UNIVERSIDADE FEDERAL DE PERNAMBUCO
CIN - CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

JOSÉ CORREIA LINS NETO

Audita-NFSe: Sistema Auxiliar de Auditoria em Notas Fiscais de Serviços Eletrônicas

Recife
2021

JOSÉ CORREIA LINS NETO

Audita-NFSe: Sistema Auxiliar de Auditoria em Notas Fiscais de Serviços Eletrônicas

Dissertação apresentada ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática (CIn) da Universidade Federal de Pernambuco – UFPE, como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Área de Concentração: Inteligência Computacional

Orientadora: Drª Flávia de Almeida Barros

Recife

2021

Catalogação na fonte
Bibliotecário Cristiano Cosme S. dos Anjos, CRB4-2290

L759a Lins Neto, José Correia
Audita-NFSe: sistema auxiliar de auditoria em notas fiscais de serviços eletrônicas/ José Correia Lins Neto. – 2021.
89 f.: il., fig.

Orientadora: Flávia de Almeida Barros.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2021.
Inclui referências.

1. Inteligência Computacional. 2. NFS-e. 3. Extração de informação. 4. Classificação de documentos. I. Barros, Flávia de Almeida (orientadora). II. Título

006.31 CDD (23. ed.) UFPE - CCEN 2021 – 119

José Correia Lins Neto

“Audita-NFSE: Sistema Auxiliar de Auditoria em Notas Fiscais de Serviços Eletrônicas”

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Aprovado em: 22/04/2021.

BANCA EXAMINADORA

_____

Profa. Dra. Patricia Cabral de Azevedo Restelli Tedesco
Centro de Informática/ UFPE

_____

Prof. Dr. André Câmara Alves do Nascimento
Departamento de Computação / UFRPE

_____

Profa. Dra. Flavia de Almeida Barros,
Centro de Informática / UFPE
(Orientadora)

AGRADECIMENTOS

Primeiramente, agradeço a Deus por todas as oportunidades que tem me proporcionado, sendo esta mais uma conquista fruto de muita dedicação.

À minha mãe, Maria Correia Lins (em memória), e ao meu Pai, Getúlio do Rego Pereira, pelo apoio dado a mim para que eu pudesse alcançar objetivos pessoais e profissionais.

À minha esposa Bruna Lins, por toda paciência, dedicação e apoio, principalmente nos momentos mais difíceis durante o período do mestrado.

Ao Centro de Informática da Universidade Federal de Pernambuco, pela oportunidade a mim concedida e por toda estrutura disponível.

À minha orientadora, professora Drª Flávia de Almeida Barros, por toda dedicação, atenção e paciência. Seus direcionamentos e orientações foram de grande importância para a conclusão deste trabalho.

Ao amigo Antônio Finizola, que com toda sua paciência me ajudou muito durante o início do curso, compartilhando seus conhecimentos e experiências.

Aos auditores fiscais e a meus superiores da Prefeitura Municipal do Ipojuca/PE, por me proporcionarem toda estrutura e conhecimento necessários à realização deste trabalho.

Aos professores do Centro de Informática da UFPE, por compartilharem seus conhecimentos com grande dedicação.

A todos que não estão listados aqui, mas que de alguma forma fizeram parte desta incrível jornada, deixo aqui meus sinceros agradecimentos!

RESUMO

Os governos estabelecem leis que regulamentam a prestação de serviços definindo elementos como alíquotas e tipos de serviços, importantes para cobrança do imposto. Periodicamente também são estabelecidas leis de isenções, incentivos ou tratamentos fiscais especiais de impostos em determinados serviços. Apesar das regulamentações existentes, os contribuintes cometem fraudes ao emitir Notas Fiscais em desacordo com a legislação, com o objetivo de pagar menos impostos. Essas fraudes também são praticadas na emissão de Notas Fiscais de Serviço Eletrônicas (NFS-e). Apesar das NFS-e estarem disponíveis em formato eletrônico, a análise dessas notas é um processo demorado, pois, em municípios que não possuem sistemas auxiliares de auditoria, é realizada de forma manual pelos auditores. Além disso, sabe-se que no Brasil, a quantidade de Auditores fiscais vem diminuindo ao longo dos anos e, por outro lado, temos um crescente número de empresas e de NFS-e emitidas em todo território nacional. Nesse cenário, este projeto de mestrado teve como objetivo principal a criação de um sistema para auxiliar os Auditores Fiscais Tributários durante o processo de análise e identificação de possíveis fraudes nas NFS-e emitidas pelos contribuintes. O sistema implementado, nomeado de Audita-NFSe, extrai das NFS-e os dados relevantes para auditoria, apresentando essas informações de forma organizada através da interface do usuário. Além disso, o sistema também informa se encontrou indícios de fraudes nas NFS-e sendo analisadas, a fim de facilitar o trabalho dos auditores fiscais. Assim, o sistema executa duas tarefas distintas e complementares: a *Extração de Informação* e a *Classificação* dos documentos. Este trabalho foi desenvolvido no âmbito da Prefeitura do município do Ipojuca (PE) para auxiliar auditores em suas ações de fiscalização do Imposto sobre Serviços (ISS). As regras de extração e de classificação foram criadas com base nas leis que regulamentam o ISS, bem como a partir do conhecimento de auditores fiscais. A construção das regras baseou-se também em um corpus com 3.080 registros de NFS-e de empresas do setor de Construção Civil do município do Ipojuca, emitidas entre os exercícios de 2010 a 2013. O desempenho do sistema foi avaliado utilizando-se um novo corpus com 300 registros de NFS-e dos exercícios de 2014 e 2015. A versão atual do sistema obteve uma média de 90% de Precisão, 86% de Cobertura, 88% de F-measure na Extração de Informação, e 86% de Acurácia na classificação dos dados no corpus de teste. Além dessa avaliação quantitativa,

também foram realizadas entrevistas com os auditores, a fim de avaliarmos a sua satisfação com o sistema. Os resultados obtidos demostram que a iniciativa é válida, e que contribui para o aumento da acurácia e para a redução do tempo gasto nas análises e detecção de possíveis fraudes em NFS-e.

Palavras-chave: NFS-e; Extração de Informação; Classificação de documentos.

ABSTRACT

Governments establish laws that regulate the provision of services by defining elements such as rates and types of services, which are important for tax collection. Periodically, special tax exemptions, incentives or tax treatment laws are also established in certain services. Despite existing regulations, taxpayers commit fraud by issuing Invoices that are not in accordance with the law, in order to pay less taxes. These frauds are also practiced in the issuance of Electronic Service Invoices (NFS-e). Although the NFS-e are available in electronic format, the analysis of these notes is a time-consuming process because, in municipalities that do not have auxiliary auditing systems, it is performed manually by the auditors. In addition, it is known that in Brazil, the number of tax auditors has been decreasing over the years and, on the other hand, we have an increasing number of companies and NFS-e issued nationwide. In this scenario, this master's project had as main objective the creation of a system to assist Tax Auditors during the process of analysis and identification of possible fraud in the NFS-e issued by taxpayers. The implemented system, named Audita-NFSe, extracts the relevant data for auditing from the NFS-e, presenting this information in an organized way through the user interface. In addition, the system also reports whether it found evidence of fraud in the NFS-e being analyzed in order to facilitate the work of tax auditors. Thus, the system performs two distinct and complementary tasks: Extraction of Information and Classification of documents. This work was developed within the scope of the City Hall of the municipality of Ipojuca (PE) to assist auditors in their inspection of the Tax on Services (ISS). The extraction and classification rules were created based on the laws that regulate the ISS, as well as from the knowledge of tax auditors. The construction of the rules was also based on a corpus with 3,080 NFS-e records from companies in the Civil Construction sector in the municipality of Ipojuca, issued between the years 2010 to 2013. The performance of the system was evaluated using a new corpus with 300 NFS-e records from 2014 and 2015. The current version of the system obtained an average of 90% Accuracy, 86% Coverage, 88% F-measure in Information Extraction, and 86% Accuracy in the classification of data in the test corpus. In addition to this quantitative assessment, interviews were also conducted with the auditors in order to assess their satisfaction with the system. The results obtained demonstrate that the initiative is valid, and that

it contributes to increase the accuracy and to reduce the time spent on the analysis and detection of possible fraud in NFS-e.

Keywords: NFS-e; Information Extraction; Classification of documents.

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE ABREVIATURAS E SIGLAS

ABRASF	Associação Brasileira das Secretarias de Finanças das Capitais
AM	Aprendizagem de Máquina
Art	Artigo
BPMN	Business Process Model and Notation
CGU	Controladoria Geral da União
CSV	Comma-separated values
Doc	Documento
EC	Engenharia de Conhecimento
EI	Extração de Informação
Fisco	Órgão público responsável pela arrecadação de impostos
FN	Falso Negativo
FP	Falso Positivo
HTML	HyperText Markup Language
IA	Inteligência Artificial
IBM	International Business Machines Corporation
ISS	Imposto Sobre Serviços
KDD	Knowledge Discovery in Databases
KDT	Knowledge Discovery in Texts
K-NN	K-Nearest Neighbors
MD	Mineração de Dados
NB	Naive Bayes
NFS-e	Nota Fiscal de Serviço Eletrônica
NL	Nota de Liquidação
PDF	Portable Document Format
PE	Pernambuco
PLN	Processamento de Linguagem Natural
RI	Recuperação de Informação
RegEx	Regular Expression
RM	Relatório de Medição
SBC	Sistemas Baseados em Conhecimento
SQL	Structured Query Language
SVM	Support Vector Machines
VN	Verdadeiro Negativo
VP	Verdadeiro Positivo
Web	Internet
WEKA	Waikato Environment for Knowledge Analysis
XML	Extensible Markup Language

SUMÁRIO

1 INTRODUÇÃO

Os governos estabelecem leis que regulamentam a prestação de serviços definindo elementos como alíquotas e tipos de serviços, importantes para cobrança do imposto. Periodicamente também são estabelecidas leis de isenções, incentivos ou tratamentos fiscais especiais de impostos em determinados serviços. Apesar das regulamentações existentes, os contribuintes cometem fraudes ao emitir Notas Fiscais em desacordo com a legislação, com o objetivo de pagar menos impostos. Essas fraudes também são praticadas na emissão de Notas Fiscais de Serviço Eletrônicas.

"A Nota Fiscal de Serviço Eletrônica (NFS-e) é um documento de existência exclusivamente digital, gerado e armazenado eletronicamente pela prefeitura ou por outra entidade conveniada, para documentar as operações de prestação de serviços" (ABRASF, 2008).

