Classificação Supervisionada de Idiomas Indígenas Brasileiros

1

Abstract. *The automatic language identification process is still a significant topic nowadays. One of the reasons is the large number of languages which are over 7000 Ethnologue. In that context, the Brazilian Indigenous languages suffer the risk of extinction by the end of this century. In that way, the classification process is fundamental to aid in the documentation process of new languages resources of those languages. Therefore, this paper explores the supervised classification of 10 different languages and provide a benchmark analysis of different classifiers.*

Resumo. *O processo de identificação automática de idiomas continua sendo um tópico importante a ser explorado. Um das razões para isso é a existência de um grande número de idiomas, chegando a mais de 7000 de acordo com Ethnologue. Nesse contexto, existem os idiomas indígenas a qual sofrem risco de extinção até o fim desse século. Sendo assim, o processo de classificação automática se torna fundamental para ajudar no processo de documentação de mais materiais textuais relacionados ao idioma. Portanto, nesse trabalho são exploradas técnicas de classificação supervisionada usando 10 idiomas, bem como, é realizada uma análise de benchmark dos diferentes modelos usados.*

1. Introdução

A identificação de idiomas tem sido um tema explorado por muitos anos e o principal objetivo é a separação automática de documentos baseada no idioma a qual eles estão escritos. Esse tipo de abordagem tem sido utilizada para a criação de *engines* de busca capazes de encontrar textos para diversos idiomas [Jauhiainen et al. 2019]. Isso é extramente útil para que algoritmos que depende de uma grande quantidade de exemplos textuais para obter resultados significativos [Jauhiainen et al. 2019]. Um exemplo disso são algoritmos de tradução, sumarização abstrativa entre outros [Vaswani et al. 2017, Zhang et al. 2019]. Portanto, a identificação de idiomas tem uma grande relevância não só para identificar documentos em diferentes idiomas, mas também como suporte para outras atividades ligadas a área de Processamento de Linguagem Natural (PLN).

Apesar de ter sido estudada por muitos anos, existem desafios a serem vencidos dentro do campo da identificação automática de idiomas. Um deles está relacionado a grande quantidade de idiomas que chega a mais de 7000 de acordo com Ethnologue [Lewis 2009]. Sendo assim, alguns grupos linguísticos acabam por não receber tanta atenção no processo de classificação automatizada. Como exemplo disso, temos o grupo de idiomas indígenas brasileiros que, de acordo com Drude et al., chega a ter entre 150 a 180 idiomas existentes dependendo da forma de classificação [Drude et al. 2007]. Ademais, todo esse grupo idiomático sofre o risco sério de extinção até o fim desse

século [Drude et al. 2007]. Dessa maneira, a identificação de idioma aparece como uma ferramenta essencial na classificação desses idiomas e no auxilio do processo de documentação dessas línguas.

Nesse contexto, os algoritmos de aprendizado de máquina tem tido um papel significativo, inclusive na identificação de idiomas similares [Çöltekin and Rama 2016, Jauhiainen et al. 2019]. Um exemplo utilizando esse tipo de classificação foi proposta no trabalho realizado por Gomez et.al. Nesse trabalho, foram utilizados métodos simples, mas eficazes de pré-processamento como a trucagem de palavras e o *padding* [Çöltekin and Rama 2016]. Além disso, a extração de características consistiu-se em um simples do uso da frequência e a frequência inversa (TF-IDF sigla em inglês) considerando caracteres *n-grams*. Os resultados obtidos usando Máquina de Vetor de Suporte (MVS) superou os resultados usando uma rede neural [Çöltekin and Rama 2016]. No trabalho [Çöltekin and Rama 2016], também foram realizados experimentos usando o algoritmo de Regressão Linear (RL), que obteve um resultado um pouco abaixo da MVS. Em outro trabalho, Tan et al. também utilizou TF-IDF para extração de informações relevantes e o pré-processamento de textos provenientes de diferentes idiomas [Tan et al. 2014]. Dessa vez, usando Multinomial Naïve Bayes (MultinomialNB), foi possível obter um resultado de 99% de acurácia. O classificador também já foi usado no âmbito da classificação de documentos de forma comparativa como no trabalho [Singh et al. 2019]. Sendo assim, algoritmos de aprendizado de máquina têm um grande potencial na área de classificação de idiomas, podendo ser perfeitamente usados para esse propósito.

