statistics are those described in Table 2 (BNC[1] , LexNorm-List [2]) and the training set words.

Resources are :-

1. **British National Corpus (BNC):** We compile a word frequency list from the BNC (Aston and Burnard, 1998).

2. **Lexical Normalization List (LexNorm-List):** Lexical normalization dataset released by (Han and Baldwin, 2011) which consists of 41118 pair of unnormalized and normalized words / phrases.

3. **slang words:** Dictionary of Internet slang words was extracted from `http://www.noslang.com`.

4. **Transliteration pairs:** We developed wordlists for English - Konkani language

[1] `http://www.natcorp.ox.ac.uk/`

[2] We use a lexical normalization dictionary created by Han et al. (2012)

pairs using ILCI[3]. The wordlists contained few overlapping words.

source Language	Words
BNC	7,60,089
LEXNORM	41,118
Konkani Dictionary[4]	15,195

Table 2: Statistics of English and Konkani Dictionary

7 Experiments and Results

7.1 Language Identification

While language identification at the document level is a well-established task (Myers-Scotton, 1982), identifying language in social media posts has certain challenges associated to it. Spelling errors, phonetic typing, use of transliterated alphabets and abbreviations combined with code-mixing make this problem interesting. Similar to (Barman et al., 2014), we performed experiments treating language identification as a three class (‘ kn’, ‘ en’, ‘ rest’) classification problem.

For the initial experimentation, the tokenized corpus of 5088 sentences is randomly shuffled and the first 80% of dataset included in the training and the remaining 20% for testing. Since our training data is entirely labelled at the word-level by human annotators, we address the word-level language identification task in a fully supervised way. Manual annotation is a laborious process.

We address the problem of Language Identification in two different ways:

1. A simple heuristic-based approach which uses a combination of our dictionaries to classify the language of a word.

2. Word-level Language Identification using supervised machine learning with SVMs[6], Random forest and sequence labelling using CRFs[7], employing contextual information.

7.1.1 Dictionary-Based Detection

A simple rule-based method is applied to predict language of a word <w1 w2 w3 w4....wn>.
A token is considered as (‘en’,‘kn’,‘rest’) class to mark its language. if any of the following conditions satisfies.

Steps are as follows.

1. Tokenise given input query.

2. Match the word in English dictionary. so; **Wen** <w1 w2 w4 ....wn >Set of words which are found in English dictionary, found words were tags as en (English word).

3. Remaining words were compared with Konkani Dictionary[8] which is described in sections 6, **Wkn** <w2 w3 w6.... wn >, found words were tags as kn (Konkani word).

4. Set of Words **Wrest** <w5 >which remains untag are tag as ‘rest’.

5. take **Wen** set and compared with Konkani Dictionary .

6. if we found any word from **Wen** set in Konkani Dictionary than we remove that word from **Wen** set and tag the word as ‘rest’. so, we get ambiguous words. Other words remaining in set Wen are tagged as en (English words). By this approach we get particular Konkani words and English words and ambiguous words.

Dictionary	Accuracy(%)
BNC + Konkani Dictionary	69.05
LexNorm + Konkani Dictionary	68.76
BNC + LexNorm + Konkani Dictionary	69.85

Table 3: Results of dictionary-based detection

Table 3 shows the results of dictionary-based detection. We try different combinations with the above dictionaries (described in section 5). We find that using a normalized frequency is helpful and that a combination of LexNormList and Konkani-English Transliteration pairs, BNC is suited best for our data. Hence, we consider this as our baseline language identification system

7.1.2 Word-Language Detection using machine learning classifier

Word level language detection from code-mixed text can be defined as a classification problem. SVMs were chosen for the experiment (Joachims,

[3]Indian Language Corpora Initiative corpus
[6]http://scikit-learn.org/stable/
[7]https://taku910.github.io/crfpp/
[8](Konkani-English Transliteration pairs) which is described in section 5.

1998). The reason for choosing SVMs is that it currently is the best performing machine learning technique across multiple domains and for many tasks, including language identification (Baldwin and Lui, 2010). Another possibility would be to treat language detection as sequence labelling tasks (Lafferty et al., 2001); previous work (King and Abney, 2013) has shown that it provides good performance for the language identification task as well. The features used can be broadly grouped as described below:

1. **Capitalization Features**: They capture if letter(s) in a token has been capitalized or not. The reason for using this feature is that in several languages, capital Roman letters are used to denote proper nouns which could correspond to named entities. This feature is meaningful only for languages which make case distinction (e.g., Roman, Greek and Cyrillic scripts).