De acordo com o Art. 1º da Lei de nº 8.846/1994,

> (...) emissão de nota fiscal, recibo ou documento equivalente, relativo à venda de mercadorias, prestação de serviços ou operações de alienação de bens móveis, deverá ser efetuada, para efeito da legislação do imposto sobre a renda e proventos de qualquer natureza, no momento da efetivação da operação (BRASIL, 1994, s/n).

Auditores fiscais ligados ao governo federal, estadual e municipal, são responsáveis por investigar e identificar fraudes de qualquer natureza nas notas fiscais.

"A partir da implementação dos sistemas de NFS-e, as Administrações Tributárias Municipais poderão obter um melhor controle fiscal e de arrecadação do ISS" (Imposto Sobre Serviços) (ABRASF, 2008).

As NFS-e utilizadas no processo de auditoria podem ser visualizadas e impressas, uma a uma, ou exportadas em grande quantidade em formato XML (*Extensible Markup Language*) ou CSV (*Comma-separated values*) a partir do portal na Web do município, para serem analisadas pelos auditores. Porém, apesar de estarem em formato eletrônico, a análise dessas notas é um processo que pode demandar muito tempo, pois, em municípios que não possuem sistemas auxiliares de auditoria, ela é feita de forma manual pelos auditores.

De acordo com Melo e Oliveira (2017), a eficiência do Fisco brasileiro se mostra muito deficitária. Este problema ocorre porque a quantidade de auditores da Receita Federal vem diminuindo ao longo dos anos. Isso se reflete também em outras esferas do Governo Federal, Estadual e Municipal. Além disso, temos o crescente número de empresas e de NFS-e emitidas em todo território nacional.

Nesse cenário, este projeto de mestrado teve como objetivo principal a criação de um sistema para auxiliar os Auditores Fiscais durante o processo de análise e identificação de possíveis fraudes nas NFS-e emitidas pelos contribuintes.

1.1 TRABALHO REALIZADO

As NFS-e apresentam diversas informações a respeito dos serviços prestados. Essas informações devem ser cuidadosamente analisadas pelos auditores, em busca de irregularidades ou fraudes. Contudo, como mencionado anteriormente, vivenciamos atualmente um déficit de auditores fiscais para o grande volume de NFS-e a serem auditadas.

Com o objetivo de auxiliar esses profissionais, este trabalho propõe uma solução computacional capaz de automatizar parte do trabalho manual do auditor fiscal, referente à análise de NFS-e visando a identificação de possíveis fraudes. O sistema implementado, nomeado de Audita-NFSe, extrai das NFS-e os dados relevantes para auditoria, apresentando essas informações de forma organizada através da interface do usuário. Além disso, o sistema também informa se encontrou indícios de fraudes nas NFS-e analisadas, a fim de facilitar o trabalho dos auditores fiscais. Na prática, essa análise é como uma classificação prévia das NFS-e fraudulentas. Assim, o sistema executa duas tarefas distintas e complementares: a *Extração de Informações* e a *Classificação dos documentos*.

À primeira vista, a tarefa de selecionar (extrair) automaticamente esses dados parece simples, uma vez que grande parte das informações de uma NFS-e é armazenada de forma estruturada (por exemplo, campos como data de prestação do serviço, Identificação do prestador de serviço, etc.). Contudo, os campos mais importantes para auditoria, *Observação* e *Discriminação dos Serviços*, são preenchidos em texto livre, sendo geralmente utilizados para descrever detalhes dos serviços prestados (tipo do serviço prestado, período de execução, valores, números de contratos, etc.). Como eles trazem informações mais detalhadas sobre os serviços

prestados, esses campos são de extrema importância para a realização de auditorias, a fim de verificar se a lei vigente foi obedecida. Porém, por serem escritos em linguagem natural livre, não é possível realizar consultas a esses campos da NFS-e com comandos de uma Linguagem de Consulta Estruturada (do inglês, *Structured Query Language - SQL*). Essa dificuldade prejudica o cruzamento de dados entre documentos, que é crucial para a realização de auditorias.

Nesse cenário, surgiu a necessidade da criação de um sistema capaz de extrair também os dados relevantes desses campos em texto livre, com o objetivo de auxiliar auditores fiscais a identificar possíveis fraudes por parte dos prestadores de serviço. As informações extraídas são exibidas em um formato que facilita sua visualização, e também possibilita seu uso como entrada para outros sistemas. Assim, o processo de fiscalização torna-se mais rápido e seguro, devido ao poder de processamento de grandes quantidades de dados por esse sistema computacional.

Este trabalho utiliza técnicas de Inteligência Artificial (IA) para realizar Mineração de Textos (MT). Em particular, o trabalho realiza duas tarefas principais da MT: (1) a *Extração de Informação* (EI) baseada em regras, para extrair informações úteis de campos das NFS-e; e (2) a *Classificação* dos documentos, indicando indícios de fraudes, para auxiliar auditores fiscais em suas ações de fiscalização do Imposto sobre Serviços (ISS).

Este trabalho foi desenvolvido no âmbito da Prefeitura do município do Ipojuca (PE). As regras de extração e classificação foram criadas com base nas leis que regulamentam o ISS, bem como a partir de entrevistas para coletar parte do conhecimento de auditores fiscais, na aplicação dessas leis em suas fiscalizações. Além dessas informações, a construção das regras baseou-se também em um corpus com 3.080 registros de NFS-e de empresas do setor de Construção Civil do município do Ipojuca, emitidas entre os exercícios de 2010 a 2013, para identificar padrões de escrita de prestação de serviços.

O desempenho do sistema foi avaliado utilizando-se um novo corpus com 300 registros de NFS-e dos exercícios de 2014 e 2015 das mesmas empresas. As medidas utilizadas para a avaliação do processo de Extração de Informação foram *Cobertura*, *Precisão* e *F-measure* de cada campo extraído. Já o processo de Classificação de Texto foi avaliado pela *Acurácia*. A versão atual do sistema obteve uma média de 90% de *Precisão*, 86% de *Cobertura*, 88% de *F-measure*, e uma média de 86% de *Acurácia*.

1.2 ESTRUTURA DO DOCUMENTO

O presente trabalho está dividido em mais 5 capítulos, além da Introdução. Eles estão organizados da seguinte forma:

• Capítulo 2: Apresenta uma breve revisão bibliográfica sobre os fundamentos da Mineração de Texto (MT), seus conceitos básicos e algumas das principais tarefas desse processo.

• Capítulo 3: Apresenta brevemente os fundamentos da Extração de Informações (EI) e da Classificação de texto, seus conceitos básicos, técnicas e as medidas de avaliação desses processos.

• Capítulo 4: Apresenta detalhes da Aplicação Audita-NFSe: Contexto da auditoria, seu processo e evidências de fraudes; a solução proposta com a visão geral e etapas do processamento; detalhes do processo de Extração e Classificação das Notas Fiscais de Serviços Eletrônicas.

• Capítulo 5: Apresenta os detalhes da implementação do protótipo, testes de avaliação e os resultados obtidos.

• Capítulo 6: Apresenta as considerações finais deste trabalho.

2 MINERAÇÃO DE TEXTO

Atualmente, vemos um número crescente de documentos textuais disponíveis em redes sociais, e-mails, notas fiscais eletrônicas, jornais, livros e revistas na Web, utilizando formatos variados, como PDF, XML, HTML, CSV, Doc, entre outros. Esses documentos textuais podem conter informações relevantes escondidas ou padrões de dados sem estrutura definida, escritas em texto livre.

A análise desses dados é de grande importância no contexto empresarial e governamental, para apoiar a tomada de decisões. Automatizar essas tarefas de análise têm sido foco de pesquisa de diversas áreas, com destaque para a Descoberta de Conhecimento em Bases de Dados (do inglês, KDD - *Knowledge Discovery in Databases*).

Segundo Maimon e Rokach (2005), KDD é um processo organizado de identificação de padrões válidos, novos, úteis e compreensíveis de conjuntos de dados grandes e complexos. É a extração automatizada de padrões que representam o conhecimento implicitamente armazenado ou capturado em grandes bancos de dados, *data warehouses*, Web e outros repositórios massivos de informações.

Os conceitos de KDD também podem ser aplicados à análise de textos não estruturados, sendo essa técnica conhecida como Descoberta de Conhecimento em Textos (do inglês, KDT - *Knowledge Discovery in Texts*), ou ainda Mineração de Texto. KDT procura extrair informações úteis de fontes de dados através da identificação e exploração de padrões relevantes (FELDMAN; SANGER, 2006).

De acordo com Brito (2016), esse processo trata da descoberta de padrões úteis a partir de uma grande quantidade de dados não estruturados, isto é, dados que não estão armazenados de forma estruturada, mas em textos escritos de forma livre.

Este capítulo tem como objetivo apresentar os principais conceitos da Mineração de Textos. Veremos brevemente as principais técnicas para lidar com dados em forma de texto não estruturados.

2.1 DEFINIÇÕES

Mineração de Textos (MT), segundo Tan (1999), é o processo de extração de padrões úteis e não triviais a partir de documentos textuais. Essa definição pode se confundir com conceitos da Mineração de Dados (MD), pois segundo Fayyad,

Piatetsky-Sahapiro e Smyth (1996), MD consiste na execução de algoritmos computacionais para extrair padrões de dados. A diferença entre as duas áreas de pesquisa está no tipo de dado analisado.

Enquanto a Mineração de Dados é aplicada a dados estruturados oriundos, por exemplo, de bancos de dados, a Mineração de Textos é aplicada a dados semiestruturados ou não estruturados. Dessa forma, a MT concentra-se em documentos textuais (e.g., páginas web, documentos livres, e-mails ou até campos de bancos de dados relacionais que armazenem informações escritas em texto livre).

Citamos aqui algumas definições de Mineração de Dados. Han, Kamber e Pei (2011) afirmam que MD é um processo de descobrir padrões e conhecimentos interessantes de grandes quantidades de dados. Dean (2014), também diz, que MD é um campo emergente que oferece grandes oportunidades para a conversão de dados em informações.

Por fim, Cortes, Porcaro e Lifschitz (2002), citam que:

> (...) a Mineração de Dados torna-se um processo altamente cooperativo entre homens e máquinas, pois visa a exploração de grandes bancos de dados, com o objetivo de extrair conhecimentos através do reconhecimento de padrões e relacionamentos entre variáveis (p.2).

