Modelos de Aprendizagem de Máquina para Identificação de Mercadorias a partir da Descrição do Item na Nota Fiscal Eletrônica

Alisson Emanuel Goes de Mendonça[1,2], Luciano Reis Coutinho[1], Roberval Gomes Mariano[2]

[1]Departamento de Informática/CCET – Universidade Federal do Maranhão (UFMA)
São Luís – MA – Brasil

[2]Secretaria de Estado da Fazenda do Maranhão (SEFAZ)
São Luís – MA – Brasil

alisson.mendonca@discente.ufma.br, luciano.rc@ufma.br, mariano@sefaz.ma.gov.br

Abstract. *One of the issues that impose limitations on tax actions in Notas Fiscais Eletrônicas is to be able to identify the goods from the item description field, both for the variety of content, since it is free to fill, and for the volume of information generated. This article presents three Machine Learning models applied in processing this field: Decision Tree, Neural Network and KNN. The experiments demonstrated that the Neural Network delivered the best accuracy and the lowest cost of prediction time. The data used are from the SEFAZ/MA. The collected results can guide solutions for detecting under-invoicing, improper exemptions, correction of the inflation lag of the tax tariff table, etc.*

Resumo. *Uma dos aspectos limitantes de ações fiscais nas Notas Fiscais Eletrônicas é conseguir identificar a mercadoria a partir do campo de descrição do item, tanto pela variedade do conteúdo, dado que é de livre preenchimento, como pelo volume de informações gerado. Este artigo apresenta três modelos de Aprendizagem de Máquina aplicados no processamento desse campo: Árvore de Decisão, Rede Neural e KNN. Os experimentos demonstram que a Rede Neural entregou a melhor acurácia com o menor tempo de predição. Os dados utilizados são da base da SEFAZ/MA. Os resultados podem direcionar soluções de detecção de subfaturamento, isenções indevidas, correção da defasagem inflacionária da pauta fiscal, etc.*

1. Introdução

A Administração Tributária viabiliza o financiamento das atividades do Estado para com a sociedade e está inserida no âmbito de todos os entes políticos: União, Estados/DF e municípios. No âmbito da competência tributária estadual, as fontes de recursos são os impostos, as taxas e contribuições de melhoria, nos termos do Art. 145 da Constituição da República Federativa do Brasil. Entretanto, em termos de representatividade na arrecadação, a principal fonte é o Imposto Sobre Operações Relativas à Circulação de Mercadorias e Sobre Prestações de Serviços de Transporte Interestadual, Intermunicipal e de Comunicação – ICMS. Em algumas unidades da federação, como no caso do

Maranhão, esse imposto compõe mais de 90% da arrecadação mensal (SEFAZ/MA, 2011).

A Nota Fiscal eletrônica (NF-e) é o principal insumo na fiscalização do ICMS, de forma que a sua correta emissão possibilita aos fiscos estaduais monitorar e cobrar o imposto devido. A legislação tributária impõe ao contribuinte a obrigação preencher as notas fiscais com informações relativas à operação praticada. Parte dessas informações são estruturadas, seguem padrões estabelecidos de acordo com a mercadoria. No entanto, há campos de livre preenchimento pelo contribuinte, como é o caso do campo descritivo do produto na NF-e. Para estes campos, de livre preenchimento, cada contribuinte pode adotar uma forma própria para descrever uma mesma mercadoria, resultando em uma grande variedade de possibilidades de representação no documento fiscal emitido, o que inviabiliza o processamento deste conteúdo em verificações automáticas realizadas por malha fiscal, que fica limitada a inspeção dos campos estruturados. Também não é possível submeter a descrição de cada item à prévia análise humana, pois o volume diário médio de notas fiscais emitidas é elevado – no Estado do Maranhão é de 900 mil – e cada uma pode abarcar até 990 mercadorias (itens). Nessas circunstâncias, um contribuinte pode manipular as informações de campos estruturados e não estruturados para não representar adequadamente a mercadoria, ou sua quantidade, fazendo com que a tributação ocorra em uma base de cálculo inferior a devida e dificultando a checagem por malhas fiscais.

Recentemente, os autores Jatobá et al. (2021) propuseram uma ferramenta que realiza um certo nível de automatização do cálculo do montante de ICMS nas operações interestaduais destinadas ao Estado de Alagoas, com o apoio da aprendizagem de máquina (supervisionada e não-supervisionada) para classificação das notas fiscais eletrônicas. Também pode-se citar o trabalho dos autores Galdino, Silveira e Fonseca (2004), que utilizaram uma Rede Neural para classificar empresas do Estado de Minas Gerais de acordo com indicadores fiscais individuais e setoriais. A Rede Neural também já foi utilizada para previsão de receitas de ICMS do Estado do Espírito Santo, conforme artigo dos autores Carmo, Boldt e Komati (2019). Um outro trabalho relacionado é o do autor Madeira (2015) que também utilizou-se dessas técnicas de aprendizagem para detectar irregularidades na Nota Fiscal de Serviços Eletrônica – NFSe do município do Rio de Janeiro, neste caso, o autor relata o processamento do campo texto que descreve o serviço prestado, também de livre preenchimento pelo emissor do documento fiscal.