2. **Contextual Features**: They constitute the current and surrounding tokens and the length of the current token. Code-switching points are context sensitive and depend on various structural restrictions.

3. **Special Character Features**: They capture the existence of special characters and numbers in the token. Tweets contain various entities like hashtags, mentions, links, smileys, etc., which are signaled by #, and other special characters.

4. **Lexicon Features**: These features indicate the existence of a token in lexicons. Common words in a language and named entities can be curated into finite, manageable lexicons and were therefore used for cases where such data was available.

5. **Character n-gram features**: we also used charagter n-grams for n=1 to 5.

We perform experiments with an different classifier for different combination of these features. The features are listed in Table 4. The accuracies with respect to different classifier and Features are shown in Table 5. All possible combinations are considered during experiments. It can be seen from the results that character gram feature provides best results . Whereas for lexical and Word gram and Contextual features provides comparable results.

7.1.2.1 System Accuracy

The approach using CRFs had a greater accuracy, which validated our hypothesis and also proved that context is crucial in this process. The results of this module are shown in Table 5.

7.2 Normalisation

Once the language identification task is complete, there will be a need to convert the noisy non-standard tokens (such as English and Konkani words inconsistently written in many ways using the Roman script) in the text into standard words. To fix this, a normalization module that performs language-specific transformations, yielding the correct spelling for a given word was built. we had used two approach for normalisation.

1)Konkani Transliterator and Normalizer
2) Noisy Channel Framework.

These are further explained in Section 7.2.1 and 7.2.2

7.2.1 Konkani Transliterator and Normalizer (Normalizer):

We use CMU Part of Speech tagger [9] on English words which reported an accuracy of 65.39% , it normalizes English words as a primary step. We used Python-Irtrans [10] developed by IIIT-Hyderabad for transliteration of Konkani words from Roman to Devanagari. We ran the konkani words on transliteration system in order to normalize it. This tool is used to convert roman into Konkani script i.e Python-Irtrans which reported an accuracy of 60.09%.

7.2.2 Noisy Channel Framework:

For transliterating the detected Romanized Konkani words and for noisy English words, we built A Two Layer Normalizer was built for both Konkani and English.

1. **Compression**

2. **Normalizer**

The message is processed using the following techniques described in following sections.

1. Compression: In Social Media platform, while chatting, users most of the time express their emotions/mood by stressing over a few characters

[9] http://www.cs.cmu.edu/ark/
[10] https://github.com/irshadbhat/indic-trans

ID	Feature Description	Type
Capitalization Features		
CAP1	Is first letter capitalized?	True/False
CAP2	Is any character capitalized?	True/False
CAP3	Are all characters capitalized?	True/False
Contextual Features		
CON1	Current Token	String
CON2	Previous 3 and next 3 tokens	String
CON3	Word length	String
Special Character Features		
CHR0	Is English alphabet word?	True/False
CHR1	Contains @ in locations 2-end	True/False
CHR2	Contains # in locations 2-end	True/False
CHR3	Contains ' in locations 2-end	True/False
CHR4	Contains / in locations 2-end	True/False
CHR5	Contains number in locations 2-end	True/False
CHR6	Contains punctuation in locations 2-end	True/False
CHR7	Starts with @	True/False
CHR8	Starts with #	True/False
CHR9	Starts with '	True/False
CHR10	Starts with /	True/False
CHR11	Starts with number	True/False
CHR12	Starts with punctuation	True/False
CHR13	Token is a number?	True/False
CHR14	Token is a punctuation?	True/False
CHR15	Token contains a number?	True/False
Lexical Features		
LEX1	Is present in English dictionary?	True/False
LEX2	Is Acronym	True/False
LEX3	Is NE?	True/False
Character n-gram Features		
CNG0	Uni-gram, bigram,trigram	vector
Word n-gram Feature		
WNG0	Uni-gram, bigram,trigram	Probability

Table 4: A description of features used.

in the word. *For example*, usage of words are *thanksss, sryy, byeeee, wowww, gooooood* which corresponds the person being obliged, needy, apologetic, emotional, amazed, etc.