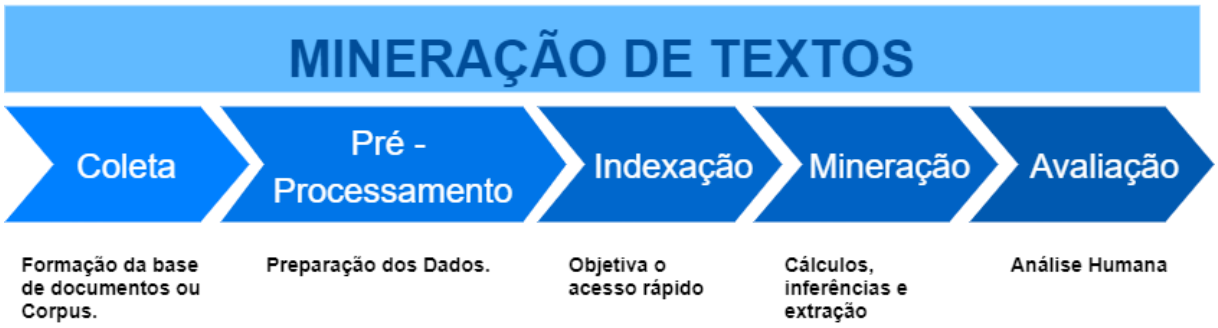
Citamos agora algumas definições de Mineração de Textos importantes. Lopes (2004), afirma que MT é o “processo de extrair padrões ou conhecimento a partir de documentos em textos não-estruturados” (p.10). Já para Andrade (2015), MT “é o processo de descoberta de conhecimento que utiliza técnicas de extração de dados a partir de textos, frases ou palavras” (p.6). Aranha e Passos (2006) afirmam que MT tem por objetivo “extrair conhecimentos de dados não estruturados ou semi-estruturados” (p.28).

A próxima seção apresenta as principais tarefas do processo de Mineração de Textos, que podem ser usadas na análise de qualquer tipo de texto semiestruturado ou não estruturado.

2.2 PRINCIPAIS TAREFAS DA MINERAÇÃO DE TEXTOS

Segundo Aranha e Passos (2006) o processo de Mineração Textos como um todo, se constitui de 5 etapas (Figura 1): Coleta, Pré-Processamento, Indexação, Mineração e Análise.

Figura 1 - Etapas da Mineração de Texto



Fonte: Adaptado de Aranha e Passos (2006)

2.2.1 Coleta de Dados

Assim, a coleta de dados é responsável pela criação da base dos documentos textuais a serem minerados, também conhecida como Corpus. Segundo Pezzini (2016), “a coleta de documentos objetiva conseguir documentos relacionados ao tipo de conhecimento que se deseja obter” (p.36). De acordo com Aranhas e Passos (2006), geralmente o processo de mineração de textos começa a partir de uma base pré-existente. Contudo, essa não é a realidade em todas as situações.

A criação da base de documentos pode ser realizada de forma manual ou automática (se houver um sistema de coleta implementado). Esta é uma etapa bastante crítica, uma vez que, em muitos casos, os dados precisam ser coletados de várias fontes distintas, estando em formatos diferentes. Existem diversas técnicas do Processamento de Linguagem Natural (PLN) e de Recuperação de Informação (RI) que podem ser utilizadas nesta etapa, a fim de auxiliar a correta realização da coleta de documentos (MARTINS et al., 2003).

2.2.2 Pré-Processamento

Nesta etapa, os documentos do corpus, escritos em linguagem natural (não estruturada), passam por um pré-processamento que objetiva estruturá-los de maneira padronizada, mas sem perder suas características naturais (PEZZINI, 2016). O objetivo desta etapa é extrair dos textos uma representação estruturada e manipulável por algoritmos (CORRÊA; MARCACINI; REZENDE, 2012).

Os documentos originais podem conter muitos termos irrelevantes, erros ortográficos, caracteres especiais, tags html, entre outros problemas. Assim sendo, é necessário aplicar algumas técnicas para seleção, transformação, limpeza e redução do volume desses dados (MORAIS; AMBRÓSIO, 2007). Dessa forma, os dados poderão ser transformados de maneira apropriada, retirando-se grande parte das anomalias, redundâncias e informações desnecessárias que possam acarretar resultados distorcidos (GONÇALVES *et al*., 2006).

A seguir serão apresentadas algumas das principais técnicas aplicadas nesta etapa: *tokenização*, remoção de *stopwords*, *stemming*, *case folding*, e remoção de tags e códigos HTML.

2.2.2.1 Tokenização

Esta técnica é utilizada para dividir o texto de entrada em unidades chamadas *tokens*, sendo cada unidade uma palavra, um número, um sinal de pontuação ou algum caractere especial. A divisão também pode ocorrer em vários níveis diferentes, como capítulos, seções, parágrafos, frases e até sílabas ou fonemas (FELDMAN; SANGER, 2006).

Figura 2 - Exemplo de Tokenização

Texto Original:'Olá Mundo! Este é um teste de string.'

Texto Tokenizado: ['Olá', 'Mundo', '!', 'Este', 'é', 'um', 'teste','de',

Fonte: O autor (2020)

2.2.2.2 Remoção de Stopwords

Stopwords são artigos, preposições, conjunções e outras palavras auxiliares que não agregam valor semântico ao texto. Geralmente, essas palavras são armazenadas em uma lista denominada *Stoplist*, para facilitar sua remoção. As *stoplists* podem ser personalizadas de acordo com o assunto ou idioma do texto (ANDRADE, 2015).

Segundo Manning, Raghavan e Schutze (2008), a eliminação dessas palavras reduz a quantidade de termos a serem considerados, diminuindo o custo computacional das próximas etapas.

Figura 3 - Exemplo de Remoção de *Stopwords*

Texto Original: 'Olá Mundo! Este é um teste de *string*.'
Texto sem *Stopwords*: ['Olá', 'Mundo', 'Este', 'teste', *'string'*]

Fonte: O autor (2020)

2.2.2.3 Stemming

O processo de *stemming* “consiste em uma normalização linguística, na qual as formas variantes de uma palavra são reduzidas a uma forma comum — *stem*” (MARTINS *et al*., 2003). Essa redução é obtida “eliminando os prefixos e sufixos que indicam variação na forma da palavra, como plural e tempos verbais” (BRITO, 2016, p.8).

Figura 4 - Exemplo de *Stemming*

PALAVRA	**STEM**
Química	Químic
Químico	Químic
Qúimicos	Químic

Fonte: O autor (2020)

Com a aplicação dessa técnica, é possível reduzir o esforço computacional, aumentando assim a precisão dos resultados (BRITO, 2016).

Contudo, é necessário observar situações em que a retirada de prefixos e/ou sufixos muda o significado do termo original. Um exemplo desse último caso seria “fazer” e “desfazer” – termos que possuem significados opostos. Neste caso, retirada do prefixo “des” muda o significado da palavra original.

Em termos práticos, na grande maioria dos casos, apenas a retirada de prefixos modifica o significado do termo original. Assim sendo, geralmente utilizamos algoritmos de *stemming* que retiram apenas os sufixos.

2.2.2.4 Case Folding

Segundo Madeira (2015), este procedimento visa converter todas as letras de um documento para um único tipo, ou maiúsculo ou minúsculo.

Figura 5 - Exemplo de *Case Folding*

Texto Original: 'Olá Mundo! Este é Um Teste de *String*.'
Texto Convertido: ‘ola mundo! este é um teste de *string*’

Fonte: O autor (2020)

Contudo, este processo deve que ser utilizado com cuidado, pois pode tornar iguais palavras que deveriam ser diferenciadas (MANNING; RAGHAVAN e SCHUTZE, 2008). Como exemplo, citamos o termo “papa”. “Papa” com letra maiúscula é o líder da Igreja Católica, enquanto “papa” é um tipo de prato preparado com leite.

2.2.2.5 Remoção de Tags e Códigos HTML

Os documentos textuais coletados da Web formatados em código HTML podem possuir em sua estrutura vários campos, como título, metadados, o corpo do documento, e suas *Tags* (AGGARWAL, 2015). Esses códigos HTML podem ser removidos utilizando-se diversas técnicas, a fim de diminuir a quantidade de caracteres no texto, reduzindo possíveis erros de processamento dos algoritmos de mineração.

Figura 6 - Exemplo de Remoção de *Tags* e Códigos HTML

Texto Codificado:<body>
Olá Mundo! Este é um teste de string.
</body>

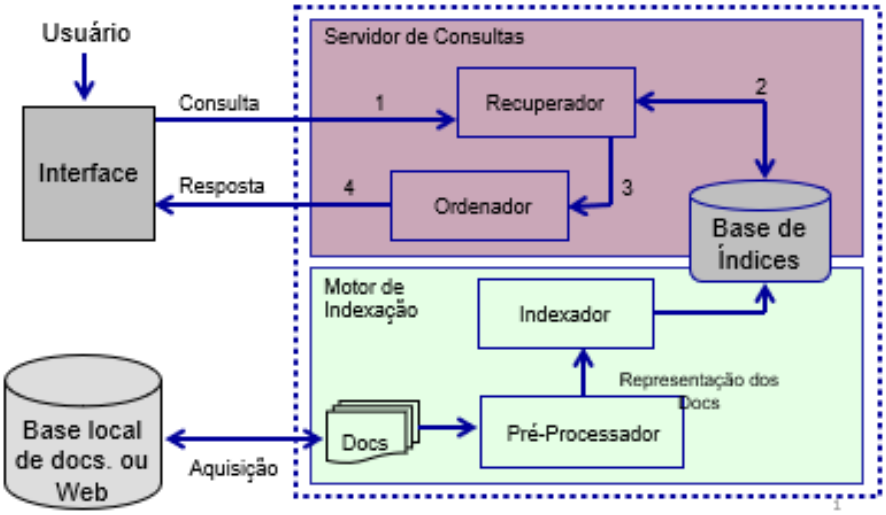
Texto Convertido: Olá Mundo! Este é um teste de string.

Fonte: O autor (2020)

2.2.3 Indexação

Segundo Feldman e Sanger (2006), cada documento de uma grande coleção pode ser associado a uma ou mais palavras-chave que descrevem seu conteúdo. Essa operação é chamada de Indexação da base de documentos. Em seguida, um sistema de RI pode recuperar os documentos de acordo com as consultas do usuário, com base nos termos principais que descrevem cada documento.