Nesse contexto de iniciativas que utilizam os recursos da Aprendizagem de Máquina para auxiliar a ação fiscal dos órgãos fazendários, o objetivo desse trabalho é apresentar um estudo comparativo de modelos de classificação para identificar as mercadorias declaradas em uma NF-e a partir do campo descritivo, buscando o melhor resultado de acurácia preditiva com o menor custo de tempo de predição, de forma a mitigar a intervenção humana na análise das descrições dos itens e a conseguir processar todo o volume de documentos fiscais em um fluxo automatizado de cálculo do tributo. A experimentação realizada se baseou em um recorte da pauta fiscal vigente para o Estado do Maranhão, do segmento Cervejas e Chopes, de onde foram extraídas 21 possibilidades de classes (mercadorias), as quais contemplam cervejas de diferentes fabricantes, bem como cerveja de um mesmo fabricante, alterando apenas a embalagem (ex. garrafa, lata 350ml, lata 269ml, long neck). Um modelo de classificação assertivo pode auxiliar a fiscalização a incrementar a arrecadação de diversas formas, a Secretaria de Estado da Fazenda do Maranhão priorizou as seguintes: identificação de operações subfaturadas;

verificação do valor da base de cálculo do ICMS de acordo com a pauta fiscal, tabela de órgão regulador ou tabela de preços sugeridos pelo fabricante; e atualização da pauta fiscal das mercadorias para corrigir a defasagem inflacionária.

O restante deste trabalho está organizado em 3 seções. A seção 2 descreve a estratégia utilizada na coleta e preparação dos dados a serem minerados. A seção 3, os resultados obtidos na classificação dos itens a partir de 3 modelos de aprendizagem: Árvore de Decisão, Rede Neural e *K-Nearest Neighbors – KNN*. Por fim, a seção 4 apresenta as considerações finais sobre o trabalho realizado.

2. Coleta e Preparação dos Dados

O conjunto de dados que servirá ao tratamento descrito neste trabalho foi extraído do Data Warehouse da Secretaria de Estado da Fazenda do Maranhão. Com intuito de atender os requisitos de sigilo fiscal, não foram extraídas quaisquer informações que possam identificar as partes envolvidas na operação. Registre-se ainda que as informações ocultadas na extração não são relevantes para o processo de classificação que se pretende nesse trabalho. O foco foi o aprendizado a partir do campo descritivo do produto, informação localizada no campo xProd do XML da Nota Fiscal (MOC, 2020).

Os dados foram extraídos a partir de uma delimitação temporal restringindo a amostra às Notas Fiscais emitidas no mês de setembro de 2021. Como o objeto central do trabalho é classificar o item da nota dentre uma das possibilidades do grupo de Cervejas e Chopes da pauta fiscal, utilizou como um filtro adicional o critério de possuir o código segundo a tabela de Nomenclatura Comum do Mercosul - NCM 2203.00.00, para a descrição de produtos que se enquadra em cervejas de malte (Receita Federal do Brasil, 2021).

O conjunto de dados trabalhado possui um total de 28.843 exemplos descritivos de itens de documentos. A distribuição dos exemplos para cada alvo pode ser observada na Figura 1.

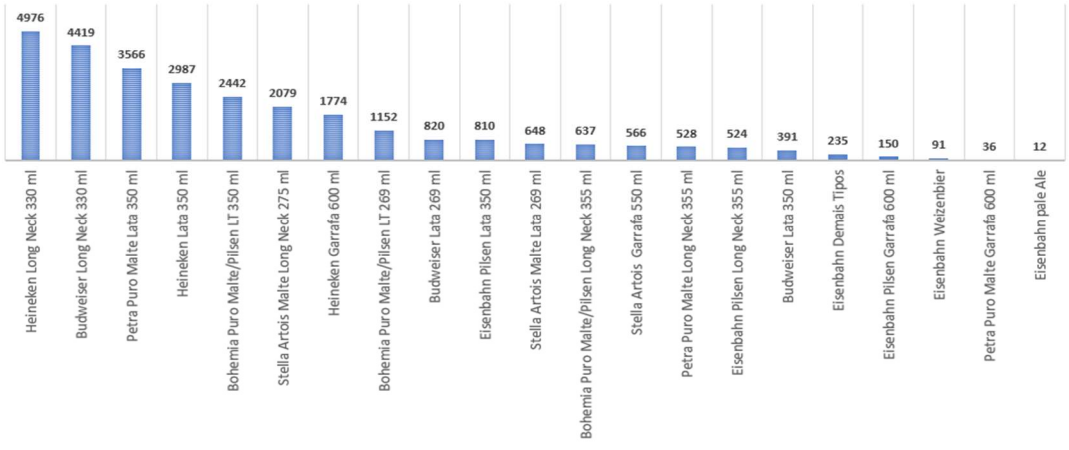

Figura 1. Distribuição dos exemplos descritivos de acordo com o item correspondente da Pauta Fiscal

Naturalmente, há repetições de uma mesma descrição pois um determinado contribuinte faz mais de uma venda por mês. Isso pode ser observado na Figura 2, que representa a distribuição em cada alvo de um total de 1.327 exemplos distintos de descrições de produtos.

