Catalogação na publicação
Seção de Catalogação e Classificação

N972c Nunes, Felipe da Cunha Andrade.
Construção e análise comparativa de métodos de classificação para categorização das mensagens direcionadas ao "Fale Conosco" da SEFAZ-PB / Felipe da Cunha Andrade Nunes. - João Pessoa, 2024.
23 f. : il.

Orientação: Thaís Gaudencio do Rêgo.
Coorientação: Yuri de Almeida Malheiros Barbosa.
TCC (Graduação) - UFPB/CI.

1. Aprendizado de máquina. 2. Processamento de linguagem natural. 3. Classificação de tópicos. I. Rêgo, Thaís Gaudencio do. II. Barbosa, Yuri de Almeida Malheiros. III. Título.

UFPB/CI CDU 004.75:004.455

Elaborado por Michelle de Kássia Fonseca Barbosa - CRB-738

Construção e análise comparativa de métodos de classificação para categorização das mensagens direcionadas ao "Fale Conosco" da SEFAZ-PB

Felipe da C. A. Nunes[1], **Thaís G. do Rego**[1], **Yuri de A. M. Barbosa**[1]

[1]Centro de Informática – Universidade Federal da Paraíba (UFPB)
João Pessoa – PB – Brazil

{felipandr,gaudenciothais}@gmail.com, yuri@ci.ufpb.br

Abstract. *Every day people come up with questions or requests for organizations where they benefit from some type of service. Therefore, a service that connects the population with organizations is essential. This is the case of services called "Contact Us". It is extremely important that such services provide good service to the public, providing intelligent options and good user experiences. In this work, the use of the "Contact Us" from the State of Paraíba Finance Department was emphasized and seeking to improve the experience of using the tool, classify messages according to the topics related to it. Thus, this work analyzes and compares the Naive Bayes, decision tree and logistic regression classification methods in conjunction with message vectorization using Term Frequency-Inverse Document Frequency. In addition, a qualitative analysis of the model with the best performance was carried out against an unobserved data set. The method with the best performance analyzed was Naive Bayes with a balanced accuracy of 51.0% and based on this, it is possible to see that the models are able to identify patterns and make good distinctions between the classes worked on, but there is still room for improvement in relation to the predictions made.*

Resumo. *Diariamente as pessoas surgem com dúvidas ou solicitações para as organizações em elas usufruem de algum tipo de serviço. Desse modo, um serviço que ligue a população com as organizações é essencial. É o caso de serviços chamados de "Fale Conosco". É de extrema importância que tais serviços prestem um bom atendimento ao público, fornecendo opções inteligentes e boas experiências de uso. Neste trabalho, foi enfatizado o uso do "Fale Conosco" da Secretaria de Fazenda do Estado da Paraíba e buscando melhorar a experiência de uso da ferramenta, realizar a classificação das mensagens nos assuntos a ela relacionados. Sendo assim, este trabalho faz uma análise e comparação dos métodos de classificação Naive Bayes, árvore de decisão e regressão logística em conjunto com a vetorização das mensagens utilizando o Term Frequency-Inverse Document Frequency. Além disso, foi realizada uma análise qualitativa do modelo com o melhor desempenho ante um conjunto de dados não observado. O método com o melhor desempenho analisado foi o Naive Bayes com uma acurácia balanceada de 51,0% e baseado nisso, é possível perceber que os modelos conseguem identificar padrões e fazerem boas distinções entre as classes trabalhadas, mas que ainda há espaço para melhorias em relação as previsões realizadas.*

1. Introdução

Uma das formas comumente utilizadas, para que a população entrem em contato com as empresas ou organizações, as quais eles utilizam algum tipo de serviço, é através de sistemas conhecidos como serviço de atendimento ao consumidor (SAC) ou “Fale Conosco”.

No ambiente digital, o “Fale Conosco” geralmente é encarado como um formulário em que as pessoas podem preencher vários campos. Esses campos, de acordo com a proposta da empresa/organização, usualmente buscam capturar informações importantes para encaminhar a mensagem da maneira mais adequada, retorno do contato com o remetente e entre outras etapas de tratamento dessa interação das organizações com a população.

Em muita das vezes, a responsabilidade do preenchimento dos campos presentes no formulário como o “Fale Conosco” fica a cargo da pessoa que busca o primeiro contato e esse preenchimento geralmente é manual. De acordo com Jarrett et al. (2009), embora os colaboradores de uma empresa/organização preencham corretamente alguma informação, facilmente o mesmo não é esperado pelos usuários do sistema. Adicionalmente, segundo os autores, cenários onde são utilizadas caixas suspensas para seleção, lidar com muitas opções é um pouco mais complicado sua utilização pelas pessoas, por exemplo. Desse modo, em certas ocasiões, sugerir opções inteligentes ou realizar a escolha automaticamente é útil, podendo mitigar os erros e facilitar o atendimento pelos colaboradores.

Essas constatações podem ser observadas na página do “Fale Conoso” da Secretaria de Fazenda do Estado da Paraíba (SEFAZ-PB), como ilustrado na Figura 1. De maneira mais clara, é visto na Figura 1 que o campo referente ao assunto é uma caixa de seleção suspensa com uma infinidade de assuntos disponíveis para a população escolher. Além da quantidade de informações, o conhecimento prévio do que cada assunto significa ou o que ele desencadeia não é algo que se possa colocar como obrigatório a população.

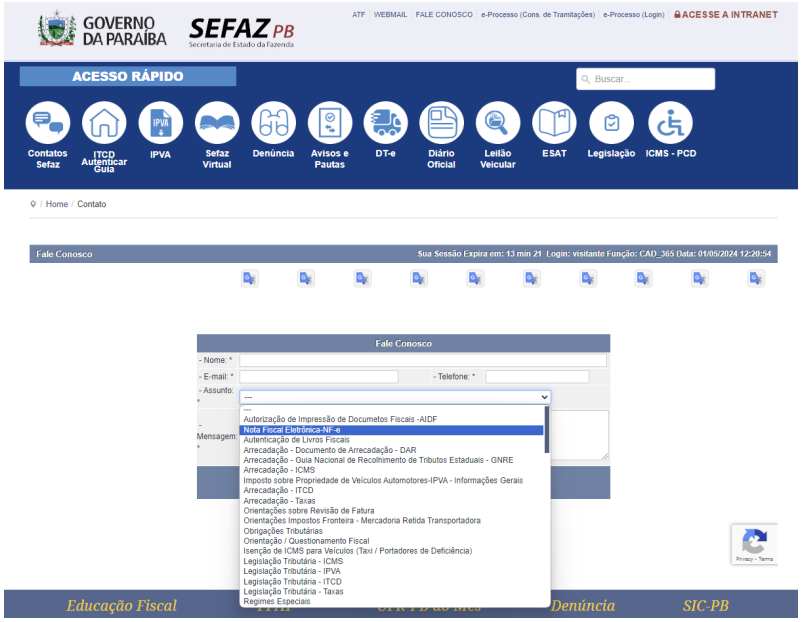


Figura 1. Recorte da página do “Fale Conosco” da SEFAZ-PB

Levando em consideração que problemas de classificação de textos têm sido amplamente estudados e abordados em muitas aplicações reais [Jiang et al. 2018, Aggarwal and Zhai 2012] apud [Kowsari et al. 2019]. É possível aproveitar os métodos de classificação textuais para categorizar certos campos que, porventura, possam ser necessários em formulários como o "Fale Conosco", para o melhor atendimento a população.