Figura 7 - Indexação



Fonte: Barros (2017)

2.2.4 Mineração

Mineração é a etapa onde ocorre a busca por conhecimentos novos a partir dos dados do *corpus*. Este processo envolve a aplicação de técnicas que processam os dados e identificam informações úteis e implícitas, que normalmente não poderiam

ser recuperadas utilizando métodos tradicionais de consulta, pois a informação contida nestes dados não pode ser obtida de forma direta, uma vez que estão armazenadas em formato de texto não estruturado (SOARES, 2008; MORAIS; AMBRÓSIO, 2007).

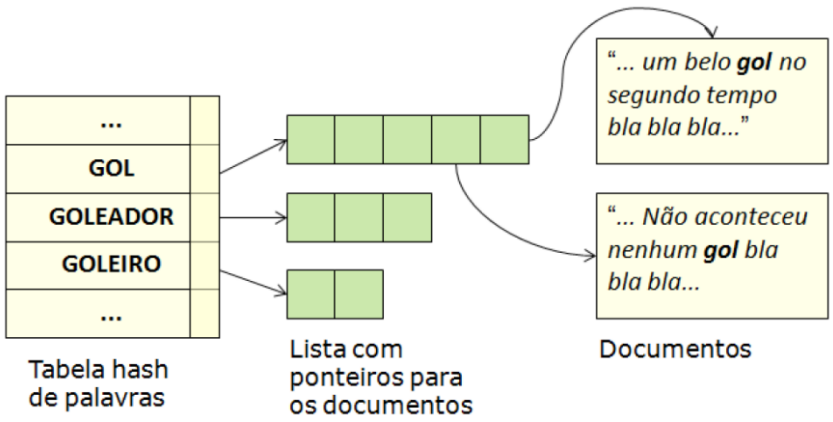
Ainda segundo Moraes e Ambrósio (2007), “as principais contribuições estão relacionadas à busca de informações em documentos, à análise qualitativa e quantitativa de grandes quantidades de textos, e a melhor compreensão do conteúdo disponível em documentos textuais” (p.15).

As técnicas utilizadas nesta etapa podem ser aplicadas isoladamente ou em combinação de duas ou mais. Veremos a seguir as principais técnicas que fazem parte desta etapa da MT.

2.2.4.1 Recuperação de Informação (RI)

RI é uma área que estuda como armazenar e recuperar dados, geralmente textuais, de forma automática. Este processo baseia-se em técnicas da ciência da computação para criar estruturas de índices, organizar e recuperar as informações de forma eficiente. As estruturas de índices permitem identificar, no conjunto de documentos (corpus), quais atendem à necessidade de informação do usuário (BRITO, 2016).

Figura 8 - Exemplo de Recuperação de Informação (RI)



Fonte: Gonçalves (2012)

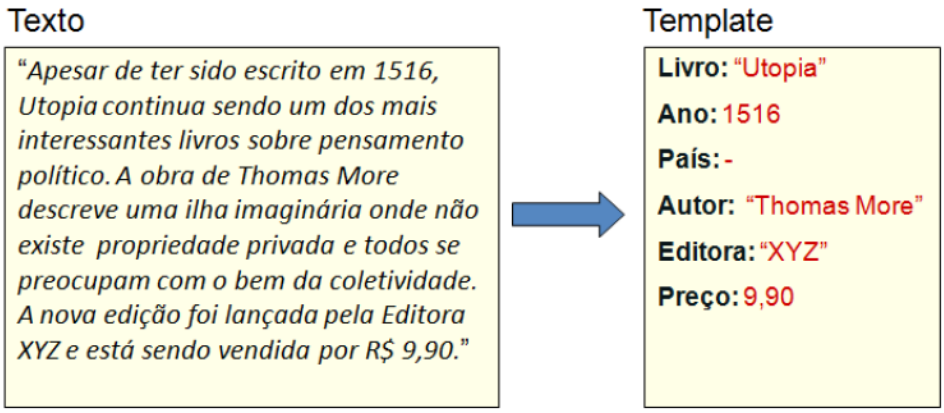
Segundo Gonçalves (2012), o objetivo dos sistemas de RI é localizar e ordenar documentos relevantes em uma coleção, de acordo com as palavras-chave digitadas em uma consulta feita pelo usuário, assim como ocorre nos sites de busca na web.

De acordo com Barion e Lago (2008), existem 3 modelos básicos de recuperação para sistemas de RI. O modelo *Booleano* é considerado o mais simples. Esse modelo considera uma consulta como uma expressão booleana formada pelos conectivos lógicos AND, OR e NOT. O modelo *Espaço Vetorial* representa documentos e consultas como vetores em um espaço multidimensional de termos. A consulta do usuário é respondida com base na similaridade (proximidade) entre o vetor que representa o documento e o vetor que representa a consulta. Por fim, o modelo *Probabilista* descreve documentos considerando pesos binários, representando a presença ou ausência de termos. O cálculo da similaridade entre consulta e documento é baseado nas probabilidades a priori de cada documento pertencer ao conjunto de documentos relevantes para aquela dada consulta.

2.2.4.2 Extração de Informações

"É o processo de extrair informação pré-definida sobre objetos e relacionamentos entre eles a partir de documentos escritos em texto livre" (LOPES, 2004, p.87). Sistemas de EI processam uma coleção de documentos que são "quebrados" em pedaços de informação. Esses pedaços (segmentos) do texto são então usados para preencher formulários (*templates*) estruturados, compostos por campos previamente definidos.

Figura 9 - Exemplo de Extração de Informação (EI)



Fonte: Gonçalves (2012)

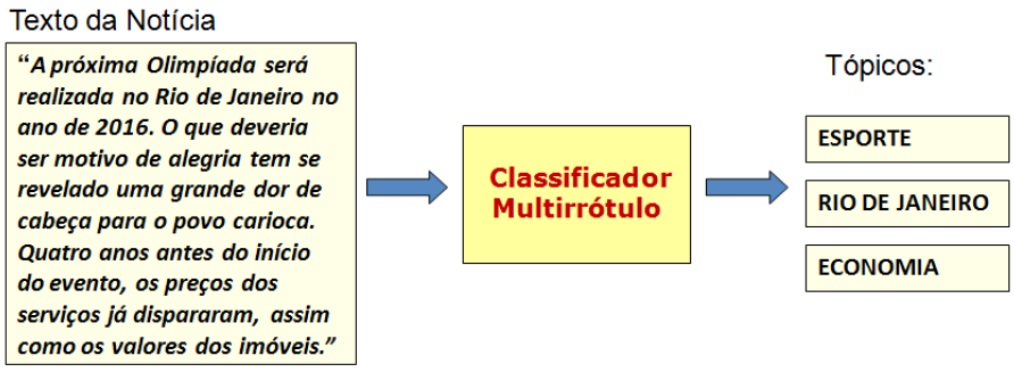
Cada campo do *template* tem características próprias que podem ser capturadas por regras ou padrões de extração (como datas, locais, nomes de empresas ou de pessoas, etc.). Segundo Álvarez (2007), a informação extraída é localizada por um conjunto de padrões ou regras de extração específicas para o domínio da aplicação. Ainda, a definição desses padrões pode ser feita manualmente, por algum especialista, ou com diferentes graus de automação.

O capítulo 3 traz uma breve apresentação sobre os fundamentos da Extração de Informações, seus conceitos básicos e algumas das principais tarefas desse processo. Esse detalhamento foi necessário por EI ser um dos processos principais da MT realizada neste trabalho de mestrado.

2.2.4.3 Classificação de Texto

A classificação de texto ou documento consiste em categorizar um dado texto em um conjunto pré-definido de categorias, assuntos ou tópicos (FELDMAN; SANGER, 2006).

Figura 10 - Exemplo de Classificação



Fonte: Gonçalves (2012)

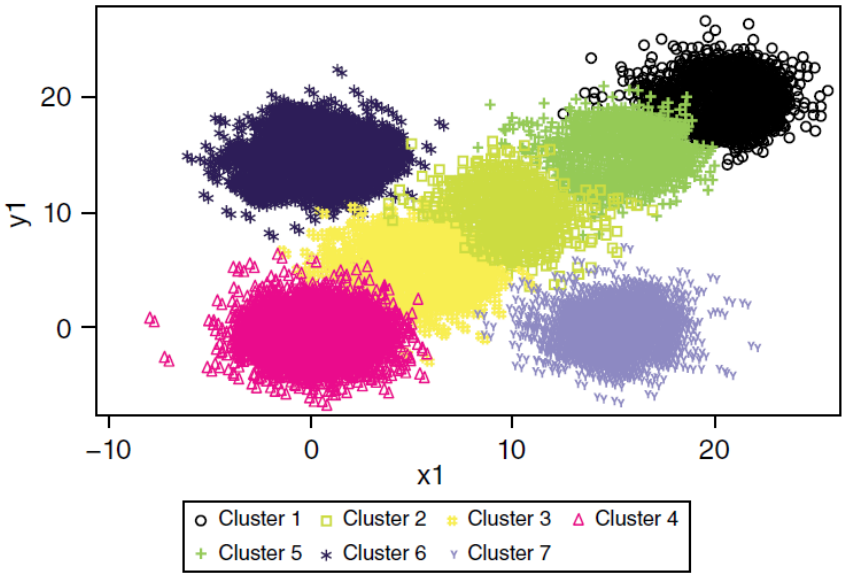
Segundo Madeira (2015), identificamos duas abordagens principais para realizar a construção de classificadores de texto: a primeira é centrada no conhecimento de especialistas, que é diretamente codificado no sistema em forma de regras de classificação (Engenharia do Conhecimento - EC); e a segunda é a Aprendizagem de Máquinas (AM), na qual um processo indutivo constrói um classificador com base no treinamento de um algoritmo de AM a partir de exemplos pré-classificados.

O capítulo 3, a seguir, apresentará mais detalhes sobre essa tarefa da MT, uma vez que ela é uma etapa vital do sistema desenvolvido neste trabalho de mestrado.

2.2.4.4 Agrupamento (*clustering*)

Agrupamento é uma tarefa de mineração na qual uma coleção de textos deve ser organizada em grupos afins (*clusters*), não havendo aqui categorias pré-estabelecidas. Para tanto, os algoritmos de agrupamento utilizam uma medida de similaridade, ou seja, textos de um mesmo grupo devem ser similares entre si, porém dissimilares em relação aos textos de outros grupos (MADEIRA, 2015).

Figura 11 - Exemplo de visualização de agrupamento



Fonte: Dean (2014)

O agrupamento pode ser usado para estruturar e sintetizar o conhecimento, facilitando a identificação de padrões, quando este é incompleto ou quando há muitos atributos a serem considerados (MORAIS, AMBRÓSIO, 2007).