Assim sendo, esse trabalho propõe a classificação de idiomas indígenas, sendo realizados 50 experimentos com diferentes abordagens. Além disso, será realizada uma análise preliminar dos dados usando métodos de redução de dimensionalidade. Também, são explorados diferentes modelos como MVS, Regressão Linear e Multinomial Naïve Bayes. Sendo assim, a exploração de classificação automática de idiomas indígenas brasileiros é fundamental para o processo de documentação desses idiomas. Isso permite que mais textos possam ser encontrados de maneira automática na internet através de *engines* de busca e outras ferramentas.

2. Classificadores

Algoritmos de Aprendizado de Máquina têm sido ótimos modelos para classificação de idiomas. Eles têm apresentado resultados promissores na categorização de idiomas embora usem métodos simples de classificação. Vários trabalhos já foram realizados usando esses modelos [Jauhiainen et al. 2019]. No trabalho aqui realizado, iremos focar em três, são eles: MVS, MulitnomialNB e Regressão Linear (RL).

Entre um dos algoritmos mais usados na classificação de idiomas, temos a MVS. Assim como as redes neurais, a MVS tem por objetivo a aproximação de uma função de multivariável [Wang 2005]. No caso de uma classificação binária, o algoritmo tenta encontrar um hiperplano de maneira que a distância entre as duas classes seja a máxima possível [Jauhiainen et al. 2019]. No contexto abordado nesse artigo, em termos gerais, é desejável que o classificador possa ser capaz de separa os dados entre um número considerável de classes. Sendo assim, uma forma de se estender esse conceito para mais de uma classe é fazer um verso o restante e obter a classificação com a maior

pontuação [Jauhiainen et al. 2019]. Além disso, o modelo é compatível com *kernel* e nesse sentido o *kernel* linear é o mais popularmente usado [Jauhiainen et al. 2019].

Outro modelo usado é a RL, que pode obter resultados tão bons quanto MSV na classificação de idiomas, e é bastante rápido em termos de tempo de execução [Jauhiainen et al. 2019]. A RL é um modelo que consiste na otimização de um conjunto de pesos de modo a minimizar a função de custo dada por *regularized negative log-likelihood*:

$$P^{LR} = C \sum_{i=1}^{l} log(1 + e^{-y_i wTx_i}) + \frac{1}{2} w^T w \tag{1}$$

onde C é a penalidade a ser definida e deve ser maior do que 0 [Yu et al. 2011]. O modelo tenta aproximar-se do rótulo pela equação:

$$P_w(y = \pm 1|x) \equiv \frac{1}{1 + e^{-y_i wTx_i}} \tag{2}$$

Embora essas equações funcionem apenas para classificação binária, o conceito pode ser facilmente estendido alterando a equação 1 e 2 para calcularem o entropia máxima. Ainda sendo, o modelo de RL permite a classificação multi-classe e isso tem permitido o uso desse algoritmo na resolução de problemas relacionados a área de PLN, principalmente de identificação de documentos [Yu et al. 2011].

Por fim, um outro modelo explorado na classificação de idiomas é o MultinomialNB [Jauhiainen et al. 2019]. O algoritmo baseia-se na análise de probabilidades referentes ao conjunto de *features* (característica) ligadas a cada classes (idioma). Isso fica evidenciado na equação 3.

$$R_{prod}(g, M) = \prod_{i}^{lM^F} vC(f_i) \tag{3}$$

Nessa função, f_i e $vC(f_i)$ correspondem a cada *feature* independente. Sendo assim, f_i é uma *feature* do exemplo de testes no documento M enquanto $vC(f_i)$ corresponde ao mesmo no modelo ao idioma g. O idioma com a maior probabilidade é o selecionado. Em dos exemplos dessa aplicação é discutida no na comparação realizada no trabalho [Singh et al. 2019].

Sendo assim, a classificação de idiomas ainda possui alguns desafios a serem vencidos [Jauhiainen et al. 2019]. Apesar disso, os algoritmos de aprendizado de máquina, como os aqui citados tem demonstrado, têm apresentado resultados relevantes para a área. Portanto, será utilizado linear MVS, MultinomialNB e RL na classificação dos idiomas indígenas explorados nesse artigo. Todos eles apresentaram resultados significativos nesse trabalho, como fica demonstrado na seção 4.

