Adaptive Voting in Multiple Classifier Systems for Word Level Language Identification

Soumik Mandal
Jadavpur University, India
mandal.soumik@gmail.com

Somnath Banerjee
Jadavpur University, India
sb.cse.ju@gmail.com

Sudip Kumar Naskar
Jadavpur University, India
sudip.naskar@cse.jdvu.ac.in

Paolo Rosso
UPV, Spain
prosso@dsic.upv.es

Sivaji Bandyopadhyay
Jadavpur University, India
sivaji_cse_ju@yahoo.com

ABSTRACT

In social media communication, code switching has become quite a common phenomenon especially for multilingual speakers. Automatic language identification becomes both a necessary and challenging task in such an environment. In this work, we describe a CRF based system with voting approach for code-mixed query word labeling at word-level as part of our participation in the shared task on Mixed Script Information Retrieval at Forum for Information Retrieval Evaluation (FIRE) in 2015. Our method uses character n-gram, simple lexical features and special character features, and therefore, can easily be replicated across languages. The performance of the system was evaluated against the test sets provided by the FIRE 2015 shared task on mixed script information retrieval. Experimental results show encouraging performance across the language pairs.

CCS Concepts

•**Computer systems organization** → **Embedded systems;** *Redundancy;* Robotics; •**Networks** → Network reliability;

Keywords

ACM proceedings; LaTeX; text tagging

1. INTRODUCTION

Though South and South East Asian languages have their own indigenous scripts, these languages are mostly written using Roman script in the social media such as tweets, blogs, etc., due to various socio-cultural and technological reasons. The use of Roman script for such languages presents serious challenges to understanding, search and language identification. Abundant use of Roman script on the Web not only for documents as well as for user queries to search the documents needs to be addressed. Although language identification at document level is a well-studied natural language problem [4], the different aspects of this problem of labeling the language of individual words within a multilingual document were addressed in [10], [8]. They proposed language identification at the word level in mixed language documents instead of sentence level identification. Recently, language identification problem in code-mixed data has been revisited in the First Workshop on Computational Approaches to Code Switching in EMNLP-2014. It was mentioned that fine-grained language identification from more than one language is still very challenging and error prone when the spans of text are smaller. Unsupervised and supervised approaches were investigated for the detection of four language pairs, Spanish-English, Modern Standard Arabic and Arabic dialects, Chinese-English and Nepalese-English, at the word level in code-switching data. The results of the task revealed that language identification in code-switching is still far from solved and warrants further natural language processing research. Shared tasks on language identification have been organized in FIRE since 2013 and various attempts [6],[7],[1],[3],[5],[9] were carried out to address language identification task.

2. TASK DEFINITION

A query or utterance q : < w1 w2 w3 ... wn > is written in Roman script. The words or tokens, w1,w2,w3 etc., could be standard English (en) words or transliterated from any of the eight Indian languages, namely Bengali (bn), Hindi (hi), Gujrati (gu), Kannada (kn), Malayalam (ml), Marathi (mr), Tamil (ta), Telugu (te) under consideration in this subtask. The main objective of this task is to perform word-level language identification (WLL), i.e. to label each token with single tag belongs to one of the five categories shown in Table 1. Though some of the categories have also finer subcategories, the identification of such subcategories is not mandatory.

3. DATA

This section describes the training and test dataset that were provided to the task participants by the task organizers. The training dataset was provided in the form of set of sentences and respective tags for each token of the sentences. The training dataset consists of 2908 utterances, whereas the test dataset contains 792. Apart from the dataset provided by the task organizers we did not use any external dataset or resources to either train or fine-tune our system.

An empirical study on the development data reveals the following facts: a) the average length of all the tokens is greater than 5 and b) majority of the tokens belong to the English language.