As we know, it is unlikely for an English word to contain the same character consecutively for three or more times hence, we compress all the repeated windows of character length greater than two, to two characters.

Each window now contains two characters of the same alphabet in cases of repetition. Let n be the number of windows, obtained from the previous step. Since average length of English word (Mayzner and Tresselt, 1965) is approximately 4.9, we apply brute force search over 2n possibilities to select a valid dictionary word. If none of the combinations form a valid English word, the compressed form is used for normalization.

Table 6 contains sanitized sample output from our compression module for further processing.

2. Normalizer: Text Message Normalization is the process of translating ad-hoc abbreviations, typographical errors, phonetic substitution and ungrammatical structures used in text messaging (SMS and Chatting) to plain English. Use of such language (often referred as Chatting Language)

Features	System		
	SVM	RF	CRF
CON*	0.86	0.89	0.89
CHR*	0.87	0.86	0.87
CAP*	0.887	0.88	0.877
LEX*	0.89	0.89	0.88
CNG*	0.93	0.92	0.94
WNG*	0.89	0.88	0.89
ALL	0.898	0.90	0.97

Table 5: System word-level accuracies (in %) for language detection from code-mixed text on the test datasets.'*" is used to indicate a group of features. Refer Table. 4 for the feature Ids.

Input Sentence	Output Sentence
I am so gooood	I am so good !
tuuu kaashe asa...	tu kashe asa...

Table 6: Sample output of Compression module

induces noise which poses additional processing challenges. While dictionary lookup based methods [11] are popular for Normalization, they can not make use of context and domain knowledge. **For example**, *yr* can have multiple translations like *year, your*.

We tackle this by building our normalzation system based on the state-of-the-art Phrase Based Machine Translation System (PB-SMT), that learns normalization patterns from a large number of training examples. We use Moses (Koehn et al., 2007), a statistical machine translation system that allows training of translation models.

PB-SMT is a machine translation model; therefore, we adapted the PB-SMT model to the transliteration task by translating characters rather than words as in character-level translation. For character alignment, we used GIZA++ implementation of the IBM word alignment model. To suit the PB-SMT model to the transliteration task, we do not use the phrase reordering model. The target language model is built on the target side of the parallel data with Kneser-Ney (Kneser and Ney, 1995) smoothing using the IRSTLM tool (Federico et al., 2008). In a bid to simulate syllable level transliteration we also built a Normalization model by breaking the English and Konkani words to chunks of consecutive characters and trained the transliteration system on this chunked data.

Training process requires a Language Model of the target language and a parallel corpora containing aligned un-normalized and normalized word pairs.

For English and Konkani word Normalization, our language model consists of 50,156 English un-normalized and normalized words taken from the web, 15195 Konkani words taken from Indian Language Corpora Initiative (ILCI) Corpus and manually transliterated.

Parallel corpora was used which is described in section 6.

Table 7. presents the obtained results.

7.2.2.1 System Accuracy

The accuracy of this system is shown in Table 7. The accuracy for the Konkani normaliser is higher than that for English.

Languages	Accuracy (%)
English Normalizer	72.81
Konkani Normalizer	77.21

Table 7: Token level Normalization Accuracy

8 Conclusion and Future Work

We have presented an initial study on automatic language identification and text normalisation with Indian language code mixing from social media communication. This is a quite complex language identification task which has to be carried out at the word level, since each message and each single sentence can contain text and words in several languages. The paper has aimed to put the spotlight on the issues that make code-mixed text challenging for language processing. we have focused on the process of creating and annotating a much needed dataset for code-mixed Konkani-English sentences in the social media context, as well as developed language identification and normalisation systems follow supervised machine learning and report final accuracies of 97.01% and 72.81% for English Normalizer , 77.81% for Konkani Normalizer for our dataset, respectively.

In the future, we intend to continue creating more annotated code-mixed social media data. We intend to use this dataset to build tools for code-mixed data like POS taggers, morph analysers, chunkers and parsers. In the future we would also like to evaluate on adding more language classes,

[11] http://www.lingo2word.com

particularly for named entities and acronyms influences the overall accuracy of our system.