Portanto, o presente trabalho tem como objetivo principal analisar e explorar métodos de classificação para categorizar as mensagens, que foram enviadas através da página "Fale Conosco" da SEFAZ-PB, de acordo com os assuntos abordados pela organização. Com esse intuito, o trabalho compreende os objetivos específicos: gerar uma base de dados de solicitações e seus assuntos; avaliar e comparar diferentes métodos de classificação supervisionados, como Naive Bayes, árvore de decisão e regressão logística; e analisar qualitativamente a reclassificação realizada pelo modelo com melhor desempenho.

Além da seção atual, este trabalho está organizado em outras cinco seções. Na Seção 2 são apresentados os trabalhos relacionados. Na Seção 3 são discutidos os métodos empregados no tratamento dos dados, o método de extração dos atributos, os métodos de classificação e as métricas aplicadas. A Seção 4 aborda os resultados e discussões dos experimentos realizados. A Seção 5 é destinada a conclusão do trabalho. Ao final, encontram-se as referências utilizadas.

2. Trabalhos relacionados

A classificação de documentos é uma área do processamento de linguagem natural amplamente estudada. No contexto deste trabalho, a classificação adequada dos documentos pode fornecer uma melhoria nos serviços disponibilizados a população e automação de parte desse processo de definição do tópico dos textos. Nesta seção serão descritos alguns trabalhos correlatos ao presente estudo.

Cortes et al. (2020) teve como objetivo realizar uma revisão dos mais recentes métodos de classificação de perguntas, considerando linguagens com baixo recurso, categorizando os métodos em níveis de dependência de recursos externos (baixa, média, alta e muito alta). Em seguida, os autores compararam a categorização em termos empíricos de dados necessários para treinamento e o desempenho em diferentes linguagens. A categorização dos métodos foi feita a partir das revisões realizas por Loni (2011) e Sangodiah et al. (2015). Os métodos escolhidos para analisar o desempenho em diferentes configurações foram: *Convolutional Neural Network* (CNN) [baixa], *Long short-term memory* LSTM [média], *Bidirectional Encoder-Decoder Transformer-Convolutional Neural Network* (CNN BERT) [média] e Support Vector Machine (SVM) [alta]. Na maior parte dos casos, a métrica utilizada foi a função de perda *Categorical Crossentropy*. Foram utilizados dois conjuntos de dados para avaliar os métodos, o UIUC e DISEQuA. Ambos possuem uma variação de textos rotulados e em diferentes idiomas. Entretanto, além da diferença entre os dois conjuntos de dados, algumas informações referente a representação deles em alguns idiomas não são disponibilizados. Ao todo, os dois conjuntos somam aproximadamente 6450 perguntas representadas em 5 línguas diferentes.

O comparativo executado entre diferentes níveis de dependência apresenta que, desde 2015, abordagens utilizando métodos com dependência média de recursos externos

alcançaram melhores resultados, que aqueles com uma alta dependência, fazendo com que a diferença entre eles seja de 0,5%, em termos de acurácia. Além disso, os autores concluem que, dentre os métodos implementados e avaliados, do ponto de vista do desempenho em diferentes línguas, o CNN BERT obteve o melhor resultado, sendo superior ao outros em todas as línguas e conjuntos de dados utilizados. O CNN BERT obteve uma *Categorical Crossentropy* de 94,3, 90,1 e 80,2, respectivamente nos idiomas inglês, português e espanhol, para o conjunto de dados UIUC. Para o conjunto de dados DISEQuA, foram obtidos 92,0, 91,4, 91,4 e 85,0, nos idiomas inglês, italiano, espanhol e holandês, respectivamente. Por fim, o CNN BERT também teve o melhor desempenho com o menor conjunto de treinamento, em comparação aos outros 3 métodos, chegando ao melhor resultado com pouco menos de 200 exemplos.

O objetivo pretendido por Mohammed and Omar (2020) foi construir um modelo de classificação baseado no domínio cognitivo da taxonomia de Bloom. Para isso, os autores propuseram utilizar uma abordagem com *Term Frequency-Inverse Document Frequency* (TF-IDF) modificado e *word2vec*. E para avaliar o desempenho do modelo proposto, os autores utilizaram dois conjuntos de dados: o primeiro conjunto foi construído através da seleções de perguntas de vários sites, livros e trabalhos prévios, contendo 141 exemplos distribuídos em 6 categorias, levemente desbalanceado e seu volume foi dividido em 80% para treinamento e 20% de teste. O segundo conjunto possui 600 exemplos distribuídos igualmente entre as 6 categorias e divididos em 80% para treinamento e 20% para testes.

O TF-IDF modificado proposto pelos autores se trata de uma etapa do pré-processamento em que foi utilizado o *Part-of-Speech (POS) tagger* para marcar as palavras de acordo com sua classe gramatical, devido a importância que verbos têm nas definições das classes na taxonomia de Blomm, de forma a buscar melhorar o TF-IDF obtido das palavras. Esse processo é chamado pelos autores como TFPOS-IDF. Além disso, após combinar o TF-IDF com *POS tagging*, os autores utilizaram o modelo pré-treinado *word2vec*, buscando extrair o contexto das perguntas. Todo esse processo pode ser representado pela equação *Vetor da pergunta* $= \sum_{t \in d} word2vec(t) \times [TFPOS - DF(t, d)]$, onde é executado o somatório de cada vetor da matriz produzida pelo *word2vec*, representando uma palavra, multiplicado pelo correspondente TFPOS-IDF.

Os algoritmos de classificação utilizados foram *K-Nearest Neighbours* (KNN), regressão logística e SVM. As principais métricas utilizadas durante as avaliações dos modelos foram o *recall*, precisão e *F1 Score* ponderados e para avaliar a abordagem proposta, os autores realizaram a comparação entre os 3 algoritmos em 3 formas de extração de características: TF-IDF, TFPOS-IDF e W2V-TFPOSIDF, observando a progressão do desempenho do TF-IDF ao W2V-TFPOSIDF. Com isso, foi concluído que o método proposto foi superior aos outros dois métodos utilizando os dois conjuntos de dados e os três métodos de classificação, com o melhor resultado obtido utilizando a combinação do W2V-TFPOSIDF com o método de regressão logística, apresentando 89,4% de *F1 Score* médio.

Minaee et al. (2020) fazem uma revisão geral de aproximadamente 150 modelos de aprendizagem para classificações, suas contribuições técnicas, similaridades e pontos fortes. Os autores também proveem um apanhado de mais de 40 conjuntos de dados e uma análise do desempenho de alguns modelos selecionados. Para tal fim, os autores dividi-

ram os métodos em 11 categorias, de acordo com os modelos arquiteturais dos modelos. Desse modo, foram relacionados modelos que passam por redes neurais e até modelos utilizando aprendizado além do supervisionado. É possível destacar que, dentro das categorias envolvendo redes neurais, é possível encontrar modelos voltados a classificação de tópicos, como o TopicRNN proposto por Dieng et al. (2017). Da parte dos conjuntos de dados mais populares no período, os autores dividiu-os em 5 grupos e dentre eles um dedicado a dados referentes a classificação de tópicos onde, para aqueles que possuíam informações referente ao volume, o menor contém 7400 documentos divididos em 23 categorias.

Os autores realizaram a análise de alguns dos modelos e do progresso da acurácia, de acordo com o objetivo de cada modelo. Vale destacar que, utilizando os conjuntos de dados referentes a classificação de tópicos, o modelo com melhor desempenho relatado pelos autores foi o XLNet, com acurácia de 99,4%, utilizando o conjunto de dados DBPedia. XLNet é um modelo de linguagem pré-treinado baseado na arquitetura *transformer* e o conjunto de dados DBPedia possuía, no momento, 630 mil exemplos no total. O outro conjunto de dados utilizado nas comparações foi o Ohsumed, com 7400 exemplos, em que o modelo o Simplfied GCN, que se trata de uma versão simplificada de uma CNN sobre Grafos, teve o melhor desempenho com acurácia de 68,5%.

