Using Character Ngrams for Word-Level Language Identification in Trilingual Code-Mixed Data (and Even More)

Yves Bestgen[1]

[1]*Laboratoire d'analyse statistique des textes - Statistical Analysis of Text Laboratory (LAST - SATLab), Université catholique de Louvain, 10 place Cardinal Mercier, Louvain-la-Neuve, 1348, Belgium*

Abstract

This paper presents the solution proposed by the SATLab to classify all the words in short utterances into one of seven categories which include three languages, two of which are closely related Dravidian languages sparsely endowed with linguistic resources (Tulu and Kannada) and a category for tokens that mix several languages. This language-agnostic system uses only character ngrams as features and a classical supervised learning procedure. After optimizing a series of parameters, it ranked first in the CoLI-Tunglish challenge, with a Macro-F1 of 0.813, virtually on a par with the second-place system. Part of its effectiveness comes from taking into account the context in which each word to be categorized is used.

Keywords

Word-level language identification, character ngrams, logistic regression, low-resource languages

1. Introduction

Identifying the language in which a document is written is a classic and important task in natural language processing, because it conditions many other tasks such as translation, summarization or sentiment analysis. Over the last ten years, researchers have turned their attention to more complex issues, such as the distinction of dialectal varieties like those existing between Swiss German dialects in VarDial 2017 [1] or the distinction of several closely related Dravidian languages mixed with English in short YouTube comments written in Roman script in VarDial 2021 [2]. In our multilingual world, code-mixing has become a very frequent phenomenon, particularly in the multitude of posts of all kinds on social networks [3, 4, 5, 6].

The code-mixed task just described (VarDial 2021) already seems particularly complex. Yet it only covers half of the job. Only one of the two languages used in an utterance is the target of discrimination. Thus, an utterance mixing English and Kannada must be identified as "Kannada". A far more complex situation exists: determining the language in which each word of an utterance is written when the utterance in question combines words from several languages

and even "words" that mix two of them, all written in the same script. This situation can be illustrated by the following example (see Table 1) from the CoLI-Tunglish dataset [7].

Table 1
Tokens of a comment to categorize in the CoLI-Tunglish dataset

Token	Category
anna	Kannada
every	English
weekg	Mixed
new	English
contentn	Mixed
padva	Tulu
pandu	Tulu
getondar	Tulu

This type of utterance is relatively frequent in multilingual situations where speakers of a regional language frequently know several other languages, as in the case of the Tuluvas in the Southern part of Karnataka in India [8].

When this task has to be carried out for languages well endowed with linguistic resources, especially electronic dictionaries and very large corpora, such as English, German or Italian, it seems fairly straightforward. This is no longer the case when some of them are low-resource languages, such as Kannada and especially Tulu, respectively the official language and a regional language of Karnataka [8]. For this reason, Hagde et al. [7] have proposed a shared task within the framework of the Forum for Information Retrieval Evaluation (FIRE 2023), asking users to classify all the words in short utterances into one of seven proposed categories. These categories include three languages, two of which are sparsely endowed with linguistic resources, but also a category for tokens that mix several languages, hence the "and even more" in the title of this paper.

To tackle this type of challenge, the SATLab has developed a language-agnostic system that uses only character ngrams as features, and therefore no other linguistic resources. The character ngrams were provided to a classical supervised learning procedure such as an SVM or a gradient boosting decision tree. The aim of the present study is to determine whether this approach is also effective for the CoLI-Tunglish task, which is even more complex than those proposed in the past. The remainder of this summary presents the task and the two systems proposed by the SATLab. The results obtained are then described.

2. Materials and Challenge Rules

The data for this challenge comes from the CoLI-Tunglish dataset [8], which is made up of short YouTube comments in Roman script. These comments contain Tulu, Kannada, and English words, but also personal names and place names, as well as "Mixed-language" and "Other" cases. In all, there are seven categories to distinguish. Table 2 shows the distribution of these

categories in the whole material provided by organizers to develop a system, which included a training and a validation part.

Table 2
Frequency of the seven categories in the learning and development set

Category	Frequency	%
Tulu	10268	47.39
English	6400	29.54
Kannada	2238	10.33
Name	1224	5.65
Other	639	2.95
Mixed	478	2.21
Location	421	1.94
Total	21668	100.00

The imbalance between the frequencies of certain categories is immediately apparent, since the Tulu category is 24 times more frequent than the Location category. This imbalance led the organizers to choose the Macro-F1 to evaluate the efficiency of the systems, an index that gives equal weight to all categories ([9]).