References

Guy Aston and Lou Burnard. 1998. *The BNC handbook: exploring the British National Corpus with SARA*. Capstone.

Timothy Baldwin and Marco Lui. 2010. Language identification: The long and the short of the matter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 229–237.

Utsab Barman, Amitava Das, Joachim Wagner, and Jennifer Foster. 2014. Code mixing: A challenge for language identification in the language of social media. *EMNLP 2014* 13.

Richard Beaufort, Sophie Roekhaut, Louise-Amélie Cougnon, and Cédrick Fairon. 2010. A hybrid rule/model-based finite-state framework for normalizing sms messages. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 770–779.

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

Monojit Choudhury, Kalika Bali, Tirthankar Dasgupta, and Anupam Basu. 2010. Resource creation for training and testing of transliteration systems for indian languages. LREC.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement* 20(1):37–46.

Paul Cook and Suzanne Stevenson. 2009. An unsupervised model for text message normalization. In *Proceedings of the workshop on computational approaches to linguistic creativity*. Association for Computational Linguistics, pages 71–78.

Anik Dey and Pascale Fung. 2014. A hindi-english code-switching corpus. In *LREC*. pages 2410–2413.

Marcello Federico, Nicola Bertoldi, and Mauro Cettolo. 2008. Irstlm: an open source toolkit for handling large scale language models. In *Interspeech*. pages 1618–1621.

Spandana Gella, Jatin Sharma, and Kalika Bali. 2013. Query word labeling and back transliteration for indian languages: Shared task system description. *FIRE Working Notes* 3.

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

Taofik Hidayat. 2012. An analysis of code switching used by facebookers (a case study in a social network site). In *Student essay for the study programme-PendidikanBahasaInggris (English Education) at STKIP Siliwangi Bandung*.

Thorsten Joachims. 1998. Making large-scale svm learning practical. Technical report, Technical Report, SFB 475: Komplexitätsreduktion in Multivariaten Datenstrukturen, Universität Dortmund.

Ben King and Steven P Abney. 2013. Labeling the languages of words in mixed-language documents using weakly supervised methods. In *HLT-NAACL*. pages 1110–1119.

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for m-gram language modeling. In *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*. IEEE, volume 1, pages 181–184.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*. Association for Computational Linguistics, pages 177–180.

John Lafferty, Andrew McCallum, Fernando Pereira, et al. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the eighteenth international conference on machine learning, ICML*. volume 1, pages 282–289.

Mark S Mayzner and Margaret Elizabeth Tresselt. 1965. Tables of single-letter and digram frequency counts for various word-length and letter-position combinations. *Psychonomic monograph supplements* .

Paul McNamee. 2005. Language identification: a solved problem suitable for undergraduate instruction. *Journal of Computing Sciences in Colleges* 20(3):94–101.

Carol Myers-Scotton. 1982. Duelling languages: Grammatical structure in codeswitching. In *Oxford University Press.*.

Dong-Phuong Nguyen and A Seza Dogruoz. 2013. Word level language identification in online multilingual communication. Association for Computational Linguistics.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. Association for Computational Linguistics.

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

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

Stephen Tratz, Douglas Briesch, Jamal Laoudi, and Clare Voss. 2013. Tweet conversation annotation tool with a focus on an arabic dialect, moroccan darija. *LAW VII & ID* 135.

Yogarshi Vyas, Spandana Gella, Jatin Sharma, Kalika Bali, and Monojit Choudhury. 2014. Pos tagging of english-hindi code-mixed social media content. In *EMNLP*. volume 14, pages 974–979.

Min Wang, Chen Yang, and Chenxi Cheng. 2009. The contributions of phonology, orthography, and morphology in chinese–english biliteracy acquisition. *Applied Psycholinguistics* 30(02):291–314.

Kam-Fai Wong and Yunqing Xia. 2008. Normalization of chinese chat language. *Language Resources and Evaluation* 42(2):219–242.

Zhenzhen Xue, Dawei Yin, and Brian D Davison. 2011. Normalizing microtext. In *Workshops at the Twenty-Fifth AAAI Conference on Artificial Intelligence*.

Hiroshi Yamaguchi and Kumiko Tanaka-Ishii. 2012. Text segmentation by language using minimum description length. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*. Association for Computational Linguistics, pages 969–978.