Por fim, o trabalho proposto aqui se diferencia dos outros ao analisar o desempenho dos algoritmos de Naive Bayes, árvore de decisão e regressão logística, métodos relativamente menos robustos que os apresentados, utilizando um conjunto de dados pequeno e desbalanceado, para a classificação de tópicos. Além disso, esse estudo busca analisar os padrões identificados a partir do algoritmo que apresentou o melhor desempenho.

3. Metodologia

Para alcançar o objetivo proposto de executar a classificação das mensagens enviadas através da página "Fale Conosco" da SEFAZ-PB com relação aos seus assuntos, realizar a comparação entre os modelos e avaliar aquele com o melhor desempenho de maneira qualitativa, nesta pesquisa, foi utilizada uma abordagem quantitativa/qualitativa, empregando métodos de pesquisa metodológica e exploratória. Esta seção divide-se em: apresentação do *corpus*, descrição do pré-processamento dos dados, do método de vetorização, dos métodos de classificação, métricas e da arquitetura onde os experimentos foram executados.

3.1. *Corpora*

Uma maneira simples de definir o que seria um *corpus* é pensar que o termo refere-se a uma coleção de textos ou falas legíveis por computador [Jurafsky and Martin 2019]. De maneira semelhante, de acordo com McEnery et al. (2006), há um consenso que *corpus* é uma coleção de textos autênticos legíveis por máquina (incluindo transcrições de dados falados), que são amostrados para serem representativos de um determinado idioma, ou variedade linguística.

O *corpus* utilizado neste trabalho está contido em um arquivo CSV, organizado em forma de planilha, onde são descritos o setor, a situação do atendimento, o período em que foi solicitado, o assunto da solicitação, a data em que foi realizado o envio da solicitação,

data da resposta, observações, a mensagem do autor da solicitação e a resposta. É importante ressaltar que os documentos reunidos no *corpus* foram anonimizados, tendo em vista que no canal de atendimento utilizado é possível fornecer dados pessoais. Esses dados foram cedidos pela SEFAZ-PB, obtidos a partir do seu canal de comunicação com o público em geral ("Fale Conosco"), em seu site.

Entretanto, nesta pesquisa, as informações relevantes para atingir o objetivo proposto são referentes aos assuntos o qual as mensagens vão ser classificadas e as próprias mensagens enviadas pelos autores através do sistema. Devido a escolha dos assuntos serem inteiramente de responsabilidade dos autores das mensagens, a chance do preenchimento incorreto não é baixa, acarretando em muitas vezes as solicitações serem encaminhadas entre os setores que não são capazes de atenderem-nas. Dessa forma, assume-se que os assuntos das mensagens poderiam ter informações incompatíveis entre eles. Consequentemente, um auditor da SEFAZ-PB analisou os dados e refez a classificação das mensagens conforme os assuntos mais adequados a elas. Além da reclassificação das mensagens com base nos assuntos, foi removido pelo auditor o assunto Arrecadação - ITCD, e os assuntos Abertura processo ITCD, Guia pagamento ITCD e Guia pagamento IPVA foram adicionados. À vista desses circunstâncias, no *corpus* há 202 documentos distribuídos, de acordo com a Tabela 1, em 9 categorias.

Tabela 1. Distribuição do primeiro *corpus* por assuntos

Assunto	**Quantidade**
Isenção de ITCD	3
Abertura processo ITCD	6
Legislação Tributária - ITCD	11
Isenção de IPVA	11
Legislação Tributária - IPVA	14
Guia pagamento ITCD	18
Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais	44
Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais	47
Guia pagamento IPVA	48

Algumas categorias contidas na Tabela 1 podem ser um pouco menos óbvia, devido ao contato com elas não ser tão comum. Como por exemplo, as categorias de legislação. Para ilustrar um pouco melhor como é formado o *corpus*, a listagem a seguir possui uma mensagem para cada assunto referente a legislação, tendo em vista justamente a complexidade deles. Nas mensagens utilizadas como exemplo, os autores buscam saber algo referente a isenção do imposto, mas as respostas podem precisar de algum suporte legal que garantam a possibilidade e condições de obtê-las. Outro cenário em que as mensagens podem ser classificadas de acordo com esses assuntos, são casos de dúvidas referentes as leis diretamente.

- Legislação Tributária - IPVA: Gostaria de saber qual o teto de valor de carro que o Estado PB concede a isenção de IPVA. Se tem o valor específico do carro para ter esse direito.

- Legislação Tributária - ITCD: Gostaria de informações de como proceder para obter o desconto de 50% no ITCMD. Como proceder no requerimento?

Como já era discutido que a quantidade de informações dispostas no primeiro *corpus* poderia ser insuficiente, durante o processo de desenvolvimento da pequisa, foi disponibilizado um segundo *corpus* com um volume de dados maior que o fornecido inicialmente. Entretanto, o segundo *corpus* não passou por um processo de revisão pelos auditores da SEFAZ-PB. Isso quer dizer que as possíveis inconsistências entre o assunto e a mensagem dos documentos não foram corrigidas e o processo de adição e remoção dos assuntos, que foi realizado no primeiro *corpus* pelos auditores, não foi executado no segundo *corpus*. Desse modo, o segundo *corpus* vai ser utilizado neste trabalho para avaliar o desempenho do modelo para os exemplos ainda não vistos por ele, observando o assunto original com o predito e fatores que levaram a essa classificação. O segundo *corpus* possui 2884 documentos e está distribuído entre 7 assuntos conforme a Tabela 2.

Tabela 2. Distribuição do segundo *corpus* por assuntos

Assunto	**Quantidade**
Isenção de ITCD	37
Legislação Tributária - ITCD	81
Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais	304
Isenção de IPVA	405
Arrecadação - ITCD	537
Legislação Tributária - IPVA	579
Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais	941

3.2. Pré-processamento

Durante o pré-processamento dos dados, foi necessário realizar inicialmente algumas modificações nos arquivos disponibilizados pelos auditores da SEFAZ-PB, devido a codificação desconhecida. Em seguida, foram efetuadas a remoção de algumas informações desnecessárias para a execução dos experimentos, como: colunas a mais do CSV, linhas com ausência de dados, padronização dos caracteres, remoção de números e palavras de parada (*stop words*) e aplicação de um método de redução de palavras. A Figura 2 ilustra o diagrama do fluxo de processamento dos dados que serão detalhados a seguir.

3.2.1. Correção da codificação do arquivo

O *corpus*, como mencionado, encontra-se em um arquivo CSV. A princípio, esse arquivo foi disponibilizado em uma codificação desconhecida que, quando aberto em UTF-8, ou em qualquer outra codificação popular, apresentou os caracteres especiais irreconhecíveis. Dessa forma, foi utilizada na biblioteca *ftfy* de Speer (2019) o método `fix_text` em todo o *corpus*, *token* a *token*. A biblioteca *ftfy* utiliza métodos heurísticos e é capaz de corrigir *unicodes* incorretos. Dessa maneira, é possível ler o arquivo em UTF-8 e corrigir os caracteres.