"O processo de agrupamento também é conhecido como aprendizado por observação, pois a organização dos objetos em grupos é realizada apenas pela observação dos padrões nos dados, sem uso de conhecimento externo" (MADEIRA, 2015, p.9 *apud* XU; WUNSCH, 2008).

Por fim, vale ressaltar que, segundo Madeira (2015), "os algoritmos de agrupamento permitem sobreposição de grupos, ou seja, um mesmo documento pode

pertencer a mais de um grupo ou possuir um grau de semelhança associado a cada grupo” (p.21).

2.2.5 Análise/Avaliação

Esta fase da MT envolve a aplicação de técnicas de análise dos resultados do processo de mineração. É possível avaliar os resultados de um determinado sistema a fim de determinar quão bem o sistema executou suas tarefas (MORAIS; AMBRÓSIO, 2007; ZAMBENEDETTI, 2002).

Identificamos duas opções de se avaliar esses resultados: a análise objetiva, através de métricas que calculam a qualidade dos resultados; e a análise subjetiva, baseada no conhecimento de um especialista do domínio (MADEIRA, 2015). É importante destacar que cada processo de análise pode adotar métricas de desempenho diferentes para calcular a qualidade dos resultados de acordo com seu objetivo e técnica escolhida.

O processo geral de MT pode ser avaliado globalmente com análises qualitativas ou quantitativas. Pode-se medir, por exemplo, a precisão do processo geral executado. Contudo, o mais comum é realizar avaliações parciais, para cada etapa do processo.

As medidas mais usadas nas avaliações são *Precisão* e *Cobertura*. Dependendo da tarefa, essas medidas são calculadas com pequenas variações (BAEZA-YATES; RIBEIRO-NETO, 1999):

- RI – A Precisão é dada pela a razão entre a quantidade de documentos relevantes recuperados e a quantidade de documentos retornados pelo processo. A Cobertura é dada pela a razão entre a quantidade de documentos relevantes recuperados e a quantidade de documentos relevantes existentes na base de documentos.
- EI - Aqui, essas medidas são calculadas para cada campo do *template*. A Precisão é dada pela a razão entre a quantidade de vezes que um dado campos do *template* foi preenchido corretamente e a quantidade de vezes que o campo foi preenchido. Já a Cobertura é dada pela a razão entre a quantidade de vezes que um dado campos foi preenchido corretamente e a quantidade de vezes que os documentos de origem possuem a informação a ser extraída.

• Classificação – Aqui, essas medidas são calculadas para cada categoria considerada pelo classificador. A Precisão é dada pela razão entre a quantidade de documentos de uma dada categoria corretamente classificados como positivos e a quantidade total de documentos que foram classificados como sendo dessa classe. A Cobertura é dada pela razão entre a quantidade de documentos de uma dada categoria corretamente classificados como positivos e a quantidade total de documentos dessa classe no *corpus* de entrada.

No capítulo 3, a seguir, veremos mais detalhes sobre as métricas de avaliação das tarefas centrais para o sistema Audita-NFSe: *Extração de informação* e *Classificação de texto*.

2.3 CONSIDERAÇÕES FINAIS

A Mineração de Textos é uma abordagem eficiente e poderosa para análise de grandes e complexos conjuntos de dados em forma de textos. Esses documentos são escritos de forma livre, não estruturada, e passam por diversas etapas para descoberta de conhecimentos novos e úteis.

Essas etapas podem incluir a Coleta dos corpus de textos; o Pré-processamento para que os textos passem por limpeza, transformação ou remoção de caracteres a fim de diminuir as anomalias existentes; Indexação, que visa armazenar os dados de forma inteligente e ordenada; Mineração, que tem como objetivo a identificação de novo conhecimento com a aplicação de técnicas como Classificação, Agrupamento, Extração de Informação e Recuperação de Informação; e a etapa de Avaliação, que é responsável por avaliar os resultados obtidos no processo geral.

É importante ressaltar que existem vários fatores responsáveis pela qualidade dos resultados obtidos na descoberta de conhecimento em textos, além do processo de MT em si. Devemos considerar também fatores como a origem e a qualidade dos dados, o cumprimento das etapas de forma bem elaborada, e sendo necessário, a realização eventual ajustes no processo inicialmente definido.

3 EXTRAÇÃO DE INFORMAÇÕES E CLASSIFICAÇÃO DE TEXTO

Com a difusão em massa do uso da internet tanto por pessoas, empresas e governo, que a quantidade de dados disponíveis em meio digital tem aumentado exponencialmente. Segundo Sette e Martins (2016), “cerca de 80% das informações contidas nas organizações estão armazenadas em documentos textuais” (p.17). Este fato ocorre principalmente pela substituição do uso do papel, pelo meio digital ao exemplo de livros, jornais, revistas, notas fiscais de serviços, entre outros.

Em geral, esses documentos são armazenados em bancos de dados estruturados. Contudo, parte dessas informações são armazenadas em formato de texto não estruturado ou semiestruturado, sendo necessário um método eficaz para buscar e extrair informações relevantes. As informações extraídas podem ser usadas para preencher formulários pré-definidos que serão apresentados para usuários, e/ou armazenados em bancos de dados.

Quando necessário, os registros com as informações extraídas podem ainda ser classificados automaticamente. Como visto no capítulo anterior, a categorização de texto é também uma tarefa básica da Mineração de Texto.

Este capítulo tem como objetivo apresentar os principais conceitos, métodos e técnicas de *Extração de Informação* e de *Classificação de texto*, que são as duas tarefas principais do sistema Audita-NFSe, desenvolvido no contexto deste trabalho de mestrado.

3.1 ABORDAGENS PARA CONSTRUÇÃO DE SISTEMAS DE EI E CLASSIFICADORES AUTOMÁTICOS

Como já mencionado no Capítulo 2, podemos identificar duas abordagens predominantes na construção de sistemas de EI, bem como de classificadores de texto, ambas oriundas da área de Inteligência Artificial: *Aprendizagem de máquina* (AM) e *Engenharia do conhecimento* (EC). Essas abordagens serão detalhadas nas seções 3.1.1 e 3.1.2 a seguir.

3.1.1 Aprendizagem de Máquina

A Aprendizagem de máquina é uma subárea da Inteligência Artificial relacionada a algoritmos que permitem que os computadores “aprendam” (i.e., criem modelos) a partir dos dados de entrada.

Segundo Segaran (2007), existem muitos algoritmos diferentes de aprendizado de máquina, cada um adequado a um tipo de problema diferente. De acordo com Rosa (2016), “o aprendizado de máquina pode ser dividido em dois importantes subgrupos de algoritmos: os que compõe a aprendizagem não-supervisionada e a aprendizagem supervisionada” (p.46).

De acordo com Silva (2008),

> (...)a diferença reside no fato de que, no aprendizado supervisionado, um conjunto de dados já classificado é fornecido ao sistema, enquanto no não-supervisionado o sistema deve, por si só, decidir quais são as classes pertinentes, simplesmente agrupando padrões semelhantes” (p.3).

Veremos a seguir mais detalhes sobre essas duas modalidades de AM.

3.1.1.1 Aprendizado Supervisionado

Em analogia aos humanos, os algoritmos aprendem através de treinamento. Para treinar esses algoritmos de AM, são utilizados conjuntos de treinamento (SEGARAN, 2007; GRUS, 2016; HARRINGTON, 2012). O algoritmo recebe um conjunto de dados etiquetados e infere informações sobre as propriedades dos dados, e essas informações permitem fazer previsões sobre outros dados que possam ser vistos no futuro. Isso é possível, porque dados não aleatórios podem conter padrões (SEGARAN, 2007).

Após a fase de treinamento, os modelos induzidos devem ser testados, a fim de se verificar seu desempenho. Geralmente, são utilizados dois conjuntos distintos de exemplos: um conjunto de dados de treinamento e outro conjunto de dados, chamado de conjunto de teste (HARRINGTON, 2012). Esse conjunto de teste é responsável por avaliar se as previsões do modelo induzido estão corretas ou não.

Esta abordagem pode ser bastante útil para problemas de *classificação* e *regressão*, em que já se sabe de antemão os resultados esperados (SILVA, 2008).

Quando a saída for um conjunto finito de valores/classes, o problema da aprendizagem será chamado de *classificação*. O trabalho de classificação é prever a que classe uma instância de dados pertence. A classificação consiste em atribuir um rótulo de saída a partir de um determinado dado de entrada (RUSSELL; NORVING, 2013).

Quando a saída for um número, o problema de aprendizagem é chamado de *regressão*, segundo Dean (2014), consiste no uso de dados para prever, o mais próximo possível, os rótulos com valores reais corretos dos pontos ou itens considerados. A regressão tem muitas características úteis; sendo normalmente de fácil interpretação dos resultados (MOHRI; ROSTAMIZADEH; TALWALKAR, 2018).

3.1.1.2 Aprendizado Não-Supervisionado

No aprendizado não supervisionado, não existe um valor de rótulo ou de destino fornecido para os dados. A ideia é que o computador seja capaz de aprender sem orientá-lo especificamente quanto ao que fazer (HARRINGTON, 2012).

Segundo Rosa (2016), os dados fornecidos como entrada são analisados e agrupados conforme a proximidade dos seus valores. A tarefa na qual agrupamos itens semelhantes é conhecida como *clustering*, que consiste em agrupar dados em classes ou grupos de objetos que, quando comparados uns com os outros dentro de um mesmo grupo, possuem alta semelhança, e quando comparados com objetos de outro grupo, são bastante diferentes (DOMINGUES, 2003).

As técnicas de aprendizado não-supervisionado, segundo Pellucci et al. (2011), são utilizadas quando o objetivo é encontrar em um conjunto de dados, padrões que auxiliem o entendimento desses dados.

3.1.2 Abordagem Baseada em Conhecimento Explícito

Segundo Abel e Fiorini (2013), “a Engenharia de Conhecimento busca capturar o conhecimento de especialistas e formalizá-lo de maneira independente de modo a permitir sua apropriação e reuso em sistemas ou processos” (p.5). O conhecimento

capturado/adquirido é representado através de regras explícitas, que vão formar a Base de Regras (ou Base de Conhecimento) do sistema.

A EC se baseia em um processo interativo no qual o engenheiro de conhecimento constrói regras explícitas manualmente com base na observação de padrões existentes em um *corpus* de dados do domínio da aplicação. A criação das regras também pode levar em conta o conhecimento de um especialista no domínio da aplicação (LOPES, 2004; SILVA, 2003; ABEL; FIORINI, 2013).