3. Matérias e Métodos

Como foi destacado, a classificação de idiomas é um assunto explorado de diferentes maneiras, existindo uma grande quantidade de técnicas existente para esse propósito. Para

atingir esse objetivo, existem métodos que buscam transforma os documentos em vetores números para que essas informações possam ser usadas por algoritmos de aprendizado de máquinas tradicionais [Çöltekin and Rama 2016, Tan et al. 2014]. Outras técnicas, como a utilizada no algoritmo TextCat, buscam apenas selecionar os caracteres n-*grams* mais relevantes para cada idioma [Cavnar et al. 1994]. Além disso, as técnicas utilizadas para a classificação automática de idiomas, em geral, são simples e utilizam algoritmos baseado em distância, algoritmos não supervisionados e algoritmos supervisionados de aprendizado de máquina [Jauhiainen et al. 2019, Gebre et al. 2013]. Por fim, testes de hipótese tem sido fundamentais na avaliação de algoritmos e será mostrado em uma breve discussão como isso ajuda na seleção de modelos no âmbito da classificação automática [Ismail Fawaz et al. 2019]. Sendo assim, a seguir, iremos explorar o uso da frequência e a frequência inversa (TF-IDF sigla em inglês) como forma de extração de características dos documentos no âmbito da classificação de idiomas e será discutido o uso de testes de hipótese.

A técnica de TF-IDF calcula o nível de informação provida por cada termo existente dentro de cada documento.

$$IDF = -log_2 \frac{df(t)}{D} \tag{4}$$

Aqui, D é o número de documentos e df é uma função que calcula a frequência do termo *t*. A função log é usada para a normalização dos valores existentes [Yamamoto and Church 2001]. Esse método se provou efetivo na classificação de idiomas mesmo quando eles são similares [Çöltekin and Rama 2016]. Essa abordagem foi usada em diversos artigo da área, tanto usando classificação supervisionada, como não supervisionada. No trabalho realizado por Gebre et al., um sistema de classificação usando algoritmos supervisionados obteve um desempenho substância na classificação de 11 idiomas usando como base a extração de informações usando TF-IDF [Gebre et al. 2013]. Sendo assim, será utilizado TF-IDF para extração de características relacionadas aos documentos de cada idioma. Seguindo a abordagem similar a desenvolvida por Pacella et al., cada documento será transformado em um vetor de características usando TF-IDF.

ANOVA e o teste de Friedman são comumente usados na avaliação de classificadores diferentes a fim de se concluir se ambos tem ao não uma performance similar. Em geral os classificares são organizados em ranques e ajudam a mostra se os resultados dos classificadores não são meramente aleatórios. A organização dos classificadores em um ranque se dá pelo método de Wilcoxon fazendo comparações dois a dois. Portanto, para ranquear e verificar se todos os modelos têm média a mesma performance ou não foi utilizado o teste de Friedman com post-hoc de Wilcoxon-Holm em um abordagem proposta por Fawaz et al. [Ismail Fawaz et al. 2019].

Portanto, nesse trabalho será explorado o uso de classificadores discutidos na seção 2, utilizando-se como pré-processamento o TF-IDF. Por fim, será realizada uma análise estatística para verificar qual o melhor classificador por meio da utilização do teste de Friedman post-hoc de Wilcoxon-Holm.

4. Análise Experimental

Nos experimentos analisados nesse trabalho foram utilizados versões da bíblia em 10 idiomas indígenas. Após um processamento para a extração de características foram usados Classificador de Vetor de Suporte Linear (LinearSVC sigla em inglês), RL e MultinomialNB implementados na biblioteca do scikit-learn como foi feito no trabalho [Çöltekin and Rama 2016] para os algoritmos LR e MVS. Nesse mesmo trabalho a regularização do modelo LinearSVC e LR foi feita considerando o parâmetro C igual a 1.00, isso foi também feito para o algoritmo LinearSVC no trabalho [Çöltekin and Rama 2016]. Nos experimentos aqui realizados o mesmo valor foi considerado para os modelos LinearSVC e LR. Foram utilizadas como forma de extrair características dos textos a técnica de TF-IDF de *bigrams* e *trigrams* caractere de cada documento como uso no trabalho [Linares and Oncevay-Marcos 2017]. O TF-IDF de cada conjunto de caracteres (bigrams, trigrams) será considerado como as *features* a serem consideradas nos experimentos.