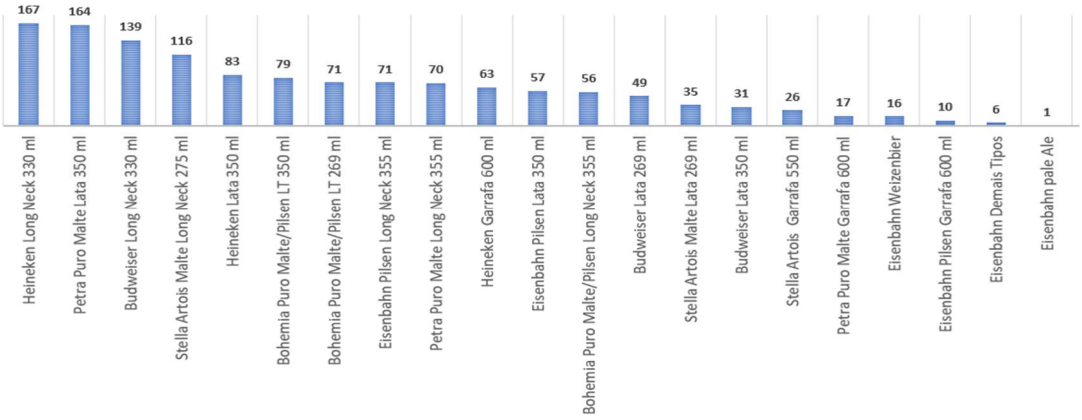

Figura 2. Distribuição dos exemplos descritivos distintos de acordo com o item correspondente da Pauta Fiscal

O objetivo central do aprendizado é fazer a classificação dos itens da Nota Fiscal no correspondente da pauta fiscal a partir da descrição informada pelo contribuinte no campo xProd. No entanto, há um problema prévio que é transformar esse conteúdo textual em uma estrutura que sirva como entrada para um modelo de aprendizado. Para tal feito, recorreu-se à segmentação representando-o em um vetor numérico (Chollet, 2017). De modo mais específico, foi utilizado *Word-level one-hot encoding*. Ela consiste em considerar cada palavra de todas as descrições dos exemplos de treino como um *token*. Cada *token* está vinculado a um índice no vetor, consequentemente, o vetor tem um tamanho equivalente a quantidade de tokens. Havendo a existência do token em cada descrição, o valor do respectivo índice no vetor é assinalado com o valor "1". Dessa forma, teremos um vetor representando o texto de cada exemplo.

Nos experimentos foi utilizado o componente Tokenizer da biblioteca Keras, que já implementa a lógica para gerar a representação do texto em um vetor. Também já disponibiliza o método para gerar uma matriz com os valores que representam cada um dos textos dos exemplos (Developer Guides Keras).

A aplicação da estratégia *Word-level one-hot encoding* pelo componente Tokenizer ao conjunto de dados objeto desse trabalho encontrou 257 tokens únicos que, quando exportados para uma matriz, gerou uma tabela com 258 colunas. Ao final, essa matriz foi transformada em um DataFrame, componente da biblioteca Pandas, e, ao DataFrame, foi adicionada uma coluna com as anotações que representam efetivamente o item da pauta fiscal correspondente a cada representação vetorizada. Os dados foram divididos de forma que 60% do conjunto de dados (o equivalente a 17.305 exemplos) foi utilizado como treino e 40% (o equivalente a 11.538 exemplos) foi utilizado como teste.

3. Modelos de Aprendizagem Aplicados

Para fazer classificação dos itens da NF-e a partir da descrição, utilizaremos a abordagem supervisionada, a qual tenta descobrir as relações entre atributos de entrada (algumas vezes chamados de variáveis independentes) e o atributo alvo (também chamado de variável dependente). As relações identificadas são representadas em uma estrutura a qual chamamos de Modelo, que geralmente descreve e explica um fenômeno oculto no conjunto de dados e pode ser utilizado para propor o valor do atributo alvo sempre que os atributos de entrada forem conhecidos (Rokach; Maimon, 2014). De modo específico, os

modelos criados foram: Árvore de Decisão, subseção 3.1; Rede Neural MLP, subseção 3.2; e *K-Nearest Neighbors – KNN*, subseção 3.3.

