Classifying 26 Brazilian Indigenous Languages with N-grams Characters Features

Anonymous ACL-IJCNLP submission

Abstract

Identifying a language is still a very important topic in natural language processing. This is especially true for languages with oral tradition, an aspect that usually lead to a small amount of texts available. In that context, lack of documentation and study of such endangered indigenous dialects lead to a high risk of extinction of such languages by the end of this century. Therefore, this paper is the first to propose a benchmark evaluation between different models aiming to categorise 26 Brazilian indigenous languages with oral tradition automatically. Besides that, it is proposed a simpler character n-gram feature extraction for language classification.

1 Introduction

Language categorisation (LC) has been an important research topic for many years. It can be used to separate documents according to the languages used, an important step to subsequent Natural Language Processing (NLP) tasks, such as search engines and automatic Machine Translation (MT) (Jauhiainen et al., 2019).

Despite the fact that LC this is well established for languages spoken by a larger part of the population, indigenous languages have not received much attention, with very few studies focused in such idiomatic groups. (Drude et al., 2007; Moore et al., 2008) One example of of those languages is the set of Brazilian indigenous languages. According to Drude et al. there are around 150 to 180 indigenous languages in the Brazilian territory depending on the criteria which were used to classify them (Drude et al., 2007). The authors also alert that all of them suffer from the risk of extinction by the end of this century. That is why linguistic documentation plays a significant role on preserving those languages (Drude et al., 2007). Therefore, automatic language identification might improve the process of categorising those languages since more data could be collected in a short amount of time.

This paper proposes a model to perform automatic classification of 26 Brazilian indigenous languages. In addition to that, it also presents a new language classification model based on a ranking of character n-grams. The evaluated methods are focused on the scenario in which only a small amount of text is available to train the models, commonly referred as few-show learning (FSL).

This work's main contributions are:

- A proposal of a new classification algorithm based on n-grams language profile.
- The first study to perform classification of 26 Brazilian Native Languages.
- A benchmark evaluation between the proposed model, TextCat language identification algorithm, Linear Support Vector Classifier (SVC) and Multinomial Naïve Bayes (NB) presented in the paper.

2 Related Works

Machine Learning and Deep Learning play a significant role in language identification. Some techniques allow ML algorithms to improve discrimination of similar languages (Çöltekin and Rama, 2016). Gomez et.al proposed a method that used a simple but effective text pre-processing techniques such as truncating of the words and padding (Çöltekin and Rama, 2016). They used term frequency (TF) and Inverse Term Frequency (IDF) as features. In the method proposed by Gomez et al., the results of ML models outperforms baseline deep learning methods. In summary, ML algorithms are straightforward to train and test, and can achieve good results both classifying single and multi-language documents

Another popular LI approach is based on language profiles. One example is TextCat, in which each language has a specific profile based on the n-grams term frequency of pre-defined word categories (Cavnar et al., 1994). TextCat achieved 99.8% accuracy in the classification of newsgroup articles from different language sources. Also, it has the advantage of be off-the-shelf classifier, with ease of use for end users (Jauhiainen et al., 2019).

3 Materials

The experiments in this work consider a corpora composed of verses from the New Testament translation of 26 Brazilian indigenous languages as well as Portuguese translations (Angelo, 2016). Standard pre-processing steps were applied, such as the removal of HTML tags, punctuation symbols and cross references. A total of 1000 verses were randomly chosen and split into train and test documents. Table 1 presents a sample of the same verse in all 26 languages considered in this study.