Em análise preliminar, pode-se ver na figura 1 a disposição dos idiomas em uma projeção 2D. Foi utilizado *t-distributed Stochastic Neighbor Embedding* (TSNE) para a redução da dimensionalidade e uma visualização dos dados. Essa é uma forma bastante popular para visualização de dados em alta-dimensionalidade. Nessa técnica os objetos são distribuídos de tal forma que pares semelhantes tem uma alta probabilidade na distribuição. O modelo é treinado para minimizar a divergência em Kullback-Leibler [Van Der Maaten 2014]. Após a redução da dimensionalidade das originais 18095 para apenas duas, os dados foram postos em uma escala entre 0 e 1.

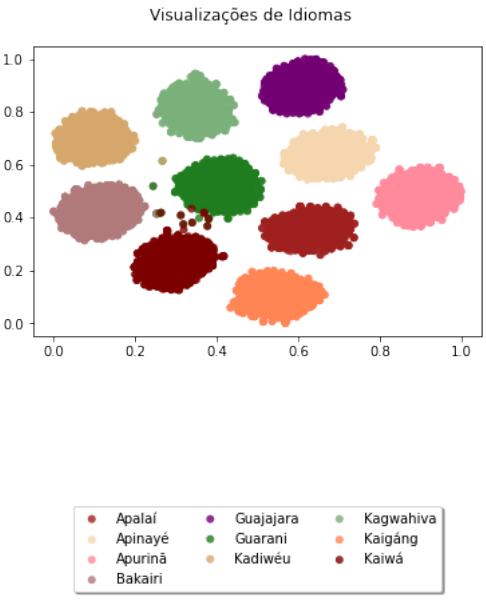

Figure 1. O ranque o mostra o desempenho de cada modelo considerando a acurácia.

Na série de experimentos realizados foram consideradas todo o vetor de características obtido através do pré-processamento usando TF-IDF. Os dados são separados em 67% para treinamento e 33% para testes sendo realizados 5 vezes 10 separações em *cross- validation* dos dados, resultando em um total de 50 experimentos para cada algo-

ritmo. O tempo médio é referente a média de tempo considerando o tempo de predição e treinamento somados em segundos. A primeira métrica utilizada foi a acurácia que provê uma visão geral do desempenho dos classificadores considerando apenas o quanto eles acertaram e erraram. Porém esse é uma métrica utilizada em trabalhos como os de avaliação de classificadores de idiomas similares [Zampieri et al. 2015]. A primeira métrica é baseada na contribuição dado tanto pela precisão do modelo quanto pelo recall [Pedregosa et al. 2011]. A precisão leva em consideração a relação entre os falsos positivos e verdadeiros positivos. Quanto menor o número de falso positivos maior é o score e vice-versa [Pedregosa et al. 2011]. O recall leva em consideração os falso negativos ao invés de falso positivos. Portanto, ela é um balanceamento em relação a precisão. Sendo assim, foi utilizado f1-score para avaliação dos algoritmos utilizados. Nesse trabalho, será utilizado o cálculo do F1-macro que calcula a média de desempenho dos modelos considerando todas as classes com pesos iguais [**?**].

Acurácia: Classificador	Média	Mediana	Desvio Padrão	Tempo Médio
LinearSVC	0.999929	0.99995	0.000041	2.398796
LogisticRegression	0.999929	0.99995	0.000041	23.829660
MultinomialNB	0.999939	0.99995	0.000038	0.391889
F1-macro Classificador	**Média**	**Mediana**	**Desvio Padrão**	**Tempo Médio**
LinearSVC	0.999930	0.999949	0.000040	2.398796
LogisticRegression	0.999929	0.999949	0.000041	23.829660
MultinomialNB	0.999939	0.999950	0.000038	0.391889

Table 1. Os resultados final das 50 execuções realizadas por cada classificador.