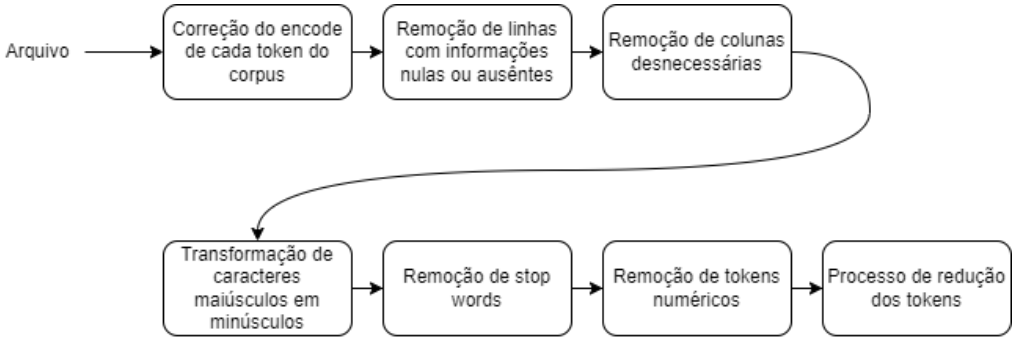


Figura 2. Diagrama do fluxo de pré-processamento

3.2.2. Remoção de colunas desnecessárias

Para o cumprimento do objetivo de classificar as mensagens por assunto, as colunas relevantes são as colunas "mensagem" e "assunto". Dessa forma, as outras colunas do *corpus* são desnecessárias e, consequentemente, foram removidas do conjunto de dados utilizando a função `drop`, da biblioteca *pandas*.

3.2.3. Remoção dos documentos com ausência de atributos ou valores nulos

A ausência de atributos, ou com informações nulas, nos documentos do *corpus* usados pelos métodos de classificação aplicados neste trabalho, não contribuem para a rotulação dos documentos adequadamente. Sendo assim, documentos que se encontravam dessa maneira foram removidos. Para tanto, foram utilizados os métodos `dropna` da biblioteca *pandas*, em conjunto com o método `drop`, para os valores nulos, ou com o texto *nan* nas colunas "mensagem" e "assunto", individualmente. Esse processo reduziu o primeiro *corpus* para 202 documentos. Já no segundo *corpus*, a quantidade final permaneceu com 2884 documentos.

3.2.4. Padronização dos documentos

Em algumas ocasiões, o uso de caracteres maiúsculos e minúsculos nos *tokens* não trazem diferenças no seu significado em documentos na língua portuguesa. Para o contexto desse trabalho, foi aplicada a transformação de todos os *tokens* para caracteres minúsculos. Isso pôde ser feito utilizando a biblioteca *pandas*, com auxílio do método `lower`, através do assessor `str`. Essa transformação só é necessária na coluna "mensagem".

3.2.5. Remoção de numerais

Tendo em vista a necessidade de avaliar somente *tokens* não numerais e também diminuir a influência que os números podem causar indevidamente, é realizada a remoção dos números presentes na coluna "mensagem". Para essa finalidade, foi efetuada uma verificação utilizando o método `isalpha`, disponível por padrão na própria linguagem Python.

3.2.6. Remoção de *stop words*

A classificação das mensagens leva em conta o conteúdo que foi escrito, ou seja, se os *tokens* (ou as palavras) utilizadas são importantes e se sua frequência também tem relevância. Entretanto, alguns *tokens* ocorrem com uma frequência alta e isso não quer dizer

que eles realmente possuem uma importância significativa para definir o tópico central do documento. Esses *tokens*, que são conhecidos como *stop words*, geralmente fazem algum tipo de conexão ajudando na construção das frases. Idealmente, é feita a remoção desses *tokens* do *corpus*. Neste trabalho, foi utilizado o modulo `stopwords` da biblioteca *nltk*, com o pacote de palavras em português. Além disso, após remover os *tokens* definidos por padrão pela biblioteca, com auxílio de uma nuvem de palavras gerada a partir de todos os textos da coluna "mensagem", foram adicionados manualmente novos *tokens* à lista de *stop words*, que foram identificados visualmente. Os *tokens* adicionados manualmente foram escolhidos consoante a necessidade de visualizar na nuvem de palavras somente aqueles que identificassem a que assunto poderia se tratar a palavra. Ademais, dentre outras características observadas nos *tokens* escolhidos como *stop words*, algumas delas estão de acordo com a listagem a seguir:

- Mascaramento de dados sensíveis utilizando o caractere "x", 9 ou 0.
- Introduções, como: prezadas, bom dia, oi, olá, *et cetera*.
- Pronomes mais comuns: ele, eu, entre outros casos.
- Sinais gráficos: ?, ! e |.
- Despedida, saudações do remetente: atenciosamente, agradeço, grata, *et cetera*.

3.2.7. Processo de redução de palavras - *Stemming*

Stemming é uma técnica para redução de palavras a sua raiz. Desse modo, é possível reduzir variações de uma mesma palavra em uma única forma. Entretanto, essa técnica pode levar a palavras que não existem e além disso, é possível que duas ou mais palavras distintas possam ser reduzidas a uma raiz comum, ou até mesmo que o processo de redução não reduza as palavras adequadamente [Orengo and Huyck 2001]. Todavia, o processo de redução das variações das palavras a sua raiz tem seu valor no pré-processamento dos dados, tendo em vista que ele diminui a interferência que as variações que uma palavra possa trazer a classificação do conteúdo [Singh and Gupta 2016].

Neste trabalho foi adotada a implementação `RSLPStemmer` da biblioteca *nltk*. A classe implementa uma versão do algoritmo proposto por Orengo and Huyck (2001). O processo de redução foi realizado individualmente em todas as palavras presentes na coluna "mensagem" do *corpus* utilizado neste trabalho. O algoritmo é dividido em 8 etapas:

1. Redução de palavras no plural para singular
2. Redução de palavras no feminino para masculino
3. Redução de advérbio
4. Redução de aumentativo, diminutivo e superlativo
5. Redução do sufixo de substantivos
6. Redução do sufixo de verbos
7. Remoção da última vogal
8. Remoção de acentos

3.2.8. Separação dos conjuntos de treinamento e teste

A separação dos conjuntos de treinamento e teste foi realizada com o uso do função `train_test_split` da biblioteca *scikit-learn*. Tendo em vista o *corpus* desbalanceado utilizado neste trabalho, a função é relativamente útil para garantir que os conjuntos

de treinamento e teste possuam a mesma proporção de elementos entre as categorias encontradas, porém a função não se restringe a unicamente isso, auxiliando no processo de embaralhamento dos dados, entre outras funcionalidades.

Com esse objetivo, foi utilizada uma estratificação, de acordo com a quantidade de classes do *corpus*, que de acordo com a Tabela 1, possui 9 classes. O conjunto de treinamento foi alocado com 80% do volume total do *corpus* e o conjunto de teste com 20%. Por fim, para garantir a reprodutibilidade, foi utilizado o valor 9 no gerador de números aleatório utilizado no embaralhamento dos dados. Em resumo, os parâmetros utilizado foram: `test_size` de 0,20, `random_state` com valor 9 e `stratify` de acordo com as categorias do *corpus*.

É importante destacar que, com 20% do *corpus* como conjunto de teste e com uma divisão estratificada, a função separará pelo menos um elemento no conjunto de teste para aquelas classes em que esse percentual for menor que um. Isso ocorre, por exemplo, nas classes representadas pelo assunto Isenção de ITCD e Abertura processo ITCD, observadas na Tabela 1.

3.3. Método de vetorização

O método de vetorização utilizado neste trabalho foi o TF-IDF, implementado através da classe `TfidfVectorizer`, da biblioteca *scikit-learn*. Os parâmetros utilizados para realizar a vetorização foram os já definidos por padrão pela própria biblioteca. A vetorização foi realizada para todos os documentos da coluna "mensagem" do *corpus*.

Vale destacar que, a biblioteca *scikit-learn* implementa uma versão um pouco diferente da fórmula padrão, além disso, o resultado da transformação é normalizado utilizando a distância euclidiana. Dado isso, o que é discutido dentro desta seção é referente a implementação empregada especificamente por essa biblioteca.