Langauge	Matthew Chapter 1, Verse 1
Apalai	Aparão mokyro Izake zumy. Izake mokyro Jako zu...
apinaye	• N Apraãw kra na pre kêp Ijak. • N Ijak kra...
Apurina	Ininiã ia atoko itxa: • Kitxakapirĩka Apraão, ...
Bakairi	Saguhoem kuru ise Jesus idamudo kâengatuly, Ab...
Guajajara	Heta Àmàrààw tayr Izak her mae izupe ae. Iz...
Guarani	Abraão ray ma Isaque, Isaque ray ma Jacó, Ja...
Kadiweu	Abraão jiijaa eliodi Isaque, Isaque jiijaa...
Kagwahiva	Ymyahũ Abraãova'ea Isaqueva'ea po'ria hako. Is...
Kaigang	Abraão v Isaque han. K Isaque v Jacó han. K...
Kaiwa	Yma ete vaekwe oiko vaekwe Abraão amyrĩ. Ta...
Karaja	Ibutumy ilabiebohonimy arelyykre. Juhuu tybyni...
Kayabi	Abraão ga Isaki ga ruwa. Isaki ga Jako ga ...
Kayapo	Ingêt Abraão ne Idjak dji. Idjak dji nhym arm...
macushi	Pena Abraão wanîpî. Mîîkîrî wanîpî Isaque yu...
Maxakali	'Ãmanãm te 'Iyak mũg tak, ha 'Iyak te Yako mũg...
munduruku	Isaque ebay Abraão osunuy. Jacó ebay Isaque o'...
Nadeb	Abaraãm taah, Isak. Isak taah, Jakóh. Jakóh ...
Nambikuara	Nxa²ha¹te¹ A³bra²ãu²ah³lai²na² sa²kx...
Portuguese	Abraão gerou Isaque; Isaque gerou Jacó; Jacó g...
parecis	Abraão atyo Isaque kaisani, Isaque atyo Jacó k...
Paumari	Isaque kaabi'i ada Abraão kohana. Jacó kaabi'i...
Rikbaktsa	Tapara Abarão niy. Iwaze Isake ta Abarão tse. ...
Satere-mawe	PIAT EWY Pyno atiatusetpehik teran mesuwe A...
Terena	Eneponeko Âbraum, há'ane neko Izáki. Kene Izák...
Tukano	Abraão Isaque pak niîk niîwĩ. Isaque k'ra J...
Urubu-kaapor	Abrahampa chulinmi Isaac. • Isaacpa chulinñata...
Xavante	Abra'ã hã Izatihi mama. Izati hã Zacoho mama. ...

Table 1: Sample translations of the Bible in indigenous languages.

4 Methods

In this section, we introduce the classification methods considered in the study: TF-IDF+Machine Learning methods and the language profile approaches, TextCat and WordRank.

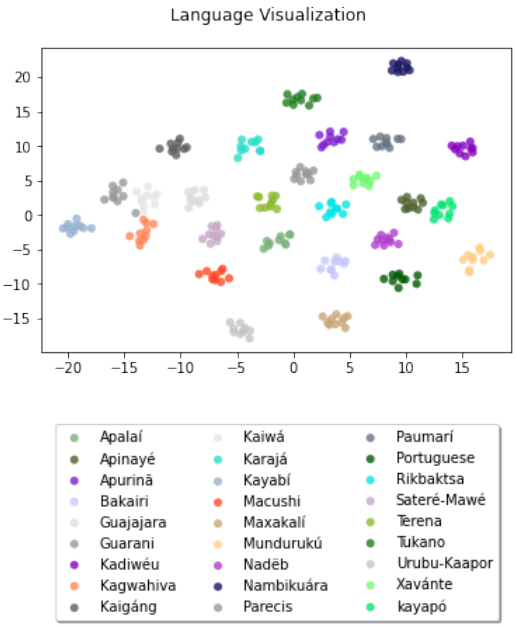

Figure 1: T-SNE projection plot of TF-IDF.

4.1 TF-IDF + Machine Learning

Term Frequency (TF) and Inverse Term Frequency (IDF) are well-known methods to extract features (Dadgar et al., 2016; Kadhim, 2019; Li and Shen, 2017) from the text. Term Frequency calculates how many times a specific term appears in a document (Yamamoto and Church, 2001). IDF calculates the inverse term frequency of each term considering all the documents (Yamamoto and Church, 2001). We applied TF-IDF as baseline feature extractor to the dataset. A visualisation of a 2-D t-SNE (LJPvd and Hinton, 2008; Krijthe and Van der Maaten, 2015; Van Der Maaten, 2014) projection of the extracted TF-IDF from 10 samples (per language) can be seen in figure 1. Each point corresponds to a verse (i.e., a document) from the dataset. One can note that the TF-IDF extracted features can provide a highly separable representation of the languages.

Linear **Support Vector Machines (SVMs)**, **Multinomial Naïve Bayes (NB)** have been used several times in the language identification task. A common feature-set considered in such works is TF-IDF, given its wide acceptance as baseline feature extractors for unstructured texts. In the work (Çöltekin and Rama, 2016) an SVM model outperformed a DNN. Also, a MultinomialNB approach achieved 99% accuracy in the experiment performed by Tan et al. (2014).

4.2 Language Profiles

4.2.1 TextCat

TextCat is a language categorization method, in which each language has a specific profile based on the n-grams term frequency of pre-defined word categories (Cavnar et al., 1994)(Jauhiainen et al., 2019). The results reported by the authors combined with the availability of many "off-the-shelf" pre-trained language profiles motivated an extreme popularity of the method (Jauhiainen et al., 2019). One of the mains issues addressed by implementations of TextCat is classification speed as well as lack of support for minority population languages (Jauhiainen et al., 2019).

The TextCat algorithm consists of extracting over-lapping character n-grams of sizes 1-5 from labelled texts. The language profiles are created by the counts of every unique n-gram over the training corpus. The k most frequent n-grams are then used as the language profile. Prediction of a test document can then be performed by the calculation of the out-of-place distance: the difference in ranks between each of the k most frequent n-grams, being equivalent to Spearman's disarray measure (Jauhiainen et al., 2019).