De acordo com Pereira (2015), A EC “visa integrar o humano à tecnologia, ou melhor, o pensamento humano aos processos gerenciados por sistemas de informação” (p.25).

Os Sistemas Baseados em Conhecimento (SBC) se caracterizam por apresentar uma arquitetura modular, com dois módulos principais: a Base de Regras/Conhecimento; e a máquina ou mecanismo de inferência, responsável pela execução das regras a partir dos dados de entrada.

Para construção de SBCs, é necessário extrair o conhecimento de humanos especialistas na área e de estruturar este conhecimento em uma representação computacional que permita o processamento em computadores (SCARINCI, 1997). Geralmente, a representação desse conhecimento é feita com pares de *condição-ação*, as quais codificam o conhecimento para a tomada de decisão sob forma de regras. Para Felfering e Kostyantyn (2006), os SBCs são programas de computador que usam o conhecimento representado explicitamente para resolver problemas. Esses sistemas manipulam a informação de forma inteligente e são desenvolvidos para serem usados em problemas que requerem uma quantidade considerável de conhecimento humano (TICOM, 2007).

O ciclo básico de construção desses sistemas envolve quatro etapas principais, descritas abaixo. Note que essas etapas não incluem a construção da máquina de inferência (MI), uma vez que este não seria um trabalho do engenheiro de conhecimento, e também porque já existem diversas MI disponíveis para uso (SANTOS, 2011).

Abaixo temos as principais etapas de construção dos SBCs. Esse ciclo se repete até que o sistema apresente resultados satisfatórios.

(1) Aquisição de conhecimento – onde o engenheiro de conhecimento entrevista especialistas e estuda todo material disponível sobre o problema em foco;

(2) Formalização do conhecimento capturado/adquirido – representação preliminar do conhecimento adquirido, que será validada pelo especialista;

(3) Construção da base de regras – representação do conhecimento validado em forma de regras explícitas;

(4) Testes e validação – quando o SBC é executado a partir de um corpus de teste e os resultados são avaliados pelo engenheiro e pelo especialista. Caso o resultado não seja satisfatório, o EC volta à etapa (1) deste ciclo.

Em comparação com a abordagem baseada em AM, os SBCs têm um tempo de desenvolvimento mais longo. A etapa (1) é o maior gargalo nessa construção, pois nem sempre é simples capturar conhecimento em domínios muito complexos - como, por exemplo, diagnóstico médico.

Porém, em alguns casos, a AM não dá bons resultados. Isso ocorre quando não existem dados de qualidade disponíveis para o algoritmo de AM aprender/induzir regras automaticamente. Nesses casos, a melhor opção é a EC, apesar do maior tempo de desenvolvimento dos seus sistemas.

Uma vantagem dos SBCs em relação à AM é a possibilidade de atualizar as regras definidas, adaptando-as a variações dos dados de entrada e para novas aplicações semelhantes à aplicação original.

3.2 EXTRAÇÃO DE INFORMAÇÃO

A recuperação de informações é um processo impreciso. Além disso, o ser humano não consegue processar esse grande volume de informação disponível sem o auxílio de aplicações computacionais (ZAMBENEDETTI, 2002; ROSA, 2016).

O objetivo da Extração de Informação é construir sistemas que identifiquem informações relevantes enquanto ignoram informações insignificantes e irrelevantes. Segundo Álvarez (2007), EI é o processo de identificar e extrair informações específicas, ou seja, apenas os dados relevantes ao usuário, a partir de documentos textuais. Já para Kushmerick e Thomas (2003), EI é uma forma de processamento que envolve o preenchimento de um banco de dados com valores extraídos automaticamente de documentos textuais.

Com a aplicação de técnicas de Extração de Informação é possível extrair dados estruturados de textos não estruturados ou semiestruturados e transformá-los em conhecimento útil. Os conhecimentos úteis são aqueles que podem ser utilizados

para apoiar algum processo de tomada de decisões, seja por sistemas autômatos, sistemas especialistas ou até especialistas humanos (SOARES, 2008).

3.2.1 Métodos e Técnicas

Como já mencionado, sistema de EI textuais têm por objetivo extrair automaticamente informação útil a partir de documentos. Técnicas de Processamento de Linguagem Natural podem ser usadas para extrair informações a partir de textos, assim como para facilitar a entrada de dados nos sistemas e a estruturação dos dados, visando preparar os dados textuais sobre quais se busca algum tipo de conhecimento (ÁLVAREZ, 2007; SANTOS et al., 2014). De acordo com Aranha e Passos (2006), o PLN pode ser definido como “um campo da Ciência da Computação que abrange um conjunto de métodos para analisar textos através do uso de programas” (p.10).

Antes de iniciar o processo de EI, é necessário definir um modelo (formulário/*template*) com base nas informações específicas do domínio que se deseja extrair. O processo de EI deverá preencher campos (slots) do formulário definido, que poderá ser armazenado em um banco de dados, ou como colunas de uma planilha, ou ainda usados como entrada para outros módulos de um sistema maior, ou mesmo de outros sistemas de informação. Os dados extraídos também podem ser diretamente apresentados ao usuário final, através da interface do sistema. Em geral, a definição do *template* é feita manualmente por um especialista humano com conhecimento na área específica em que a EI será realizada. Dessa forma, espera-se minimizar erros no processamento, extraindo apenas informações relevantes.

A construção do módulo extrator pode ser feita com base nas técnicas de Aprendizagem de máquina (seção 3.1.1) ou Engenharia do conhecimento, usando regras explícitas (seção 3.1.2).

Segundo Silva (2003), o treinamento automático permite que o sistema aprenda a extrair informações submetendo um algoritmo de treinamento a um corpus de textos, onde as informações a serem extraídas são destacadas com padrões de extração. Após o treinamento, os algoritmos são executados no conjunto de documentos a partir dos quais se deseja extrair informações. Quando aplicado a novas coleções de textos, não são necessárias regras pré-definidas, apenas informação contida no conjunto de treinamento é usada no processamento (LOPES, 2004). Já na abordagem baseada em EC, um engenheiro do conhecimento constrói manualmente as regras de extração

que serão usadas para localizar e extrair os dados de interesse a partir dos documentos de entrada.

3.2.2 Ferramentas de EI

Nesta seção são apresentadas ferramentas utilizadas no processo de construção de sistemas de EI.

3.2.2.1 Expressões Regulares (RegEx)

Expressões regulares (do inglês, *Regular Expressions* – RegEx[1]) são usadas para se especificar um padrão de texto, que pode ser uma composição de símbolos, caracteres com funções especiais que, agrupados entre si e com caracteres literais, formam uma sequência (uma expressão). A expressão regular é interpretada como uma regra de extração que será executada com sucesso sempre que uma entrada de dados obedecer a todas as suas condições (JARGAS, 2012). RegEx são úteis para a identificação de datas, preços, números de telefone, CEP, entre outros (GONÇALVES, 2012).

Após a expressão definida, é realizada a busca do padrão na *string* que é passada como parâmetro. Por exemplo, se a expressão regular for (ac?|b*), as expressões “a”, “ac”, “abbb”, serão correspondentes (RIBEIRO, 2013). Essa ferramenta é comumente usada para criação de regras baseadas em conhecimento explícito, sendo muito utilizada no processo de EI aplicada a textos livres, por ser capaz de manipular *strings*.

3.2.2.2 WEKA – Waikato Environment for Knowledge Analysis

O WEKA é um software de aprendizado de máquina de código aberto que pode ser acessado por meio de uma interface gráfica do usuário, aplicativos de terminal ou APIs. É amplamente utilizado para ensino, pesquisa, aplicações industriais e contém uma grande quantidade de ferramentas de aprendizagem de máquina (WEKA, 2020). Esta é uma ferramenta muito utilizada no processo de Extração de Informações (EI)

[1] RegEx- https://regexr.com/

em bases textuais, e pode ser utilizada tanto para Aprendizado supervisionado, quanto para o Aprendizado não-supervisionado.

Essa ferramenta fornece implementações de algoritmos de aprendizado que podem facilmente ser aplicados a um conjunto de dados. Todos os algoritmos recebem sua entrada na forma de uma única tabela que pode ser lida de um arquivo ou gerada por uma consulta ao banco de dados. O conjunto de algoritmos oferecido é diversificado e abrangente, sendo acessado através de uma interface comum para que seus usuários possam comparar métodos diferentes e identificar os que são mais apropriados para o problema em mãos (FRANK; HALL; WITTEN, 2016).

3.2.3 Avaliação de Sistemas de EI

As principais medidas de desempenho utilizadas para avaliar sistemas de EI, segundo Zambenedetti (2002), são *Precisão* e *Cobertura* (do inglês, *Precision* e *Recall*). Essas duas medidas são normalmente utilizadas para descrever a habilidade de um sistema de EI em extrair corretamente todas as informações disponíveis nos dados de entrada.

Considerando um corpus de documentos de entrada e um dado *template* de saída, essas medidas de avaliação são calculadas individualmente para cada campo do *template* de saída. Para calcular essas medidas, três variáveis são consideradas (ZAMBENEDETTI, 2002):

> • #E expressa o número total de informações extraídas, isto é, o número total de vezes que o campo foi preenchido (correta ou incorretamente) pelo algoritmo de extração;
> • # EC indica o número total de vezes que o campo foi preenchido corretamente pelo processo de extração;
> • # ID indica o número de vezes que a informação a ser extraída ocorre nos documentos de entrada.

Para um dado campo do *template*, a *Precisão* se refere à capacidade do algoritmo de extrair corretamente a informação referente a esse campo, sendo calculada pela relação entre a quantidade de vezes que o campo foi preenchido

corretamente (#EC) e a quantidade de vezes que esse campo foi preenchido -correta ou incorretamente- pelo algoritmo de extração (#E) (SILVA, 2003). Veja Equação 1 abaixo.

$$Precisão = \frac{\#EC}{\#E}$$

(Eq. 1)

Para um dado campo do *template*, a Cobertura se refere à capacidade do algoritmo de extrair todas as ocorrências existentes da informação referente a esse campo nos documentos de entrada, sendo calculada pela relação entre a quantidade de informações extraídas corretamente (# EC) e a quantidade de informações disponíveis nos documentos de entrada para preencher esse campo (#ID) (SILVA, 2003). Veja Equação 2 abaixo.

$$Cobertura = \frac{\#EC}{\#ID}$$

(Eq.2)

Note que essas medidas são complementares. É importante ajustar o algoritmo de extração, de modo que ele seja capaz de extrair corretamente dos documentos de entrada, todas as ocorrências de cada campo do *template* de saída. Essa situação se configura quando temos P=R=1.