O TF-IDF consiste no produto de duas métricas: TF e IDF. O método busca identificar os *tokens* mais relevantes para um documento no *corpus*. O fator TF no produto representa a quantidade de vezes que o *token* ocorre em um determinado documento. No *scikit-learn* o fator é contabilizado da mesma maneira. Já o fator IDF procura identificar o quão comum ou raro um termo é em relação ao *corpus*, no caso da implementação padrão empregada pelo *scikit-learn*, o fator IDF pode ser calculado através da equação 1, onde t representa o termo que está sendo avaliado, N, o tamanho do *corpus* e o $df(t)$, a quantidade de vezes que o termo aparece dentro do *corpus*.

$$IDF(t) = \ln\left(\frac{1+N}{1+df(t)}\right) + 1 \tag{1}$$

Desse modo, conforme um termo possua uma frequência maior entre os documentos do *corpus*, o valor do IDF tende a zero. Ou seja, conforme o quão comum um *token* seja aos documentos do *corpus*, menos relevante ele é para o documento avaliado e, consequentemente, o valor TF-IDF é inferior. De forma semelhante, caso o termo não seja comum dentro do *corpus* para o documento avaliado, sua relevância aumenta.

3.4. Métodos de classificação

Neste trabalho foram escolhidos para comparação três algoritmos de classificação: Naive Bayes, árvore de decisão e regressão logística. Todos os algoritmos são categorizados

como métodos supervisionados. A escolha dos hiperparâmetros são importantes, de forma com que seja minimizado os erros na generalização [Bergstra and Bengio 2012]. Encontrar os valores dos hiperparâmetros de um modelo é uma tarefa custosa, mas necessária para quase todos os modelos e conjuntos de dados [Müller and Guido 2016]. A identificação dos melhores hiperparâmetros para cada algoritmo foi realizada utilizando a classe `GridSearchCV` da biblioteca *scikit-learn*, que é capaz de testar todas as combinações dos hiperparâmetros informados a ela. Com a finalidade de avaliar com mais qualidade a capacidade de generalização dos modelos, foi utilizada uma validação cruzada estratificada com 2 subconjuntos (`cv` igual a 2). A justificativa para essa divisão dá-se devido a quantidade de elementos na categoria Isenção de ITCD ser de 2 exemplares no conjunto de treinamento. A Tabela 3 reúne os algoritmos citados, as classes e bibliotecas empregadas e a lista de hiperparâmetros avaliados.

Tabela 3. Algoritmos, implementação e hiperparâmetros utilizados para análise

Algoritmo	Implementação	Hiperparâmetros
Naive Bayes	`MultinomialNB` da biblioteca *scikit-learn*	- alpha = [0,01, 0,1, 0,5, 1,0, 10,0, 12,0, 100,0, 0] - fit_prior = [True, False] - class_prior = [None, [0,0310559, 0,23602484, 0,08695652, 0,23602484, 0,2173913, 0,05590062, 0,01242236, 0,06832298, 0,05590062]]
Árvore de decisão	`DecisionTreeClassifier` da biblioteca *scikit-learn*	- criterion = ['gini', 'entropy', 'log_loss'] - splitter = ['best', 'random'] - max_depth = [None, 3, 5, 9, 202, 772] - min_samples_split = [2, 5, 7, 10, 13, 18 ,20] - min_samples_leaf = [1, 2, 3, 5, 8, 10, 20, 50] - min_weight_fraction_leaf = [0,0, 0,001, 0,003, 0,005, 0,1, 0,2, 0,4] - max_features = [None, 0,1, 0,3, 0,5] - random_state = 9 - max_leaf_nodes = [None, 5, 7, 9, 10, 14, 17, 20, 28, 35, 41, 48] - class_weight = [None, 'balanced']

Algoritmo	Implementação	Hiperparâmetros
Regressão logística	`LogisticRegression` da biblioteca *scikit-learn*	- penalty = ['l2', None] - C = [1,0, 0,1, 0,001, 1,10, 10, 30, 50, 80, 100] - fit_intercept = [True, False] - class_weight = [None, 'balanced'] - random_state = 9 - solver = ['lbfgs', 'newton-cg', 'sag', 'saga'] - max_iter = [100, 850, 1000, 1500] - multi_class = 'multinomial' - n_jobs = -1

3.5. Métricas

Neste trabalho, os métodos de classificação utilizados foram avaliados do ponto de vista de cinco métricas:

- Acurácia balanceada: Acurácia balanceada é definida como a média do *recall* obtido de cada classe. Essa métrica, assim como a acurácia convencional, proporciona uma visão geral do desempenho do modelo. Entretanto, a acurácia balanceada busca lidar com conjunto de dados desbalanceados [Pedregosa et al. 2011].
- Precisão: A precisão é a métrica que mede a proporção de exemplos que o modelo classifica corretamente como positivos. Ela indica a capacidade do modelo de não classificar como positivo um exemplo que seja negativo e evitar falsos positivos. No caso da precisão ponderada, utilizada neste trabalho, é levado em consideração o desbalanceamento das classes considerando a média ponderada, com base nos exemplos verdadeiros de cada classe [Pedregosa et al. 2011].
- *Recall*: O *recall* também é uma métrica que mede a proporção de exemplos que o modelo classifica corretamente como positivos. Entretanto, essa métrica indica a capacidade do modelo de encontrar todos os exemplos positivos e evitar falsos negativos. No *recall* ponderado, também é levado em consideração o desbalanceamento das classes, de acordo com a média ponderada obtida dos exemplos de cada classe [Pedregosa et al. 2011].
- *F1 Score*: O *F1 Score* é uma métrica que combina a precisão e *recall* em uma única. Ela é a média harmônica entre as duas métricas. Ela demonstra uma medida balanceada entre as duas métricas, sem favorecer uma em relação a outra. Também é útil para conjunto de dados desbalanceados. No caso da métrica ponderada, assim como nas outras métricas, o peso de cada classe é calculado com base nos exemplos verdadeiros de cada classe [Pedregosa et al. 2011].
- Área sob a Curva de ROC: A área sob a curva ROC é uma métrica que ajuda na avaliação comportamental do modelo. A curva ROC é um gráfico do *recall*, em função da taxa de falsos positivos para diferentes fronteiras de classificação [Müller and Guido 2016]. No caso de problemas multiclasses, pode-se utilizar as estratégia um-contra-o-resto, onde cada classe é tratada como a classe positiva em uma comparação com todas as outras classes. Devido ao desequilíbrio das classes, também foi utilizado pesos diferentes, de acordo com as classes.

A métrica utilizada durante a validação cruzada foi a acurácia balanceada. Os melhores hiperparâmetros identificados durante o processo foram utilizados para avaliar o conjunto de testes. O modelo que apresentou o melhor resultado na comparação perante o conjunto de testes foi utilizado como referência durante as análises dos resultados.

3.6. Arquitetura

O desenvolvimento desta pesquisa foi realizado em um sistema operacional Windows 11 Pro 64 bits na versão 23H2, distribuição Python da Anaconda na versão 23.7.4 com a linguagem de programação Python na versão 3.10.0.

O hardware utilizado para o desenvolvimento possui 16GB de memória RAM, um processador AMD Ryzen 5 3600 com frequência base de 3.59 GHz e *boost* de até 4.2 GHz, contendo 6 núcleos e 12 processadores lógicos (*threads*).