4.2.2 Word Rank

This paper proposes a simplification of TextCat to identify a document based on its language. To reduce prediction latency, the proposed method only counts how many of character n-grams from the test document are present in each language profile instead of sorting the character n-grams of test document by its frequency as in the original TextCat algorithm (Cavnar et al., 1994).

There is a crucial difference in the prediction step which can improve the proposed method significantly better. It is not necessary to arrange the character n-grams of the test profile by the frequency. Different from the original proposed since the proposed algorithm does not calculate the out-of-place distance, there is no need of sort the n-gram character in the prediction step. The reason why is that the only information necessary is how many of the character n-grams from the test document are present in each training profile. It could improve the method's speed while decreasing the costs.

More specifically, as in TextCat, the WordRank algorithm produces a language profile of character n-grams (Jauhiainen et al., 2019; Cavnar et al., 1994), containing the 300 most frequent n-gram character terms of a text from a specific language (Cavnar et al., 1994). However, with the WordRank algorithm, the prediction step consists of simply counting how many character n-grams of the test document are present in the set of the 300 most frequent n-grams of each language profile. The language profile with the biggest number of matches is the predicted language.

Language	Character trigrams
Apalai	apa, par, arã, rão, ão , o m, mo, mok, oky, k...
Apinaye	• n, n, n , a, ap, apr, pra, raã, aãw, ã...
Apurina	ini, nin, ini, niã, iã , ã i, ia, ia , a a, ...
bakairi	sag, agu, guh, uho, hoe, oem, em , m k, ku, k...
guajajara	het, eta, ta , a à, àm, àmà, màr, àrà, ràà, à...

Table 2: The table shows the characters trigrams for the first 5 languages from table 1 (from the same sample verse)

5 Experiments

As in (Linares and Oncevay-Marcos, 2017), this paper explore the use the low resource data to train the models evaluated.It was one of the issues exposed in a recent survey by (Jauhiainen et al., 2019).

Besides, the 1000 documents (verses) from each language were split in 80% for training and 20% for testing. In order to test the performance of each classifier considering a small amount of data, it was used performed a progressive test. In each round, a randomly chosen sample from the training set were considered for training, i.e., in the first experiment, only 1 example was selected per language, in the second 2 examples per language were chosen, up to 10 examples per language. In all the tests, the test set remained fixed as the 20% of the first split of data.

The experiments were performed with the Natural Language Toolkit (NLTK) (Loper and Bird, 2002) implementation of TextCat[1]. It created the profiles for each language using the training documents of each experiment. Furthermore, each language profile was produced considering the character trigrams. Besides, the WordRank algorithm ranked the 300 most frequent character n-grams for each language. The Machine Learning models, were evaluated considering character trigrams and bigrams as features, as well as TF-IDF, as proposed in (Linares and Oncevay-Marcos, 2017).

[1]https://www.kite.com/python/docs/nltk.textcat

Table 3 and figure 2 shows the result of the experiments considering the weighted average F1-score as performance metric.

Method	Mean	Median	Standard Deviation	Mean Prediction Time (sec)
LinearSVC	0.999074	0.999074	0.000131	0.034423
MultinomialNB	0.999037	0.999074	0.000331	0.035134
TextCat	0.996876	0.997963	0.002537	63.042895
WordRank	0.998889	0.999074	0.000370	23.124780

Table 3: F1-scores for each training set size.

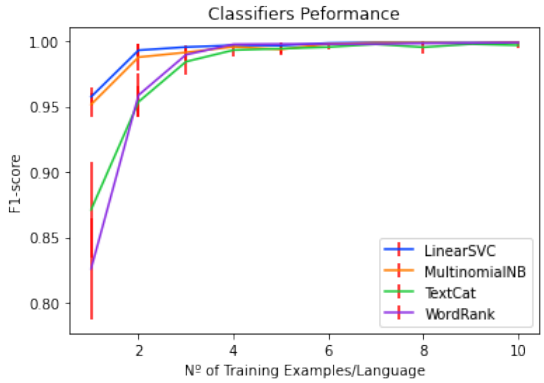

Figure 2: The figure shows the progressive performance of each classifier along the experiment. The x-axis is the number of examples per language and the y-axis is the performance in terms of F1-score.

6 Discussion

Considering the machine learning models, it is clear that they dominate the language identification challenge in this case. It was possible to have significant results even with a small amount of data to train the models as showed in 3 and 2. Further, they also spend much less time than their competitors 3. In summary, the ML models remained as the best models in terms of performance. It corroborates with the results of previous works (Çöltekin and Rama, 2016; Tan et al., 2014) which use TF-IDF and Machine Learning regardless the difference of corpus used and methods details.