As explained above, words are included in comments, so comments constitute a higher hierarchical level structuring the data. This second level is used in one of the two systems proposed in the next section. The complete training set contains 3,629 comments ranging from 2 to 21 tokens. Almost 95% of the comments are between 3 and 12 tokens long. While 15% of these comments contain words from only one category, 29% include words from three different categories, 9% from four and 1.60% from five.

The test material consisted of 10,505 words, or approximately one third of the complete CoLI-Tunglish dataset. Participating teams were allowed to use any additional linguistic resource, a possibility that the SATLab did not use. They could submit a maximum of three solutions, and were completely unaware of their performance until the end of the challenge.

3. The Two Developed Systems

3.1. Basic System

The basic system, used for Run 1, is derived from the one that won first place at VarDial 2021. The features used are composed exclusively of character ngrams. The major advantage of these is that they enable the development of a system that can be applied without any modification to any writing script or even combination of scripts and to any language, including those that do not explicitly signal separations between words.

When extracting the features of each word, several parameters were set on the basis of a cross-validation procedure using four folds in which all words of an instance are in the same fold:

- The length of the character ngrams, which ranged from 1 to 5.
- The frequency threshold a feature must reach to be used, which was set at 2.
- The weighting scheme applied to the frequency of each feature of an instance (i.e., binary, logarithmic...). In the present case, a sublinear weighting TfIdf was used.
- The weighting scheme applied to the features of an instance, i.e. L2 normalization.

Inferences were performed by a LIBLinear L2-regularized logistic regression model (dual, -s 7) for classification [10]. Three parameters of this procedure were set by means of the cross-validation procedure :

- The regularization parameter C, which was set at 12.
- The -wi options for adjusting the parameter C of different categories, which was set at 4 for "Mixed" and at 3 for "Other", and let to 1 for the five other categories. Table 2 shows that the two overweighted categories are not the rarest, contrary to what might have been expected. These two categories were chosen because they were those that the system predicted too infrequently, and because the use of these weights significantly improved the system's Macro-F1.
- The bias parameter (-B), which shifts the separating hyperplane from the origin, which was set to 1.

3.2. Context-Sensitive System

The basic system extracts the character ngrams of each word independently of all the other words making up the comment in question. This approach, which treats each word in isolation, seems justified, since the word following or preceding the target may a priori belong to any other category. The consequence, however, is that information from a higher hierarchical level, the comment, is not taken into account. To compensate at least partially for this limitation, a second system has been designed. It is also based on a LIBLinear L2-regularized logistic regression model (dual, -s 7) for classification, but takes as input not the character grams, but the output of the basic system. More precisely, for each word, the features are the probability of belonging to each of the seven categories computed by the base system and those of the two neighbors to the left and to the right (if any) in the corresponding comment. The LIBLinear parameters were identical to those used for the base system except that C was set to 1.

4. Results

4.1. Cross-Validation Performance of Both Systems

Table 3 shows the performance of the systems in 4-fold cross-validation using the following five indices: accuracy, weighted-averaged F1 (WA-F1), Macro-F1, Mean Recall and Mean Precision. As a reminder, the official challenge measure is Macro-F1.

This table shows that the basic system is relatively efficient, achieving a Macro-F1 of 0.767 for a seven-category problem. The context-sensitive system is slightly more efficient than the basic system, but the difference is only 0.016 of Macro-F1. Precision is also significantly higher than recall.

Table 3

System performance in 4-fold cross-validation

System	Accuracy	WA-F1	Macro-F1	Recall	Precision
Base	86.80	0.864	0.767	0.730	0.817
Contextual	87.60	0.873	0.783	0.746	0.830

4.2. Challenge Results

Table 4 shows the results of the best run of five teams who took part in the challenge, as provided by the organizers. The SATLab's context-sensitive system came in first, but only by a small margin, since its lead over the BFCAI team was only 0.001 of Macro-F1. The third team is also quite close, with a gap of only 0.014. The best SATLab performance was obtained using the context-sensitive system. The basic model obtained a Macro-F1 of 0.80 and was thus outperformed by the BFCAI system.

It is noteworthy that the Macro-F1 on test data is significantly higher than that obtained in cross-validation (Table 3). If the separation into training material versus test material was carried out in a completely random way, this suggests that having more training data is quite useful, since all the training material is used for the test phase, whereas only 75% of it is used in cross-validation.