4. SYSTEM DESCRIPTION

Our word identification process involves three steps- At first we have independently applied multiple classifiers which

Table 1: Tagset of different categories

Category	Possible Tags	Subcategory
Language	en, bn, gu, hi, kn, ml, mr, ta or te	
Named Entity	NE	Person (NE_P), Location (NE_L), Organization (NE_O), Abbreviation (NE_PA, NE_LA), Inflectional form (NE-Ls, where Ls is the language of the suffix) or none of the above (NE_X)
Mixed	MIX	Mix_Lr_Ls: Lr and Ls are root and suffix language respectively.
Punctuation	X	
Others	O	

have been developed using CRF. Then voting approach has been employed over the outputs of the classifiers which are applied in first step. Finally, we have employed a classifier which deals with NE and MIX tags. Also we have tackled the conflict situations those come up through voting (discussed in section 4.2).

4.1 WLL classification Features

We have developed in total nine classifiers. Eight different IL(N) where N=BN,GU,HI,KN,ML,MR, TA,TE classification models were built for eight Indian languages (ILs), namely BN-classifier, GU-classifier, HI-classifier, KN-classifier, ML-Classifier, MR-classifier, TA-classifier and TE-classifier. While training a IL(N) classifier, tokens of the type NE, MIX, Others and all other ILs were assigned R tag.The output of IL(N) classifier could be one of the four- i) N ii) X (for punctuation) iii) en (for English) and iv) R (for any IL except N, NE, MIX and Others). Apart from eight IL(N) classifiers, we have trained another classifier (namely ALL-classifier) using all the existing tags in the supplied training dataset. The ALL-classifier has dealt with the NE, MIX and Others tokens as well as served as tie breaker (discussed in section 4.2).

In this work, Conditional Random Field (CRF) has been employed to build all of the classifier models. We used CRF++ toolkit1 which is a simple, customizable, and open source implementation of CRF. All of these nine classifiers used the same set of features listed below in the following subsections.

4.1.1 Character n-grams

Recent studies [6],[8] had shown that the character n-gram feature can produce reasonable success in language identification problem. Therefore, following them, we also used character n-grams as features in our system. Keeping the average token length of training set in mind, we decided to consider up to 6-grams. Other than the n-grams, the entire token was also considered as a feature in the system. However due to fixed length vector constraint, we decided to consider on the maximum length of a token to be 10 for generating the character n-grams. So, if the length of a particular token is greater than 10 then only first 10 characters of that token were used to generate the n-grams and the rest of the characters were ignored. Thus irrespective of the token length, the system always generates a total of 46 n-grams i.e. the token itself, 10 unigrams, 9 bigrams, 8 trigrams, 7 four-grams, 6 five-grams and 5 six-grams.

4.1.2 Symbol character

A token might either start with some symbol, e.g. #aapstorm, @timesnow or it may contain such symbols within, e.g. a***a, bari-r etc. Sometimes the entire word is built up of a symbol, e.g. ", ?.

$$has\_symbol(token) = \begin{cases} 1 & \text{if } token \text{ has symbols} \\ 0 & \text{otherwise} \end{cases}$$

4.1.3 Links

This feature was used as a binary feature. If a token is a link, i.e. if it starts with "http://", "https://" or "www." then the value is set to 1, otherwise it is set to 0.

$$is\_link(token) = \begin{cases} 1 & \text{if } token \text{ is a link} \\ 0 & \text{otherwise} \end{cases}$$

4.1.4 Presence of Digit

In case of chat dialogue the use of digit(s) in a word often means different than their traditional use. For example, '$n8$' could mean 'night', '2' could mean 'to' or 'too'. It is also found that most of the cases such words contain numerical digits in single position. Therefore, in our system we have used the presence of single digit in any alphanumeric word as binary feature.

$$has\_digit(token) = \begin{cases} 1 & \text{if } token \text{ has numerical digits} \\ 0 & \text{otherwise} \end{cases}$$

4.1.5 Word suffix

It is an established fact that any language dependent feature increases the accuracy of language identification systems for that particular language. Also recent studies on fixed length suffix feature had been carried out and were successfully used by [2] in the Bangla named entity recognition task. Following these facts, we decided to create a small set of most frequent suffixes for the en words present in the training dataset based on our own automated suffix extractor algorithm. The list of most frequent en suffixes extracted in this method were -ed, -ly, -'s, -'t, -'ll and -'ing and the presence of these suffixes was marked as binary features in the classifiers, i.e.

$$has\_suffix(token) = \begin{cases} 1 & \text{if } token \text{ has the suffix} \\ 0 & \text{otherwise} \end{cases}$$

4.2 Voting Approach

Once the outputs of all the classifiers are gathered, a voting mechanism is applied to decide the final label of each token. The voting approach is based on some rules, which are listed below:

4.2.1 No conflict situation

This case is straight forward, i.e. no conflict between the outputs of all the eight IL(N) classifiers for a single token, meaning all the IL(N) classifiers agree on the tag of that token.