Despite do not outperform Machine Learning models, the profile based algorithms had significant results. All the profile based had F1-score over0.80 considering only one labelled document per language. Despite the simplicity involving profile based algorithms, they achieved 100% accuracy with only 10 examples. In that way, the WordRank classifiers much less time consuming than

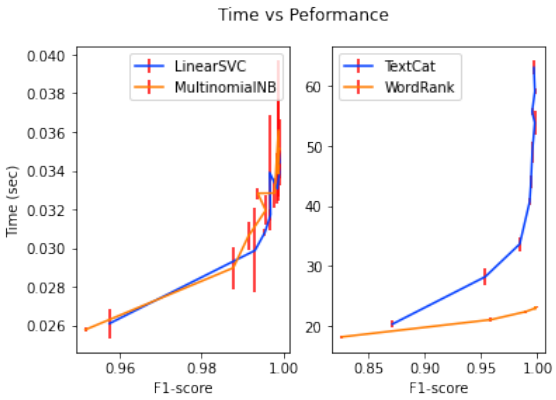

Figure 3: The figure shows the progressive performance of each classifier along the experiment and the time spend in the prediction step.

the TextCat and appears as an option as a profile based algorithm with low latency is required.

7 Conclusion

This paper presents an experimental evaluation of major Language Classification baseline methods for the identification of 26 Brazilian indigenous languages task. It also evaluated a simplification of the TextCat algorithm, which demonstrated to have equivalent performance, with a much smaller computational cost.

Moreover, this paper shows that it is possible to achieve satisfactory results using a small amount of data to train algorithms for Brazilian indigenous Language Identification, with both ML and profile based methods.

References

Angelo. 2016. 26 versões da bíblia em idiomas indígenas para mysword.

William B Cavnar, John M Trenkle, et al. 1994. N-gram-based text categorization. In *Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval*, volume 161175. Citeseer.

Çağrı Çöltekin and Taraka Rama. 2016. Discriminating similar languages with linear svms and neural networks. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 15–24.

S. M. H. Dadgar, M. S. Araghi, and M. M. Farahani. 2016. A novel text mining approach based on tf-idf and support vector machine for news classification. In *2016 IEEE International Conference on Engineering and Technology (ICETECH)*, pages 112–116.

Sebastian Drude, Nilson Gabas Jr, and Ana Vilacy Galucio. 2007. Avanços da documentação sobre línguas indígenas no Brasil. page 4.

Tommi Sakari Jauhiainen, Marco Lui, Marcos Zampieri, Timothy Baldwin, and Krister Lindén. 2019. Automatic language identification in texts: A survey. *Journal of Artificial Intelligence Research*, 65:675–782.

A. I. Kadhim. 2019. Term weighting for feature extraction on twitter: A comparison between bm25 and tf-idf. In *2019 International Conference on Advanced Science and Engineering (ICOASE)*, pages 124–128.

Jesse H Krijthe and L Van der Maaten. 2015. Rtsne: T-distributed stochastic neighbor embedding using barnes-hut implementation. *R package version 0.13, URL https://github. com/jkrijthe/Rtsne*.

Y. Li and B. Shen. 2017. Research on sentiment analysis of microblogging based on lsa and tf-idf. In *2017 3rd IEEE International Conference on Computer and Communications (ICCC)*, pages 2584–2588.

Alexandra Espichán Linares and Arturo Oncevay-Marcos. 2017. A low-resourced peruvian language identification model. In *CEUR Workshop Proceedings*. CEUR-WS.

Maaten LJPvd and GE Hinton. 2008. Visualizing high-dimensional data using t-sne. *J Mach Learn Res*, 9:2579–2605.

Edward Loper and Steven Bird. 2002. Nltk: The natural language toolkit. In *In Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. Philadelphia: Association for Computational Linguistics*.

Denny Moore, Ana Vilacy Galucio, and Nilson Gabas Jr. 2008. O desafio de documentar e preservar as línguas amazônicas. *Scientific American Brasil*, 3:36–43.

Liling Tan, Marcos Zampieri, Nikola Ljubešic, and Jörg Tiedemann. 2014. Merging comparable data sources for the discrimination of similar languages: The dsl corpus collection. In *Proceedings of the 7th Workshop on Building and Using Comparable Corpora (BUCC)*, pages 11–15. Citeseer.

Laurens Van Der Maaten. 2014. Accelerating t-sne using tree-based algorithms. *The Journal of Machine Learning Research*, 15(1):3221–3245.

Mikio Yamamoto and Kenneth W. Church. 2001. Using suffix arrays to compute term frequency and document frequency for all substrings in a corpus. *Computational Linguistics*, 27(1):1–30.