4. Resultados e discussões

Nesta seção são apresentados os resultados obtidos na pesquisa, assim como os pontos de discussões referentes ao desempenho dos algoritmos utilizados na comparação. Como dito anteriormente, a comparação dos algoritmos de classificação foi realizada através da acurácia balanceada como métrica principal, obtida utilizando os hiperparâmetros identificados por meio da busca exaustiva executada pelo `GridSearchCV`. Desse modo, a Tabela 4 apresenta as melhores métricas obtidas pelos modelos, de acordo com os métodos descritos na Seções 3.4 e 3.5 e o conjunto de testes utilizando o primeiro *corpus*.

Tabela 4. Comparativo dos algoritmos de classificação com relação as métricas utilizando o conjunto de testes

Algoritmo	**Acurácia balanceada (%)**	**Precisão ponderada (%)**	***Recall* ponderado (%)**	***F1 Score* ponderado (%)**	**Área sob a curva ROC (%)**
Naive Bayes	**51,0**	58,6	51,2	51,8	86,9
Árvore de decisão	36,8	52,2	58,5	54,3	77,8
Regressão logística	40,7	54,0	56,1	54,7	90,5

O algoritmo que demostrou o melhor desempenho foi o Naive Bayes, com uma acurácia de aproximadamente 51%, com os melhores hiperpâmetros selecionados pelo `GridSearchCV`, relacionados na Tabela 5. As melhores métricas obtidas através da validação cruzada para este modelo se encontram na Tabela 6.

Ao analisar as outras métricas, relacionadas na Tabela 4, é possível identificar que o algoritmo de árvore de decisão foi superior aos outros, do ponto de vista da métrica de *recall*, enquanto o algoritmo de regressão logística foi superior ao observar o *F1 Score*. Tendo em mente que o objetivo é classificar os tópicos, a escolha do algoritmo de árvore de decisão não seria uma má definição, de acordo com o *recall*. Por exemplo, ao classificar algumas mensagens dentre os tópicos de ITCD, como o tópico de informações gerais de

Tabela 5. Melhores hiperparâmetros identificados pelo `GridSearchCV` para o Naive Bayes

Hiperparâmetro	Valor
alpha	0,1
class_prior	None
fit_prior	False

Tabela 6. Melhores métricas obtidas pelo Naive Bayes durante a validação cruzada

Acurácia balanceada (%)	Precisão ponderada (%)	*Recall* ponderado (%)	*F1 Score* ponderado (%)	Área sob a curva ROC (%)
45,4	60,0	54,6	54,5	89,1

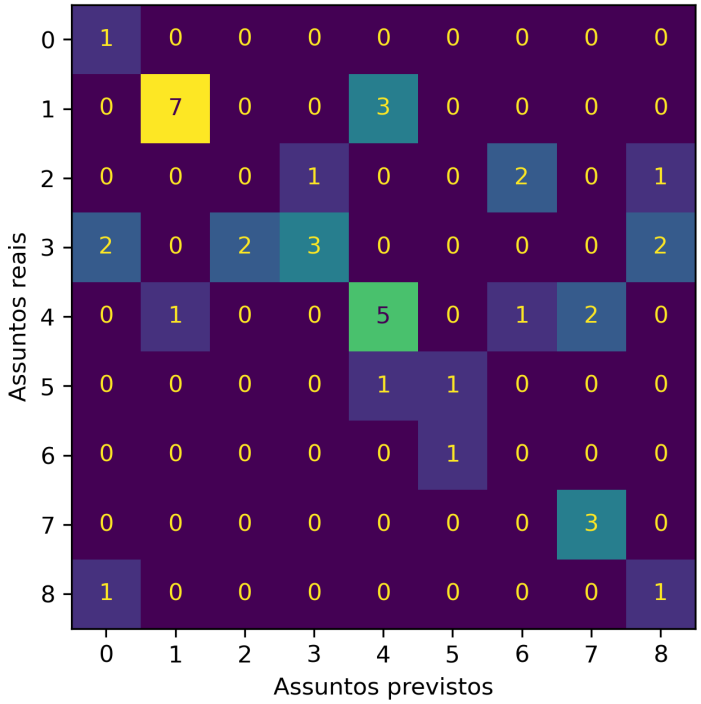
ITCD, ainda seria aceitável como previsão, tendo em vista que informações gerais pode tratar assuntos abrangentes.

Uma acurácia balanceada de no mínimo 50,0% significa que o modelo é semelhante a um classificador aleatório em problemas de classificação binária. Desse modo, no presente cenário com 9 categorias, um contexto semelhante de aleatoriedade seria caracterizado em casos de acurácias balanceadas próximo a 11,1%. Sendo assim, todos os modelos apresentam um desempenho um pouco melhor que classificadores aleatórios, com o Naive Bayes o melhor posicionado entre eles, porém para todos os modelos ainda há espaço para melhorias.

Ao analisar a matriz de confusão na Figura 3, gerada a partir da classificação realizada pelo algoritmo de Naive Bayes, é possível identificar que poucas vezes o modelo confundiu o tópico ITCD com IPVA. Isso pode sugerir que o modelo está conseguindo identificar alguns padrões entre essas duas grandes categorias e esse comportamento também pode estar correlacionado a distribuição dos dados identificados entre eles. No caso, agrupando todos os assuntos referentes ao ITCD é somado 85 documentos, enquanto documentos voltados ao IPVA agregam 117.

Em suma, a Área sob a curva ROC em todos os modelos sugere que eles possuem uma capacidade aceitável em distinguir entre as classes de um modo geral, porém, essa circunstância pode estar sendo causada pelo desiquilíbrio entre as classes, devido a área sob a curva ROC calculada considerar o peso delas. Porém, as outras métricas com valores relativamente abaixo do esperado, podem indicar que os modelos utilizados possivelmente possuem problemas em lidar com o conjunto de dados desbalanceados, mesmo que o algoritmo de Naive Bayes possua características de assumir independência condicional entre os atributos. É possível notar isso através, também, da matriz de confusão da Figura 3, onde, de maneira geral, o classificador utilizando Naive Bayes saiu-se bem nas classes majoritárias comparativamente as outras minoritárias. Além disso, o tamanho relativamente pequeno de 202 documentos do primeiro *corpus* pode ter limitado o aprendizado dos modelos e dificultado a identificação de padrões entre as classes presentes de maneira mais assertiva, principalmente aquelas com poucos exemplos para representá-las.

De todo modo, o modelo com o melhor desempenho, dado as condições estabe-



0 - Abertura processo ITCD
1 - Guia pagamento IPVA
2 - Guia pagamento ITCD
3 - Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais
4 - Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais
5 - Isenção de IPVA
6 - Isenção de ITCD
7 - Legislação Tributária - IPVA
8 - Legislação Tributária - ITCD

Figura 3. Matriz de confusão Naive Bayes

lecidas, foi o com Naive Bayes e baseado nele foi realizado a classificação do segundo *corpus*, com o objetivo de entender quais seriam os padrões que o modelo exibiria e identificar alguns componentes que geraram as classificações realizadas. No segundo *corpus*, o resultado da classificação é apresentado na Tabela 7, onde é possível visualizar a distribuição dos documentos baseado no assunto.