3.1. Árvore de Decisão (Decision Tree)

Nesta subseção, utilizaremos o modelo Árvore de Decisão, que usa uma estrutura de árvore para representar um número de possíveis caminhos de decisão e um resultado para cada caminho (Grus, 2016). Considerando o objetivo pretendido, a métrica utilizada para o Modelo de Árvore de Decisão foi obtida a partir de um histórico relacionando a acurácia com o tamanho da árvore criada para representar o aprendizado. Também foram catalogados os valores absolutos de acertos e erros.

Para isso, fez-se a repetição do processo de aprendizado e de validação, variando-se o parâmetro que limita a profundidade da árvore. O componente utilizado para a realização do aprendizado utiliza do algorítmo de Árvore CART (Classification and Regression Trees) que constrói árvores binárias usando *feature* e *threshold* que geram o maior ganho de informação em cada nó (User Guide Scikit Learn).

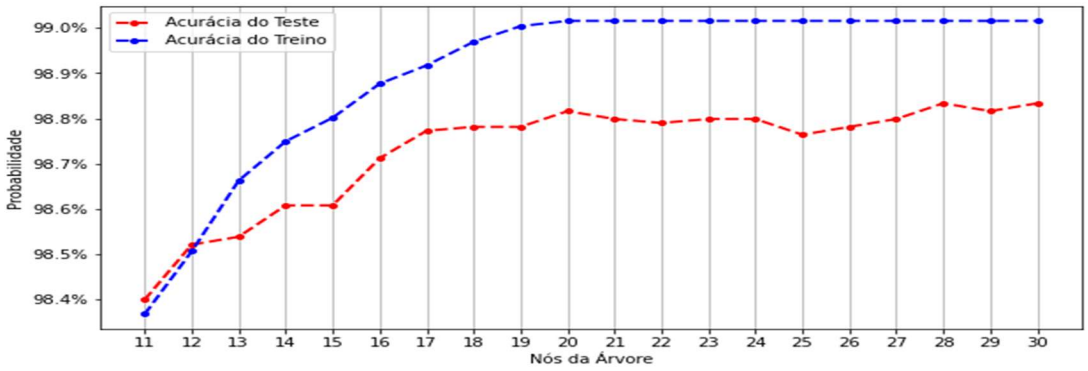

Figura 3. Resultados de Acurácia do Modelo baseado em Árvore de Decisão

A partir da Figura 3 e da Tabela 1, constata-se que após a profundidade de 12 nós, a evolução da acurácia de treino se descola da acurácia de teste, indicando que o aprendizado está começando a restar prejudicado. Após o tamanho 17, observa-se o efeito do *overfitting*, qual seja, a acurácia de testes apresenta uma distanciação com tendência crescente em relação a de treino. No entanto, a acurácia atinge um valor máximo de 98,83% para uma profundidade de 28 nós.

O tempo de predição médio foi de 0,073 segundos para a classificação dos 11.538 itens da massa de teste. Percebe-se que o desvio padrão foi de apenas 0,011 segundos, o que indica que a profundidade não tem efeito negativo em relação a performance.

Tabela 1. Resultados de Acurácia e Tempo de Predição do Modelo baseado em Árvode de Decisão

Profundidade da Árvore	Acurácia do Teste	Tempo de Predição (Segundos)
1	32,61%	0,066
2	55,09%	0,065
3	71,64%	0,1
4	82,53%	0,058
5	88,75%	0,068
6	93,37%	0,064
7	96,33%	0,059
8	96,97%	0,072
9	97,26%	0,07
10	97,85%	0,074
11	98,40%	0,07
12	98,52%	0,074
13	98,54%	0,069
14	98,60%	0,075
15	98,60%	0,067
16	98,71%	0,074
17	98,77%	0,074
18	98,78%	0,066
19	98,78%	0,068
20	98,81%	0,076
21	98,80%	0,064
22	98,79%	0,076
23	98,80%	0,086
24	98,80%	0,104
25	98,76%	0,081
26	98,78%	0,064
27	98,80%	0,065
28	98,83%	0,085
29	98,81%	0,095
30	98,83%	0,07

3.2. Rede Neural

Nesta subseção, utilizaremos o modelo Rede Neural, descrito como uma função matemática que mapeia uma dada entrada em uma saída desejada (Loy, 2019), essa estrutura matemática, também chamada de *Perceptron*, pode ser estruturada em uma arquitetura *multilayer perceptron* - *MLP*, que define: uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída (Bhasin, 2021). Considerando o objetivo pretendido, as métricas utilizadas para o Modelo de Rede Neural foram obtidas a partir de um histórico relacionando as variáveis: acurácia com a variável que representa a quantidade de neurônios na camada oculta, fixando-se o número de épocas; e, acurácia com a variável que representa a quantidade de épocas, fixando-se o número de neurônios na camada intermediária. Em ambos os casos, a avaliação executou com apenas uma camada oculta. O componente utilizado para a realização do aprendizado implementa o algorítmo *multi-layer perceptron* (MLP) que treina usando *Backpropagation* (User Guide Scikit Learn).