Na prática, contudo, à medida que o sistema é ajustado para aumentar a taxa de precisão, a cobertura cai (e vice-versa). Observe que quando as regras de extração são ajustadas para serem mais rigorosas na extração (a fim de melhorar a precisão), mais dados serão desconsiderados (piorando a cobertura); e quando as regras são mais flexíveis (para aumentar a cobertura), mais dados errados serão extraídos (piorando a precisão).

Nesse cenário, é importante observara média harmônica entre a *Precisão* e *Cobertura*, chamada de *F-measure*. Essa métrica é sensível as mudanças na distribuição dos resultados da precisão e cobertura (THARWAT, 2020), sendo capaz de identificar o ponto de equilíbrio na “calibragem” das regras dos sistemas de EI. Essa métrica é calculada a partir da seguinte fórmula (Eq. 3).

$$F - measure = 2\ x\ \frac{Precisão\ x\ Cobertura}{Precisão + Cobertura}$$

(Eq.3)

3.2.4 Exemplos de sistemas de EI

Nesta seção são apresentados exemplos trabalhos relacionados a sistemas de EI:

- Extração Semântica de Informações: O grande volume de dados em arquivos pessoais obtidos a partir da internet cria uma complexa tarefa de gerenciar os mesmos. Com o objetivo de selecionar e manipular essas informações, o SES – Sistema de Extração Semântica de Informações realiza a extração e sumarização dos dados, utilizando conceitos de EI e SBC, que permite a classificação dos dados extraídos em classes significativas para o usuário e a determinação da validade temporal destes dados a partir da geração de uma base de dados (SCARINCI, 1997).
- Extração de informações em artigos científicos: O objetivo específico do sistema FIP – Ferramenta Inteligente de Apoio à Pesquisa desenvolvido, é induzir de forma automática, um conjunto de regras para extração de informações de artigos científicos. O sistema de extração proposto, inicialmente, analisa e extrai informações presentes no corpo dos artigos (título, autores, afiliação, resumo, palavras chaves) e, posteriormente, foca na extração das informações de suas referências bibliográficas (ALVAREZ, 2007).
- Sistema de Extração de Informação com Interface para Linguagem Natural: A extração de informação relevante na internet é um grande desafio. Utilizando conceitos de EI, o objetivo do sistema InfoMovie é realizar a extração de dados estruturados baseados na ontologia “cinema”, fornecendo uma interface de linguagem natural para que o usuário possa encontrar informações relacionadas ao assunto (RIBEIRO, 2013).

3.3 CLASSIFICAÇÃO DE TEXTO

A Classificação de Texto tem o propósito de atribuir rótulos pré-definidos a documentos textuais (BRITO, 2016). Ela consiste em examinar as características (geralmente, palavras ou termos compostos) nos dados e atribuir uma classe previamente definida a cada documento de acordo com suas características (CÔRTES; PORCARO; LIFSCHITZ, 2002).

Esta é uma importante tarefa da MT, pois textos escritos de forma livre, como e-mails, jornais, livros ou páginas de internet, são fontes de informações valiosas. Classificar esses textos, ou parte deles, tem se tornado uma importante forma de aprimorar processos, incluindo a tomada de decisões (MATOS, 2020). Esta tarefa pode ser aplicada em diversos tipos de soluções, como detecção de spam, análise de sentimentos, entre outras (SANTOS et al., 2014).

Neste trabalho, a classificação de texto é aplicada na detecção de fraudes em NFS-e. Após a tarefa de EI, apresentada na seção anterior, é aplicada a classificação de texto para categorizar automaticamente NFS-e que possuem indícios de fraudes.

3.3.1 Métodos e Técnicas

A construção de classificadores de texto, assim como sistemas de EI, pode ser feita com base nas técnicas da Engenharia do conhecimento e de Aprendizagem de máquina.

Quando os textos são classificados em apenas uma categoria, a classificação é chamada de *single-label*. Nos casos em que os textos podem ser classificados em mais de uma categoria ao mesmo tempo, a classificação é chamada de *multi-label* (RODRIGUES, 2009).

3.3.1.1 Engenharia do Conhecimento

A EC, utilizando técnicas da abordagem baseada em conhecimento explícito, constrói o classificador seguindo as etapas apresentadas na seção 3.1.2. A formalização do conhecimento é feita com pares de condição-ação. Se uma condição

IF é consistente para o problema, o sistema continua a ação com a cláusula IF, tornando a ação ELSE caso a condição IF não seja considerada (TICOM, 2007).

Ainda segundo Ticom (2007), por intermédio desse método existem duas linhas de raciocínio que podem ser seguidas. A primeira é o encadeamento regressivo, que parte da suposição que cada provável solução é verdadeira, com isso, tenta-se comprovar ser correta a solução previamente considerada. A segunda é o encadeamento progressivo, no qual as informações são fornecidas ao sistema pelo usuário, que com suas respostas desencadeiam o processo de busca até encontrar a solução para o problema.

3.3.1.2 Aprendizagem de Máquina

Na abordagem baseada em AM, comumente são utilizados métodos de aprendizado supervisionado para induzir classificadores (ANDRADE, 2015). Segundo Finizola (2019), os principais algoritmos empregados para classificação textual são: *Support Vector Machines* (SVM), *K-Nearest Neighbors* (K-NN) e *Naive Bayes* (NB), além da Árvore de Decisão.

3.3.1.2.1 Support Vector Machines (SVM)

O objetivo do SVM é encontrar no espaço dimensional um hiperplano ideal, onde a distância que separaduas classes distintas seja maximizada (FINIZOLA, 2019). Dada as instâncias etiquetadas de treinamento, ao encontrar um hiperplano ideal, fica mais fácil a classificação de novos textos (MATOS, 2020). Segundo Rodrigo (2009), podem existir infinitos hiperplanos, e quanto maior a distância que separa as classes mais próximas, menor é o risco do algoritmo SVM classificar erroneamente um novo texto.

3.3.1.2.2 K-Nearest Neighbors (K-NN)

O objetivo do K-NN é agrupar instâncias de acordo com a proximidade que elas têm entre si. Os dois principais parâmetros a serem definidos no K-NN são o valor de k e a métrica de distância, onde a variável k determina a quantidade de instâncias

mais próximas que serão usadas para atribuir a qual classe a nova instância pertence (MATOS, 2020).

O algoritmo assume que as instâncias correspondem a pontos num espaço n-dimensional. A proximidade entre as instâncias é definida por uma métrica, comumente utiliza-se a distância euclidiana (MATOS, 2020). Para classificar uma nova instância x, o algoritmo atribui a x a classe mais frequente entre as k instâncias de treinamento mais próximas, ou seja, que tenham menor distância para x (RODRIGUES, 2009).

3.3.1.2.3 Naive Bayes (NB)

O algoritmo NB calcula a probabilidade da classe mais provável para o texto de entrada com base na fase de treinamento. É conhecido como um classificador ingênuo porque assume que os atributos das instâncias são independentes entre si. Quando um atributo indica ou não a ocorrência de um evento no texto, o modelo é chamado de binário. Quando se leva em consideração a quantidade de vezes em que cada evento ocorre no texto, o modelo é chamado de *multinomial* (ANDRADE, 2015).

3.3.1.2.4 Árvore de Decisão

De acordo com Funchal et al (2016), a árvore de decisão é um classificador que possui uma estrutura hierárquica, constituída por nós e arestas. A representação da árvore é algo simples, pois são nós encadeados com funções de decisão. Dado certo valor de entrada com seus atributos, vai se percorrendo a árvore pelos nós e em cada nó decidindo pelo atributo qual será o próximo nó, até chegar ao nó folha que terá o resultado (BERTOZZO, 2019).

De acordo com Rosa (2016), a árvore de decisão é um método bastante popular devido a simplicidade em se montar as árvores e devido a maioria dos algoritmos trabalhar com variáveis de entrada contínuas quanto categóricas. Além disso, as árvores de decisão não necessitam de nenhuma informação quanto à distribuição dos atributos de entrada ou da função-objetivo e são eficientes com o aumento do tamanho da base de dados utilizada.

Diante das técnicas apresentadas acima, é importante mencionar que não há um método ideal. Cada modelo pode apresentar resultados diferentes, de acordo com a aplicação. Além disso, existe também a possibilidade de se utilizar a combinação de técnicas diferentes.

3.3.2 Ferramentas

Existem diversas ferramentas disponíveis para a construção de classificadores, além de algumas linguagens de programação e outros softwares completos, que abstraem os recursos das linguagens de programação. Abaixo veremos alguns exemplos.

Antes de avançar nessa apresentação, é importante mencionar aqui a ferramenta *WEKA* para indução de classificadores. Essa ferramenta, já apresentada na seção 3.2.2, é utilizada com grande frequência para a construção de classificadores baseados em AM. Ela oferece diversos algoritmos de AM, incluindo os três algoritmos citados acima.

A linguagem de programação *Python* é uma das mais utilizadas na construção de classificadores de texto, sendo muito poderosa para processamento de dados. Essa linguagem oferece grande quantidade de bibliotecas, kits e recursos que podem diversificar seu uso. *Python* pode ser utilizada para construção de classificadores baseados em regras explicitas, através da utilização de comandos “IF-ELSE”, como também utilizando AM, através da biblioteca *scikit-learn*, que não é nativa do *Python* (precisa ser instalada) (BERTOZZO, 2019).

IBM Watson Studio provê uma interface através da qual o usuário pode criar modelos preditivos, usando a técnica de aprendizagem de máquina, com opções de arrastar-e-soltar. Com esta aplicação é possível reunir modelos de software livre como *scikit-learn* e também linguagens de programação como *Python* (BERTOZZO, 2019).

Microsoft Azure Machine Learning pode ser usada para qualquer projeto de aprendizado de máquina. É possível escrever códigos de programação ou criar os projetos utilizando uma área de trabalho mais amigável. Ela permite a integração com ferramentas como *scikit-learn* e *PyTorch* (MICROSOFT, 2020).

3.3.3 Avaliação de Classificadores

Geralmente, os classificadores são avaliados por medidas mais específicas do que Precisão e Cobertura apenas. São utilizadas 4 variáveis na computação de diversas medidas de desempenho. Considerando um classificador binário, as 4 variáveis base são (HAN; KAMBER; PEI, 2011):

• Verdadeiros Positivos (VP) - quantidade de documentos da classe positiva corretamente classificados pelo classificador;

• Verdadeiros Negativo (VN) - quantidade de documentos da classe negativa que foram corretamente classificados pelo classificador;

• Falsos Positivos (FP) - quantidade de documentos da classe negativa que foram erroneamente classificados como positivos pelo classificador;

• Falsos Negativos (FN) - quantidade de documentos da classe positiva que foram erroneamente classificados como negativos pelo classificador.