Nesse caso, é possível notar um super ajuste dos modelos ao conjunto de dados. Além disso, o classificador *LinearSVC* é o que apresenta o melhor resultado em termos de acurácia média e desvio padrão. Contudo, o modelo *Multinomial NB* também possuí ótimos resultados com um resultado um pouco inferior, porém os resultados são em média mais rápidos. Além disso, nesses experimentos fica claro que há sinais de *overfitting* em todos os modelos usados. Sendo assim, é necessário a análise de técnicas que possam vir a mitigar esse problema. O teste de hipótese confirma a análise inicial de que os algoritmos *LinearSVC* têm um resultado muito similar e portanto a hipótese nula é aceita e, portanto, todos os classificadores têm um desempenho equivalente. Em última análise, pode-se perceber que classificação considerando as mais de 18 mil *features*, pode-se obter um excelente resultado em relação a todos os modelos. Isso incluí tanto o modelo probabilístico como os lineares. Após uma análise realizada considerando o gráfico 1, pode-se inferir que o problema é relativamente simples de ser resolvido principalmente considerando a alta dimensionalidade dos dados.

5. Conclusão

A classificação de idiomas indígenas é de grande relevância, principalmente para os idiomas que ainda não possuem um classificador automático. Nesse sentido, esse trabalho

apresenta um avanço na classificação de idiomas indígenas. Primeiramente, a análise da extração de características usando TF-IDF mostrou o quanto a técnica é poderosa para garantir um resultado ótimo na classificação dos idiomas aqui testados. Isso pode ser útil para direcionar a extração de característica para classificadores não supervisionados e outras abordagens relacionadas a classificação de idiomas. Sendo assim, ficou evidente que nos primeiros experimentos todos os algoritmos obtiveram um resultado similar e de quase 100%. Isso confirma a premissa de que, pela análise inicial dos características vistas na figura 1, o problema é de fácil classificação, causando um *overfitting*. Por fim, é possível que as técnicas aqui exploradas possam ser utilizadas para outras abordagens do problema de classificação de idiomas, bem como, auxiliar na documentação automática dos idiomas indígenas brasileiros.

References

Cavnar, W. B., Trenkle, J. M., 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.

Çöltekin, Ç. and Rama, T. (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.

Drude, S., Jr, N. G., and Galucio, A. V. (2007). Avanços da documentação sobre línguas indígenas no Brasil. page 4.

Gebre, B. G., Zampieri, M., Wittenburg, P., and Heskes, T. (2013). Improving native language identification with tf-idf weighting. In *the 8th NAACL Workshop on Innovative Use of NLP for Building Educational Applications (BEA8)*, pages 216–223.

Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., and Muller, P.-A. (2019). Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery*, 33(4):917–963.

Jauhiainen, T. S., Lui, M., Zampieri, M., Baldwin, T., and Lindén, K. (2019). Automatic language identification in texts: A survey. *Journal of Artificial Intelligence Research*, 65:675–782.

Lewis, M. P., editor (2009). *Ethnologue: Languages of the World*. SIL International, Dallas, TX, USA, sixteenth edition.

Linares, A. E. and Oncevay-Marcos, A. (2017). A low-resourced peruvian language identification model. In *CEUR Workshop Proceedings*. CEUR-WS.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Singh, G., Kumar, B., Gaur, L., and Tyagi, A. (2019). Comparison between multinomial and bernoulli naïve bayes for text classification. In *2019 International Conference on Automation, Computational and Technology Management (ICACTM)*, pages 593–596.

Tan, L., Zampieri, M., Ljubešic, N., and Tiedemann, J. (2014). Merging comparable data sources for the discrimination of similar languages: The dsl corpus collection. In *Pro-*

ceedings of the 7th Workshop on Building and Using Comparable Corpora (BUCC), pages 11–15. Citeseer.

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

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Wang, L. (2005). *Support vector machines: theory and applications*, volume 177. Springer Science & Business Media.

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

Yu, H.-F., Huang, F.-L., and Lin, C.-J. (2011). Dual coordinate descent methods for logistic regression and maximum entropy models. *Machine Learning*, 85(1-2):41–75.

Zampieri, M., Tan, L., Ljubešić, N., Tiedemann, J., and Nakov, P. (2015). Overview of the dsl shared task 2015. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*, pages 1–9.

Zhang, J., Zhao, Y., Saleh, M., and Liu, P. J. (2019). Pegasus: Pre-training with extracted gap-sentences for abstractive summarization.