Table 4

Official results of the CoLI-Tunglish shared task

Rank	Team Name	Precision	Recall	Macro-F1
1	SATLAB	0.851	0.783	0.813
2	BFCAI	0.859	0.777	0.812
3	Poorvi	0.821	0.781	0.799
4	MUCS	0.807	0.743	0.770
5	IRLab@IITBHU	0.740	0.571	0.602

5. Conclusion

The aim of the CoLI-Tunglish shared task is to develop automatic systems for word-level language identification in code-mixed Tulu, Kannada and English short YouTube comments in Roman script. The systems proposed by the SATLab are a slightly modified version of those that had achieved excellent results in several VarDial and HASOC challenges, whose common feature was to deal with low-resources languages. The first system is based solely on character ngrams, while the second also takes into account contextual information from the commentary.

The contextual system came first in the challenge, but the difference with the second-place system (BFCAI) was very small. Clearly, this difference is insufficient to favor one system over

the other. The criterion must therefore be the complexity of the systems [11]. The SATLab system is clearly a very simple one, requiring few computational resources and no linguistic resources apart from the learning material. When the BFCAI team's report is available, it will be possible to determine how complex their system is. If the SATLab system is much simpler, it would be natural to give it preference. If this is not the case, it would be preferable to consider the two systems as a tie.

The major advantage of the systems proposed by the SATLab is that they are based solely on character ngrams. They can therefore be deployed for any combination of languages for which learning material is available. The main development that can be envisaged is better context awareness. But how this could be possible is not easy to determine unless it can be shown that code-switching at least partially obeys some kind of syntactic rules, which is far from obvious. It seems far more likely that code-switching results from differences in the accessibility of concepts in the different languages known to the author and in the frequency of usage on social networks. Building a system capable of taking such data into account does seem very challenging.

Acknowledgments

References

[1] 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 (VarDial), Valencia, Spain, 2017, pp. 115–123.

[2] Y. Bestgen, Optimizing a supervised classifier for a difficult language identification problem., in: Proceedings of the Eigth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial), 2021, pp. 96–101.

[3] K. Bali, J. Sharma, M. Choudhury, Y. Vyas, "I am borrowing ya mixing ?" an analysis of English-Hindi code mixing in Facebook, in: Proceedings of the First Workshop on Computational Approaches to Code Switching, Association for Computational Linguistics, Doha, Qatar, 2014, pp. 116–126. URL: https://aclanthology.org/W14-3914. doi:10.3115/v1/W14-3914.

[4] A. Das, B. Gambäck, Code-mixing in social media text, Traitement Automatique des Langues 54 (2013) 41–64. URL: https://aclanthology.org/2013.tal-3.3.

[5] S. Thara, P. Poornachandran, Code-mixing: A brief survey, in: 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2018, pp. 2382–2388. doi:10.1109/ICACCI.2018.8554413.

[6] F. Balouchzahi, S. Butt, A. Hegde, N. Ashraf, H. Shashirekha, G. Sidorov, A. Gelbukh, Overview of coli-kanglish: Word level language identification in code-mixed kannada-english texts at icon 2022, in: Proceedings of the 19th International Conference on Natural

Language Processing (ICON): Shared Task on Word Level Language Identification in Code-mixed Kannada-English Texts, 2022, pp. 38–45.

[7] A. Hagde, F. Balouchzahi, S. Coelho, S. Hosahalli Lakshmaiah, H. A Nayel, S. Butt, Overview of coli-tunglish: Word-level language identification in code-mixed tulu texts at fire 2023, in: Forum for Information Retrieval Evaluation FIRE - 2023, 2023.

[8] A. Hegde, M. D. Anusha, S. Coelho, H. L. Shashirekha, B. R. Chakravarthi, Corpus Creation for Sentiment Analysis in Code-Mixed Tulu Text, in: Proceedings of the 1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages, 2022, pp. 33–40.

[9] J. Opitz, S. Burst, Macro F1 and macro F1, 2021. `arXiv:1911.03347`.

[10] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, C.-J. Lin, LIBLINEAR: A library for large linear classification, Journal of Machine Learning Research 9 (2008) 1871–1874.

[11] J. Dodge, S. Gururangan, D. Card, R. Schwartz, N. A. Smith, Show your work: Improved reporting of experimental results, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 2185–2194. URL: https://www.aclweb.org/anthology/D19-1224. doi:`10.18653/v1/D19-1224`.