Detalhando melhor o resultado da classificação do segundo *corpus*, a Tabela 8, nos anexos, demonstra a categoria original e a categoria prevista, que são os assuntos originais e previsto, respectivamente. Dessa forma, destacam-se que 39,5% da categoria Legislação Tributária - ITCD foi reclassificada como Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais. A categoria Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais manteve, do seu volume original, 28,6% dos documentos. No assunto Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais, o segundo maior percentual de mudança em todo o segundo *corpus*, teve reclassificado para Guia pagamento IPVA 46,3% do seu volume original. Legislação Tributária - IPVA teve 44,2% dos seus documentos modificados para Guia pagamento IPVA. Arrecadação - ITCD, com maior percentual de mudança, contou com

Tabela 7. Distribuição do segundo *corpus* por assuntos, após a classificação com o modelo desempenho superior

Assunto	Quantidade
Isenção de ITCD	31
Abertura processo ITCD	86
Legislação Tributária - ITCD	147
Isenção de IPVA	211
Legislação Tributária - IPVA	235
Guia pagamento ITCD	209
Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais	659
Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais	442
Guia pagamento IPVA	864

52,5% dos documentos reclassificados em Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais. Em Isenção de IPVA 34,3% foram reclassificados para Guia pagamento IPVA e Isenção de ITCD com 45,9% dos documentos reclassificados como Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais.

Supondo que os assuntos das mensagens de fato representavam adequadamente a categoria dos documentos, alguns casos chamam ainda mais atenção por divergirem entre o assunto original e o predito pelo modelo. É o caso da assunto Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais em que 15,1% dos documentos foram reclassificados como Legislação Tributária - IPVA, onde as duas categorias são bem distintas. Ao extrair os atributos que obtiveram a maior probabilidade para levar a classificação, identificamos os termos *parcel* com 1,68%, *atras* com 1,39%, *ipv* com 1,38%, *nom* 1,32% e *carr* com 1,15%. É possível identificar que o termo *ipv* tem uma das maiores probabilidades encontradas no subconjunto. Ao utilizar uma nuvem de palavras baseada nas mensagens do subconjunto, podemos visualizar na Figura 4 que o termo IPVA ocorre com uma grande frequência, juntamente com pagar, moto, entre outros termos que rementem a categorias de assunto relacionados a IPVA.

Dentro desse subconjunto, entretanto, algumas mensagens de fato podem não pertencer a ele. Por exemplo, a mensagem "Boa tarde queria saber qual é o motivo da divida ativa do veiculo", que foi classificada como Legislação Tributária - IPVA, pode ter uma resposta simples referente as pendências que o veículo possui. Em contrapartida, a mensagem "Prezados Bom dia poderiam me ajudar sanando uma dúvida por gentileza? Estou com um caso de uma moto que está com licenciamento e IPVA atrasados desde moto completou XX anos e possui isenção da taxa de IPVA, logo não tenho como pagar a taxa. O governador sancionou uma lei para perdoar dívidas de outros anos, mas na lei deixa claro que devesse pagar o IPVA, mas a moto está isenta dessa taxa em XXXX. Gostaria de saber as dívidas dos anos anteriores serão perdoadas com a moto sendo isenta de IPVA devido a sua idade. Estou com receio de pagar licenciamento, não pagar IPVA e ser pego com alguma irregularidade", pode ser necessário de um conhecimento mais especializado referente a lei para respondê-la, reforçando a previsão como correta. Mas de modo ge-

Figura 4. Nuvem de palavras das mensagens da categoria Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais reclassificadas como Legislação Tributária - IPVA

ral, ambas as mensagens não caracterizam o assunto original, voltado a ITCD, o que está também de acordo com o analisado até aqui.

O assunto originalmente Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais também tem uma relação semelhante. Entretanto, 46,3% dessa categoria foi reclassificada como Guia pagamento IPVA, uma categoria um pouco mais próxima que o exemplo anterior. Os atributos com as maiores probabilidades, que influenciaram essa classificação, foram os termos: *pag* (3,82%), *bolet* (3,77%), *ipv* (3,49%), *veicul* (2,67%) e *ren* (2,61%). O termo *ipv* nessa categoria tem uma probabilidade maior que o encontrado no assunto Legislação Tributária - IPVA, possivelmente devido a diferença no volume de dados. A nuvem de palavras ilustrada na Figura 5, dessa vez, relaciona palavras que de fato podem ser comuns aos dois assuntos. Entretanto, é possível destacar que, além da palavra IPVA, a palavra boleto, imprimir e outras são destacadas, sugerindo que nesse subgrupo são bastante relevantes. Isso pode ser verdade, tendo em vista que as mensagens foram reclassificadas para Guia pagamento IPVA.

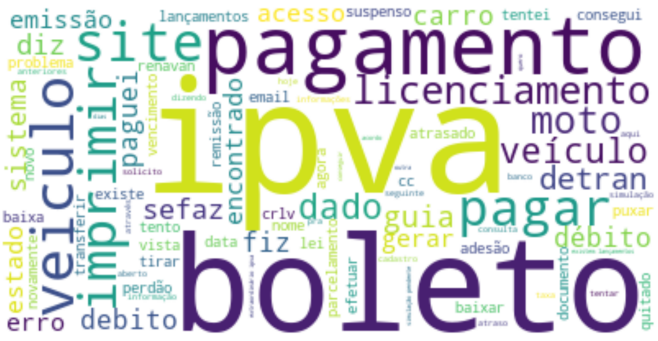


Figura 5. Nuvem de palavras das mensagens da categoria Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais reclassificadas como Guia pagamento IPVA

Devido a proximidade entre a classe original e a prevista, é esperado que nesse subconjunto seja encontrado com mais facilidade mensagens que poderiam pertencer a categoria original, ou que cause um pouco de confusão ou dúvida com relação a sua classificação. A exemplo disso, a mensagem “Paguei um IPVA indevidamente pois entrei

no portal da SEFAZ da Paraíba achando que estava no portal da SEFAZ do Ceará. Comprei um carro no Ceará e preciso pagar no Ceará, como faço para estornar esse pagamento que fiz erroneamente a SEFAZ da Paraíba? No aguardo" abre um leque de possibilidades, dando entender que ela poderia ter sido classificada até como informações gerais. Entretanto, a mensagem "Bom dia Venho por meio deste solicitar emissão e envio do IPVA. Dados do veículo" classificada também como Guia Pagamento IPVA já é mais direta e ratificando que algumas mensagens do assunto original de fato faziam parte de outro grupo.

Uma última análise a ser feita é a reclassificação do assunto Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais para ele mesmo. 28,6% dos documentos desse assunto continuaram na mesma categoria. Nesse cenário, o atributo com maior probabilidade que contribui para esse classificação é o termo *itcd*, com 2,34%, seguido por *document* (1,35%), *declar* (1,23%) e *cadastr* (1,16%). Nesse contexto, a nuvem de palavras gerada desse subconjunto, ilustrada na Figura 6, destaca justamente a palavra itcd. Porém, é possível ver que muitas das mensagens possuíam a palavra guia, inventário e processo. De maneira geral, não seria um problema que outros assuntos relacionados a ITCD fossem classificados nessa categoria, tendo em vista que é um assunto de informações gerais.

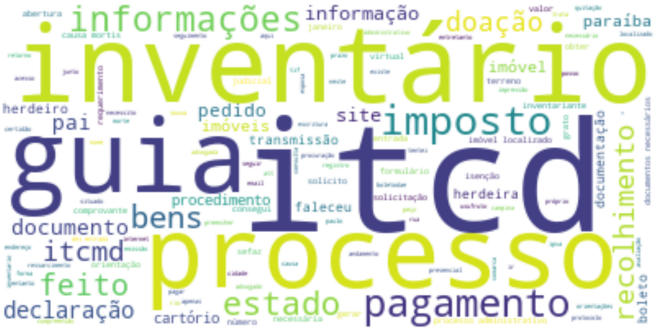


Figura 6. Nuvem de palavras das mensagens da categoria Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais reclassificadas como Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais

Nesse cenário, é possível observar mensagens que mantiveram a classe incorreta. Por exemplo, a mensagem "Gostaria de emitir guia para pagamento de ITCD causa mortis relativo a imóvel localizado em Lucena PB solicito orientações", poderiam se encaixar no assunto Guia Pagamento - ITCD, tendo em vista que o autor solicita justamente a guia de pagamento do imposto. Porém, como ressaltado, o assunto sob análise busca tratar os temas mais gerais, podendo casos como esse serem aceitáveis, mas não ideais.