Rule 1: This rule is applicable for only En and X tags. If the output of all the classifiers for a particular token is same and either EN or X, then that particular tag is chosen as the final tag for the given token. For example, the token #aapsweep is labeled as X by all eight classifiers. Thus, the final tag of this token becomes X.

#aapsweep X X X X X X X X ⇒ X

Rule 2: If all the tags are same but other than EN or X, then we consider the output of the ALL-classifier for the said token as the final tag. This phenomenon only occurs when all the eight IL(N) classifiers identified the token as R. For example, in the following example, the token 'saaf' is marked as R by all the eight IL(N) classifiers. Since the label generated by the ALL-classifier for 'saaf' is HI, so the final tag of the token becomes HI.

saaf R R R R R R R ⇒ HI

4.2.2 *Conflict between two tags*

In this scenario, output of all the classifiers for a given token is limited between two tags. Based on the tags involved in such conflicts this situation is further classified into subcategories which are discussed in the following subsections.

Rule 3: If conflict is between R and any other language tag including EN, then the tag other than R marked by the classifier is selected as the final tag of the token. In the following example, the token doctor is marked as either EN or IL by the language classifiers. Therefore, the final tag of doctor is EN.

doctor R R R R R R EN EN ⇒ EN

Rule 4: If the classifiers differ in between two tags other than R, then a voting is counted in support of each of the two tags. Finally the tag with maximum votes is assigned as the final tag for the given token. In the example, the no. of votes in favor of EN tag for the token take is greater than the no. of votes supporting BN.

take BN EN EN EN EN EN EN EN ⇒ EN

4.2.3 *Conflict between three tags*

Rule 5: If the conflict involves a) R, b) EN or X and c) any of the eight Indian Language tags, then we first replace all the R tags with the other Indian Language tag involved in the conflict, thus reducing the conflict between three tags scenario into conflict between two. Finally Rule 4 is applied to decide the final tag. For example;

ore BN EN R R R R R R

⇓

ore BN EN BN BN BN BN BN BN ⇒ BN

Rule 6: If the conflict involves three tags and none of those three are R, then simple majority voting was applied to choose the final tag.

4.2.4 *Conflict between more than three tags*

Rule 7: In case there is disagreement between more than three language classifier for a single token, the final label of that token is decided by the All-classifier. The occurrence of such cases is very rare.

4.3 Handling NE and MIX tags

Since we have not included any feature specifically to handle the NE or MIX tokens, we have depended entirely on the All-classifier to mark the NE and MIX tokens. So, if a token is marked as NE by the All-classifier then the final tag of the token becomes NE, irrespective of the outputs of the eight language classifiers for the same token. The same procedure is applied to mask MIX tokens.

5. RESULT AND ERROR ANALYSIS

Table 2 represents the results obtained by our language identification system in different categories other than Language. As the table depicts, our system has achieved best accuracy of 0.9293 in case of punctuation category, whereas the results for MIX category is too low to report. Out of 24 MIX-tagged token only 2 are correct (precision and recall values of the Mix category are not provided by the organizer). Even, in case of NEs the accuracy is too low at 0.4136 when compared to that of punctuation category; still it is the best score obtained in the NE category among all the teams participated in the subtask as per the task organizers. To be noted is our system has not marked any token as O category.

Table 2: Token level accuracy category-wise

Category	Precision	Recall	F-measure
Punctuation	0.8883	0.9742	0.9293
Named Entity	0.3316	0.5494	0.4136
Mix			

In case of language category maximum accuracy is achieved for en tokens, which is 0.7838. Whereas, the accuracy is pretty low for Gujrati, Malayalam and Kannada languages (shown in Table 3). We have dwelled upon the result and observed that it is due to the lower amount of tokens presence in the development set for these three languages. For example, the number of gu tokens present in the development set is only 890, which is very few when compared to that of en tokens, i.e. 17957.