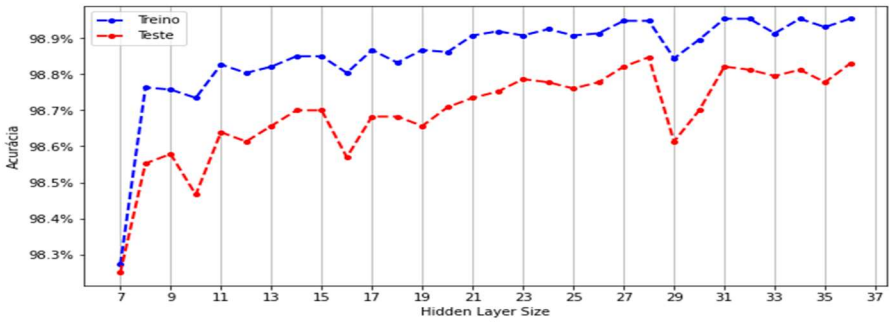

Figura 4. Resultados de Acurácia do Modelo baseado em Rede Neural – Variação da quantidade de neurônios

A partir da Figura 4 e da Tabela 2, constata-se que o modelo apresenta um valor de acurácia máximo de 98,85% com 28 neurônios na camada oculta.

O tempo de predição médio foi de 0,039 segundos para a classificação dos 11.538 itens da massa de teste. Percebe-se que o desvio padrão foi de apenas 0,008 segundos, o que indica que a variação da quantidade de neurônios não tem efeito negativo em relação a performance da predição.

Tabela 2. Resultados de Acurácia e Tempo de Predição do Modelo baseado em Rede Neural – Épocas constante em 15

Nº de Neurônios	Acurácia do Teste	Tempo de Predição (Segundos)
7	98,25%	0,031
8	98,55%	0,031
9	98,58%	0,031
10	98,47%	0,031
11	98,64%	0,031
12	98,61%	0,031
13	98,66%	0,047
14	98,70%	0,047
15	98,70%	0,047
16	98,57%	0,047
17	98,68%	0,031
18	98,68%	0,047
19	98,66%	0,031
20	98,71%	0,05
21	98,73%	0,031
22	98,75%	0,047
23	98,79%	0,031
24	98,78%	0,047
25	98,76%	0,031
26	98,78%	0,035
27	98,82%	0,047
28	98,85%	0,031
29	98,61%	0,031
30	98,70%	0,047
31	98,82%	0,048
32	98,81%	0,047
33	98,80%	0,047
34	98,81%	0,047
35	98,78%	0,047
36	98,83%	0,031

A seguir, estão representados os resultados obtidos a partir da variação do número épocas, mantendo-se constante em 28 o número de neurônios da camada oculta.

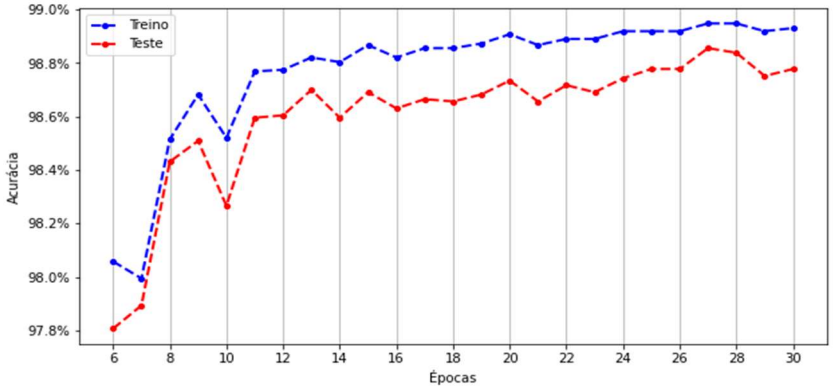

Figura 5. Resultados de Acurácia do Modelo baseado em Rede Neural – Variação do número de épocas

A partir da Figura 5 e da Tabela 3, constata-se que o modelo apresenta um valor de acurácia máximo de 98,86% para um número de épocas de 27.

O tempo de predição médio foi de 0,039 segundos para a classificação dos 11.538 itens da massa de teste. Percebe-se que o desvio padrão foi de apenas 0,007 segundos, o que indica que a variação da quantidade de neurônios não tem efeito negativo em relação a performance da predição.