Por fim, vale reforçar que os resultados obtidos aqui demonstram que ainda há o que melhorar no processo de classificação dos assuntos das mensagens do canal "Fale Conosco" da SEFAZ-PB. Todos os algoritmos demostraram a capacidade em distinguir as classes, de acordo com a área sob a curva ROC, de maneira aceitável. Ao analisar alguns exemplos, utilizando o segundo *corpus*, que não tinha sido visto pelo modelo, também é possível identificar que o classificador possui uma capacidade de distinguir entre as classes e, principalmente, encontrar padrões entre ITCD e IPVA em termos mais gerais. Porém, as outras métricas e principalmente a acurácia balanceada, indicam que

os modelos precisam melhorar e que o desbalanceamento das classes pode ter sido um problema para os modelos.

5. Conclusão

Nesta pesquisa buscamos classificar as mensagens enviadas ao "Fale Conosco" da SEFAZ-PB de acordo com os assuntos já disponíveis no canal de atendimento, utilizando para isso três algoritmos de classificação. Dentre os algoritmos comparados, aquele que obteve um melhor desempenho foi o Naive Bayes. Porém, mesmo com o melhor desempenho entre eles, não se pode afirmar que, dentro das condições da pesquisa, foi possível obter os melhores resultados com o modelo.

Como possíveis melhorias, em primeiro lugar, a utilização de um conjunto de dados maior para avaliar a viabilidade de um melhor aprendizado pelos algoritmos. Isto pode ser possível através da análise e categorização manual do segundo *corpus* pelos auditores, podendo ser executada com base na reclassificação realizada pelo modelo baseado no Naive Bayes. Em seguida, buscar equilibrar o *corpus* utilizado, através de métodos de geração de dados sintéticos, ou buscando mais informações com a SEFAZ-PB. Analisar se o uso do TF-IDF como método de vetorização é o mais adequado para o cenário, tendo em vista que, de acordo com a fórmula do TF-IDF, *tokens* comuns ao *corpus* tendem a ter um valor menor, entretanto tais termos podem ser úteis na definição dos tópicos estudados, por exemplo: IPVA e ITCD. Por fim, durante o processo de redução de palavras, é importante realizar testes com outros algoritmos no pré-processamento, por exemplo, diminuindo a chance de palavras com sentidos diferentes serem reduzidas a um mesmo *token*, a título de exemplo, usar o processo de lematização implementado pela biblioteca *spaCy*, onde é possível tratar palavras em português.

Referências

Aggarwal, C. and Zhai, C. (2012). *Mining Text Data*. Springer New York.

Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13:281–305.

Cortes, E., Woloszyn, V., Binder, A., Himmelsbach, T., Barone, D., and Möller, S. (2020). An empirical comparison of question classification methods for question answering systems. In Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., and Piperidis, S., editors, *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5408–5416, Marseille, France. European Language Resources Association.

Dieng, A. B., Wang, C., Gao, J., and Paisley, J. (2017). Topicrnn: A recurrent neural network with long-range semantic dependency.

Jarrett, C., Gaffney, G., and Krug, S. (2009). *Forms that Work: Designing Web Forms for Usability*. Interactive Technologies. Elsevier Science.

Jiang, M., Liang, Y., Feng, X., Fan, X., Pei, Z., Xue, Y., and Guan, R. (2018). Text classification based on deep belief network and softmax regression. *Neural Computing and Applications*, 29:61 – 70.

Jurafsky, D. and Martin, J. H. (2019). *Speech and Language Processing*. Pearson, Upper Saddle River, NJ, 3rd edition.

Kowsari, Meimandi, J., Heidarysafa, Mendu, Barnes, and Brown (2019). Text classification algorithms: A survey. *Information*, 10(4):150.

Loni, B. (2011). A survey of state-of-the-art methods on question classification.

McEnery, T., Xiao, R., and Tono, Y. (2006). *Corpus-Based Language Studies: An Advanced Resource Book*. Routledge, London, U.K.

Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., and Gao, J. (2020). Deep learning based text classification: A comprehensive review. *CoRR*, abs/2004.03705.

Mohammed, M. and Omar, N. (2020). Question classification based on bloom's taxonomy cognitive domain using modified tf-idf and word2vec. *PLOS ONE*, 15(3):1–21.

Müller, A. C. and Guido, S. (2016). *Introduction to Machine Learning with Python: A Guide for Data Scientists*. O'Reilly Media, Inc.

Orengo, V. and Huyck, C. (2001). A stemming algorithm for the portuguese language. In *Proceedings Eighth Symposium on String Processing and Information Retrieval*, pages 186–193.

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.

Sangodiah, A., Muniandy, M., and Heng, L. E. (2015). Question classification using statistical approach: A complete review. *Journal of Theo*.

Singh, J. and Gupta, V. (2016). Text stemming: Approaches, applications, and challenges. *ACM Comput. Surv.*, 49(3).

Speer, R. (2019). ftfy. Zenodo. Version 5.5.

6. Anexos

Tabela 8. Distribuição do segundo *corpus* por assuntos originais e assuntos preditos

Assunto original	Assunto predito	Quantidade
Legislação Tributária - ITCD	Abertura processo ITCD	8
	Guia pagamento IPVA	2
	Guia pagamento ITCD	26
	Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais	32
	Isenção de IPVA	1
	Isenção de ITCD	4
	Legislação Tributária - ITCD	8
Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais	Abertura processo ITCD	9
	Guia pagamento IPVA	25
	Guia pagamento ITCD	44
	Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais	87
	Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais	33
	Isenção de IPVA	25
	Isenção de ITCD	4
	Legislação Tributária - IPVA	46
	Legislação Tributária - ITCD	31
Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais	Abertura processo ITCD	8
	Guia pagamento IPVA	436
	Guia pagamento ITCD	16
	Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais	6
	Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais	347
	Isenção de IPVA	28
	Isenção de ITCD	5
	Legislação Tributária - IPVA	88
	Legislação Tributária - ITCD	7
Legislação Tributária - IPVA	Abertura processo ITCD	1
	Guia pagamento IPVA	256

Assunto original	Assunto predito	Quantidade
	Guia pagamento ITCD	9
	Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais	7
	Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais	183
	Isenção de IPVA	37
	Isenção de ITCD	3
	Legislação Tributária - IPVA	70
	Legislação Tributária - ITCD	13
Arrecadação - ITCD	Abertura processo ITCD	49
	Guia pagamento IPVA	5
	Guia pagamento ITCD	110
	Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais	282
	Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais	7
	Isenção de ITCD	4
	Legislação Tributária - IPVA	1
	Legislação Tributária - ITCD	79
Isenção de IPVA	Abertura processo ITCD	7
	Guia pagamento IPVA	139
	Guia pagamento ITCD	2
	Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais	11
	Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais	88
	Isenção de IPVA	115
	Isenção de ITCD	9
	Legislação Tributária - IPVA	30
	Legislação Tributária - ITCD	4
Isenção de ITCD	Abertura processo ITCD	4
	Guia pagamento IPVA	1
	Guia pagamento ITCD	2
	Imposto de Transmissão de Causa Mortis e Doação-ITCD - Informações Gerais	17
	Imposto sobre Propriedade de Veículos Automotores-IPVA - Informações Gerais	1

Assunto original	Assunto predito	Quantidade
	Isenção de IPVA	5
	Isenção de ITCD	2
	Legislação Tributária - ITCD	5