Table 3: Token level Language Accuracy

Category	Precision	Recall	F-measure
Bengali	0.75	0.8208	0.7838
English	0.9506	0.6147	0.7466
Gujrati	0.1622	0.3704	0.2256
Hindi	0.5	0.8186	0.6208
Kannada	0.2876	0.7713	0.419
Malayalam	0.1991	0.6667	0.3067
Marathi	0.5815	0.7586	0.6584
Tamil	0.7514	0.7757	0.7633
Telugu	0.3473	0.657	0.4544

Overall, our system achieved the accuracy (weighted f-measure) of 0.700373312. Out of 11999 tokens in the testset 8582 tokens were marked correctly. However, as our system didn't consider any contextual information, the accuracy achieved at the utterance level was expectedly very low at 0.128788. Only in 102 occasions all the tokens of an entire utterance was labeled with correct tags. More detailed analysis of the result can be done once the gold standard data is shared by the task organizers.

6. CONCLUSIONS AND FUTURE WORKS

In this paper, we have presented a brief overview of our hybrid approach to address the automatic WLL identification problem. We have observed that the voting approach on multiple classifiers output provides better results than use of a single classifier system. For our participation in Query Word Labeling subtask, we have submitted two runs: the first one, i.e. Run1 using the system as described above and the other, i.e. Run2 using only the ALL-classifier without the need of any voting mechanism, and the obtained results confirm that the overall accuracy of Run1 is more than 10% higher when compared to Run2.

As future work, we would like to explore more sophisticated features to handle NE or O tags and better post-processing heuristics for handling MIX tags in the WLL identification task and try to improve the performance of system by using context modelling. We also plan to incorporate more language specific feature in our future work to improve the accuracy of the system.

7. ACKNOWLEDGMENTS

We acknowledge the support of the Department of Electronics and Information Technology (DeitY), Government of India, through the project "CLIA System Phase III".

The research work of the second last author was carried out in the framework of WIQ-EI IRSES (Grant No. 269180) within the FP 7 Marie Curie, DIANA-APPLICATIONS (TIN2012-38603-C02-01) projects and the VLC/CAMPUS Microcluster on Multimodal Interaction in Intelligent Systems.

8. REFERENCES

[1] S. Banerjee, A.Kuila, A. Roy, S. N. P. Rosso, and S. Bandyopadhyay. A hybrid approach for transliterated word-level language identification: Crf with post-processing heuristics. In *FIRE*. ACM Digital Publishing, 2014.

[2] S. Banerjee, S. Naskar, and S. Bandyopadhyay. Bengali named entity recognition using margin infused relaxed algorithm. In *TSD*, pages 125–132. Springer International Publishing, 2014.

[3] U. Barman, J. Wagner, G. Chrupala, and J.Foster. Identification of languages and encodings in a multilingual document. page 127. EMNLP, 2014.

[4] K. R. Beesley. Language identifier: A computer program for automatic natural-language identification of on-line text. pages 47–54. ATA, 1988.

[5] M. Carpuat. Mixed-language and code-switching in the canadian hansard. page 107. EMNLP, 2014.

[6] G. Chittaranjan, Y. Vyas, K. Bali, and M. Choudhury. Word-level language identification using crf: Code-switching shared task report of msr india system. pages 73–79. EMNLP, 2014.

[7] A. Das and B. GambÃd'ck. Code-mixing in social media text:the last language identification frontier? *Traitement Automatique des Langues (TAL): Special Issue on Social Networks and NLP*, 54(3/2013):41–64, 2014.

[8] B. King and S. Abney. Labeling the languages of words in mixed-language documents using weakly supervised methods. pages 1110–1119. NAACL-HLT, 2013.

[9] C. Lignos and M. Marcus. Toward web-scale analysis of codeswitching. In *Annual Meeting of the Linguistic Society of America*, 2013.

[10] A. K. Singh and J. Gorla. Identification of languages and encodings in a multilingual document. In *ACL-SIGWAC's Web As Corpus3*, page 95. Presses univ. de Louvain, 2007.