Tabela 3. Resultados de Acurácia e Tempo de Predição do Modelo baseado em Rede Neural – Nº de Neurônios da Camada Oculta constante em 28

Nº de Épocas	Acurácia do Teste	Tempo de Predição (Segundos)
6	97,81%	0,045
7	97,89%	0,031
8	98,43%	0,047
9	98,51%	0,035
10	98,27%	0,035
11	98,60%	0,031
12	98,60%	0,031
13	98,70%	0,047
14	98,60%	0,031
15	98,69%	0,047
16	98,63%	0,031
17	98,67%	0,047
18	98,66%	0,047
19	98,68%	0,031
20	98,73%	0,031
21	98,66%	0,032
22	98,72%	0,048
23	98,69%	0,031
24	98,74%	0,047
25	98,78%	0,047
26	98,78%	0,047
27	98,86%	0,031
28	98,84%	0,047
29	98,75%	0,031
30	98,78%	0,047

3.3. K-Nearest Neighbors – KNN

Nesta subseção, utilizaremos o modelo *K-Nearest Neighbors – KNN*, que pertence a categoria de aprendizado baseado em instâncias. Neste caso, não há um modelo parametrizado, mas sim um rearranjo das amostras, a fim de acelerar consultas específicas. (Bonaccorso, 2019). Ou seja, quando um teste é demandado para o algoritmo fazer a predição, ele utilizará a maioria dos pontos próximos (*K-nearest points*) para determinar a classe correspondente (Johnston; Mathur, 2019). Considerando o objetivo pretendido, a métrica utilizada para o Modelo K-nn foi obtida a partir de um histórico relacionando a acurácia com a quantidade de vizinhos. O componente utilizado implementa três algorítmos: Brute Force, K-D Tree e Ball Tree (User Guide Scikit Learn). Na simulação retratada a seguir, o componente foi parametrizado para seleção automática.

Dada a característica inerente ao funcionamento de Lazy Classifiers, a massa utilizada para treino foi duplicada e, desta, foram excluídos os exemplos idênticos, restando 595 exemplos dos 17.305. Cabe ressaltar que apenas foram excluídos os exemplos duplicados, sendo correto afirmar que não há um exemplo dentre os 17.305 da base completa que não encontre um correspondente exato dentre os 595 da base reduzida.

Foram criados dois modelos de conhecimento para classificação. O primeiro foi aplicado à base de treino com a totalidade dos exemplos (que será referenciada como “Completa”), e o segundo, à base com exemplos distintos (que será referenciada como “Reduzida”).

Ainda, o procedimento descrito acima foi aplicado em dois contextos: no primeiro, a distância de cada vizinho não exerce influência (o peso, *weights*, é igual a *uniform*); no segundo, a distância de cada vizinho exerce influência (o peso, *weights*, é igual a *distance*).

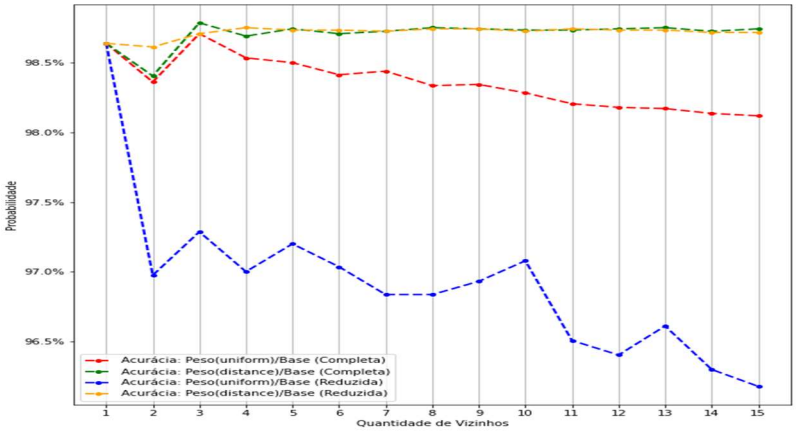

Figura 6. Resultados de Acurácia do Modelo baseado em KNN – Variação da quantidade de vizinhos

A partir da Figura 6 e da Tabela 4, restringindo-se inicialmente a análise à variação do parâmetro *weights*, observa-se que a tendência da acurácia é decrescente (linhas azul e vermelha) quando se descarta a influência da distância dos vizinhos (*weights* = *uniform*) à medida o número de vizinhos considerados na predição aumenta. Já no cenário em que a distância dos vizinhos é considerada (*weights* = *distance*), observa-se uma tendência de a acurácia permanecer constante ou com leve inclinação, indicando um efeito positivo na assertividade do modelo (linhas verde e amarela).

Analisando o efeito na acurácia do modelo de se variar o tamanho da base de treino, reduzindo de um total de 17.305 para 595, observa-se que essa estratégia catalisa a tendência de redução da acurácia do modelo quando a distância dos vizinhos não exerce influência na predição. E, quando essa distância exerce influência, percebe-se uma indiferença na acurácia do modelo.

Tabela 4. Resultados de Acurácia do Modelo baseado em KNN – Variação do número de Vizinhos

		Acurácia (%)			
		Base Completa		Base Reduzida	
		Peso Uniform	Peso Distance	Peso Uniform	Peso Distance
Quantidade de Vizinhos	1	98,64%	98,64%	**98,64%**	98,64%
	2	98,36%	98,41%	96,98%	98,61%
	3	**98,71%**	**98,79%**	97,29%	98,71%
	4	98,54%	98,69%	97,00%	**98,75%**
	5	98,50%	98,74%	97,20%	98,73%
	6	98,41%	98,71%	97,04%	98,73%
	7	98,44%	98,73%	96,84%	98,73%
	8	98,34%	98,75%	96,84%	98,74%
	9	98,34%	98,74%	96,93%	98,74%
	10	98,28%	98,73%	97,08%	98,73%
	11	98,21%	98,73%	96,51%	98,74%
	12	98,18%	98,74%	96,40%	98,73%
	13	98,17%	98,75%	96,61%	98,73%
	14	98,14%	98,73%	96,30%	98,72%
	15	98,12%	98,74%	96,18%	98,72%
	Média	98,36%	98,71%	96,92%	98,72%

Com intuito de enriquecer a análise, a seguir são retratados os resultados do tempo de resposta, uma vez que o modelo de classificação adota uma estratégia de postergar a criação de um modelo para o momento da predição, e isso pode afetar sua utilização a depender das expectativas em relação aos requisitos de performance e escala do problema a ser trabalhado.

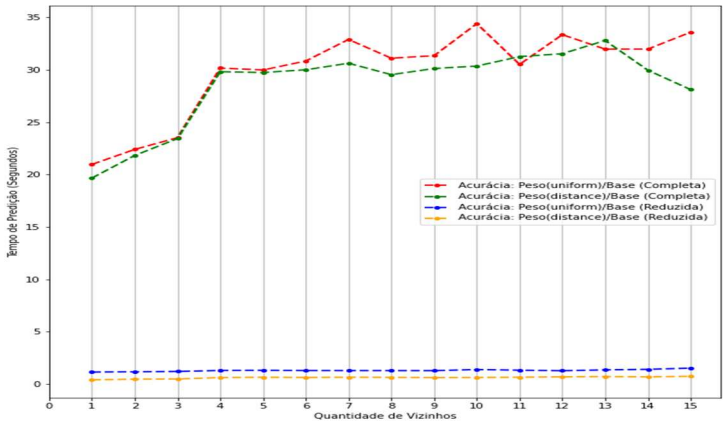

Figura 7. Resultados de Tempo de Predição do Modelo baseado em KNN – Variação da quantidade de vizinhos

Da análise da Figura 7, observa-se que o tempo do processo de classificação é afetado de forma relevante pelo tamanho da base de treino.

Também é possível observar na Tabela 5 que, variando-se o parâmetro *weights*, o processo de predição é mais demorado quando todos os vizinhos exercem a mesma influência na classificação, independentemente da distância.

Tabela 5. Resultados de Tempo de Predição do Modelo baseado em KNN – Variação do número de Vizinhos

		Tempo (Segundos)			
		Base Completa		Base Reduzida	
		Peso Uniform	Peso Distance	Peso Uniform	Peso Distance
Quantidade de Vizinhos	1	20,97	19,67	1,13	0,40
	2	22,40	21,82	1,16	0,46
	3	23,52	23,44	1,19	0,48
	4	30,18	29,83	1,29	0,61
	5	29,99	29,75	1,31	0,64
	6	30,85	30,01	1,28	0,62
	7	32,91	30,64	1,28	0,64
	8	31,11	29,54	1,28	0,63
	9	31,35	30,13	1,27	0,61
	10	34,42	30,37	1,38	0,62
	11	30,53	31,27	1,31	0,64
	12	33,36	31,55	1,26	0,68
	13	31,98	32,80	1,34	0,71
	14	31,99	29,97	1,40	0,68
	15	33,60	28,12	1,51	0,73
	Média	29,94	28,59	1,29	0,61

4. Conclusão

A busca da excelência no controle fiscal das operações sujeitas à incidência do ICMS não pode relevar a análise das diversas variáveis que descrevem o item na NF-e. Conseguir correlacioná-las de forma a compor uma percepção consistente do objeto negociado

viabiliza sistemas de controle e monitoramento fiscais mais assertivos e tempestivos nas ações de fiscalização. Acerca da assertividade, o principal ganho está em mitigar o embaraço gerado por ações fiscais em empresas que cumprem com suas obrigações tributárias, na medida que apenas as que incorressem em irregularidades previamente identificadas é que se submeteriam a tal procedimento. E acerca da tempestividade, o principal ganho está em impedir a concorrência desleal em função da diferença tributária sonegada, gerando o efeito predatório ao comércio local, além de aumentar a eficácia arrecadatória ao se aproximar os momentos da ação fiscal e do fato gerador.

Considerando o cenário de alto volume de informações reportado na introdução deste artigo, adotar um modelo que entregue maior acurácia e menor tempo de predição é fundamental para mitigar a necessidade de intervenção humana, atender a vazão pretendida e reforças os aspectos de assertividade e tempestividade, respectivamente. Extraindo os resultados máximos de acurácia, e os respectivos tempos de predição da massa de testes, observa-se na Tabela 6 que o modelo de Rede neural apresentou a relação mais benéfica, realizando a classificação de 11.538 itens com uma acurácia de 98,86% e em apenas 0,031 segundos.

Tabela 6. Resultados Máximos de Acurácia x Tempo de Predição

Modelo de Classificação	Acurácia	Tempo(Segundos)
KNN	98,75%	0,61
Árvore de Decisão	98,83%	0,085
Rede Neural	98,86%	0,031

Não obstante a elevada assertividade na identificação do produto, outra variável também importante é a quantidade da mercadoria declarada na operação. Embora a NF-e possua campos específicos para se informar a quantidade da mercadoria (campo numérico) e a unidade adotada (campo texto de livre preenchimento), ocorre de contribuinte referenciar a unidade de forma agrupada na descrição do produto. A seguir, dois exemplos de descrições identificadas na base de dados utilizada no processo de aprendizagem:

CERV HEINEKEN LT 350ML

CERV HEINEKEN PIL 0,350LT DESC 12UN PBR

Em ambos os casos, os campos da NF-e de quantidade e de unidade estavam preenchidos, respectivamente, com os valores, “1” e “UN”. Dessa forma, utilizar o valor da mercadoria classificada – Heineken Lata 350 ml – e multiplicar pela quantidade declarada incorreria em um erro de determinação da base de cálculo no segundo exemplo, uma vez que a quantidade real seria de 12 unidades. A correção, inevitavelmente, envolveria um novo processo de mineração para determinar um fator de conversão de unidades, de forma a também compor o cálculo. Para a primeira descrição, teríamos o valor 1 e, para a segunda, 12. Esse tema, no entanto, será desenvolvido em um trabalho futuro.

Agradecimentos

Os autores agradecem à FAPEMA (Fundação de Amparo à Pesquisa do Estado do MA), à CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior -

código financeiro 001) e à SEFAZ/MA (Secretaria de Estado da Fazenda do Maranhão) pelo suporte na realização desse trabalho.

Referencias

SEFAZ/MA (2011). Arrecadação Online. Disponível em http://sistemas.sefaz.ma.gov.br/arrecadacaoonline/arrecadacaoperiodo.html. [Acesso em 01/03/2022].

Jatobá, A., Moura, D., Martins, I., Ramos, H., Aquino, A. (2021) CALT: Uma Ferramenta Automática para Cobrança do ICMS em Operações Interestaduais. In *Anais do IX Workshop de Computação Aplicada em Governo Eletrônico.*

Galdino, G., Silveira, M. R., Fonseca Neto, R. (2004). Uma rede neural artificial de múltiplas camadas aplicadas ao combate à sonegação fiscal de icms. In *Anais do XXXVI Simpósio Brasileiro de Pesquisa Operacional (SBPO)*.

Carmo, M., Boldt, F., Komati, K. (2019). Previsão de receitas de icms do estado do espírito santo através de seleção de características em cascata e técnicas de aprendizado de máquina. In *Anais do XVI Encontro Nacional de Inteligência Artificial e Computacional*, pages 118–129, Porto Alegre, RS, Brasil.

Madeira, R. d. O. C. (2015). Aplicação de técnicas de mineração de texto na detecção de discrepâncias em documentos fiscais. Dissertação (mestrado) – Fundação Getúlio Vargas, Escola de Matemática Aplicada.

MOC (2020). Manual de Orientação ao Contribuinte. Disponível em: http://www.nfe.fazenda.gov.br/portal/listaConteudo.aspx?tipoConteudo=ndIjl+iEFdE=. [Acesso em 24/02/2022].

Receita Federal do Brasil, (2021). Download NCM - Nomenclatura Comum do MERCOSUL. Disponível em https://portalunico.siscomex.gov.br/classif/#/nomenclatura/tabela?perfil=publico. [Acesso em 10/03/2022].

Chollet, F. (2017). Deep Learning with Python, Manning 1st Edition

Developer Guides Keras. Disponível em https://keras.io/guides/. Acesso em 03/03/2022

Rokach, L. and Maimon, O. (2014), Data Mining With Decision Trees: Theory And Applications, World Scientific, 2nd edition.

Grus, J. (2016), Data Science do Zero, Alta Books, 1ª Edição.

User Guide Scikit Learn. Disponível em https://scikit-learn.org/stable/user_guide.html. Acesso em 08/03/2022.

Loy, J. (2019), Neural Networks Projects with Python, Packt Publishing, 1st edition.

Bhasin, H. (2020) Machine Learning for Beginners, BPB Publications, 1st edition.

Bonaccorso, G. (2019), Hands-On Unsupervised Learning with Python, Packt Publishing, 1st Edition.

Johnston, B. and Mathur, I. (2019), Applied Supervised Learning with Python, Packt Publishing, 1st Edition.