Essas variáveis são utilizadas para compor a Matriz de Confusão (Figura 12), que dá uma visão geral do número absoluto de acertos e erros do classificador.

Figura 12 - Matriz de Confusão

	Classificação Manual	
Classificação Automática	**Positivas**	**Negativas**
Positivas	Verdadeiros Positivos (VP)	Falsos Positivos (FP)
Negativas	Falsos Negativos (FN)	Verdadeiros Negativos (VN)

Fonte: O autor (2020)

Diversas são as métricas que são utilizadas para avaliação de desempenho de classificadores a partir da observação da matriz de confusão. Uma das principais medidas é a *acurácia*, calculada conforme a Equação 4, que apresenta uma avaliação geral dos acertos do classificador em relação ao total de documentos existentes no corpus (NASCIMENTO, 2019).

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN}$$
$$(Eq. 4)$$

A *precisão*, é a proporção de verdadeiros positivos em relação a todas as predições positivas. Esta medida, que é calculada conforme a Equação 5, pode facilmente induzir a uma conclusão errada sobre o desempenho do sistema por ser suscetível a desbalanceamentos do conjunto de dados (MATOS, 2020).

$$Precisão = \frac{VP}{VP + FP}$$
$$(\text{Eq. } 5)$$

A *sensibilidade*, é a proporção de verdadeiros positivos em relação ao total de positivos. Esta medida, que é calculada conforme a Equação 6, tem a capacidade de predizer corretamente exemplos da classe positiva (MATOS, 2020).

$$Sensibilidade = \frac{VP}{VP + FN}$$
$$(\text{Eq. } 6)$$

A *F1-Score*, é a combinação da precisão e sensibilidade que indica a qualidade geral do modelo. Esta medida, que é calculada conforme a Equação 7, apresenta bons resultados até com conjuntos de dados desbalanceados de exemplos das classes positiva e negativa (MATOS, 2020).

$$F1 - Score = \frac{2 * precisão * sensibilidade}{precisão + sensibilidade}$$
$$(\text{Eq. } 7)$$

3.3.4 Exemplos de Sistemas de Classificação

A classificação de textos é uma técnica bastante explorada. Diversas aplicações e usos diferentes são encontrados na literatura. A seguir, temos alguns exemplos de aplicações:

- Classificação de Riscos: O estudo realizado por Funchao et al. (2016) utiliza técnica de *Árvore de Decisão* para realizar classificação de riscos em unidades de pronto atendimento. A classificação de riscos é a triagem do paciente de acordo com os sintomas apresentados. Inicialmente são preenchidos formulários e, ao final, o sistema classifica os pacientes em cores, de acordo com a urgência de atendimento.
- Triagem automática de denúncias: O estudo realizado por Andrade (2015), utiliza métodos de aprendizado de máquina para realizar a triagem de denúncias registradas na Controladoria Geral da União (CGU). As denúncias são provenientes de todos os estados da Federação, sendo classificadas manualmente e encaminhadas para 91 destinos diferentes. Com este estudo o autor demostrou a viabilidade de realizar a triagem de forma automática, sem perda de qualidade quando comparada à triagem realizada manualmente.
- Análise de *feedbacks*: Finizola (2019), estudou diversos métodos de aprendizagem de máquina para classificar *feedbacks*, relatando problemas, críticas, defeitos e sugestões em produtos de telefonia. Testadores de software, realizam análise de forma manual e utilizam esses *feedbacks*, para melhorar a qualidade dos testes exploratórios. O autor criou um classificador para automatizar as atividades de análise, a fim de obter informações úteis de forma mais ágil.

3.4 CONSIDERAÇÕES FINAIS

A busca por conhecimento em documentos textuais é um processo complexo devido aos dados não possuírem estrutura definida. A aplicação de técnicas de Extração de Informação e Classificação de textos é muito importante nesse processo

Como visto, as principais técnicas da EI e da Classificação de textos são baseadas em Aprendizagem de Máquina e em Regras baseadas em conhecimento explícito. Uma dificuldade na utilização da AM nesse contexto se refere à grande quantidade de documentos necessários, como exemplos das informações que se deseja extrair, para treinar o algoritmo. Já a maior dificuldade na utilização da técnica de Regras baseadas em conhecimento explícito se refere à necessidade de um especialista humano, com grande conhecimento nos dados.

Contudo, essas técnicas podem ser aplicadas em diversas áreas do conhecimento e é importante considerar que não apenas o algoritmo, mas também a origem dos dados, são muito importantes para a qualidade dos resultados obtidos a partir do sistema implementado.

4 AUDITA-NFSE: SISTEMA AUXILIAR DE AUDITORIA EM NOTAS FISCAIS DE SERVIÇOS ELETRÔNICAS

“Auditoria é a atividade de análise de demonstrações financeiras, documentos, processos e procedimentos de uma entidade empresarial ou pública, para emissão de relatório, parecer sobre sua aderência ou não a legislação vigente”. (SANTOS, 2020, p.18).

“Auditoria fiscal é o ramo da auditoria voltado para a análise do correto cumprimento das obrigações tributárias pelos contribuintes” (SANTOS, 2015, p.29). Uma das principais obrigações tributárias para prestadores de serviços é a emissão da nota fiscal de serviços eletrônica (NFS-e).

“A nota fiscal é o instrumento utilizado pelo governo federal, estadual e municipal para a arrecadação de impostos e para a regulamentação das operações de vendas e/ ou prestações de serviços” (ARAUJO, 2019, p.12).

A auditoria de NFS-e por parte do Fisco Municipal analisa todas as informações que foram declaradas pelo contribuinte na nota fiscal, a fim de identificar indícios de fraudes por parte dos prestadores de serviço. Algumas dessas informações incluem o tipo de serviço prestado, discriminação dos serviços realizados, período de realização dos serviços, valor dos impostos, entre outras.

Este projeto de mestrado teve como objetivo a criação de um sistema que auxilie os Auditores Fiscais Tributários durante o processo de análise e identificação de possíveis fraudes nas NFS-e emitidas pelos contribuintes.

4.1 ETAPAS DE CONSTRUÇÃO DO SISTEMA

Para a construção do sistema Audita-NFSe, foi necessário elicitar/adquirir conhecimento junto aos auditores da área de fiscalização tributária e estruturar este conhecimento em forma de regras. Para realizar este processo, foi necessário seguir algumas etapas, descritas nas seções a seguir.

4.1.1 Aquisição de Conhecimento

Nesta etapa foram realizadas entrevistas com auditores do Município do Ipojuca/PE, que fazem parte do quadro de servidores há mais de 8 anos e que atuam na área de fiscalização do Imposto Sobre Serviços.

Nas entrevistas, foram realizadas perguntas com o propósito de extrair o conhecimento dos auditores no processo de auditoria. Essas entrevistas trataram dos seguintes tópicos: principais procedimentos para realização de uma fiscalização; principais informações analisadas em uma NFS-e; definição da área a ser fiscalizada; identificação das leis que regulamentam a prestação de serviços; formas de identificar possíveis fraudes em NFS-e; e problemas enfrentados no processo de fiscalização.

Como resultado das entrevistas, ficou definido que a área de Construção Civil seria escolhida como objeto de estudo. Foi definido um período entre os anos de 2010 e 2013, num total de 3080 NFS-e de empresas que já haviam sido fiscalizadas e que foram autuadas por fraudes identificadas nas NFSe, para serem utilizadas para identificar as informações relevantes utilizadas no processo de fiscalização. Não foi possível utilizar notas mais recentes para construção e testes do sistema porque era necessário escolher notas já auditadas, a fim de possibilitar uma avaliação concreta dos resultados obtidos com o Audita-NFSe.

Outros problemas identificados durante as entrevistas se referem ao processo de auditoria atual, que é realizado manualmente, e à forma de identificação de fraudes. Esses problemas serão detalhados na seção a seguir.

4.1.1.1 Processo manual de auditoria

Em um trabalho de auditoria, é necessário o cumprimento de etapas que incluem: intimação, declaração de recebimento de documento, abertura de processo de fiscalização, levantamento fiscal, lançamento da notificação prévia, declaração de devolução de documento, encerramento da auditoria, mediante notificação de valores de crédito, termo de encerramento de auditoria (MUNHOZ, 2020).Esses documentos, que são imprescindíveis para a fiscalização, incluem as Notas Fiscais de Serviços, Declarações de serviços, e ainda relatório dos recolhimentos mensais do ISS (PETRI, 2016 apud TAUIL, 2003).

Com o advento da tecnologia, esses documentos foram substituídos por versões eletrônicas. As NFS-e utilizadas no processo de auditoria podem ser visualizadas e impressas uma a uma, ou exportadas em grande quantidade em formato XML ou CSV, a partir do portal na web, para serem analisadas pelo Auditor.

A análise dessas notas é um processo que demanda muito tempo, pois, nos municípios que não possuem sistemas auxiliares, ela é feita de forma manual pelos auditores. Em pesquisa realizada no portal de emissão de Notas Fiscais de Serviços Eletrônicas dos Municípios de Ipojuca/PE (Figura 13), Recife/PE (Figura 14), São Paulo/SP (Figura 15) e Rio de Janeiro/RJ (Figura 16), é possível verificar a quantidade de notas fiscais emitidas.

Figura 13 - NFS-e da Prefeitura do Ipojuca/PE

Fonte: Prefeitura do Ipojuca (2020)

Figura 14 - NFS-e da Prefeitura do Recife/PE

Fonte: Prefeitura do Recife (2020)

Figura 15 - NFS-e da Pref. de São Paulo/SP

Fonte: Prefeitura de São Paulo (2020)

Figura 16 - NFS-e da Pref. do Rio de Janeiro/RJ

Fonte: Prefeitura do Rio de Janeiro (2020)

Esses números ilustram uma das principais dificuldades enfrentadas pelos auditores no processo de análise, que é a grande quantidade de notas fiscais

eletrônicas a serem auditadas. Os números informados nas Figuras 13, 14, 15 e 16 não são fixos, crescem diariamente com a continuidade de emissão de notas por parte dos prestadores de serviço.

4.1.1.2 Evidência de fraudes

De acordo com o Art. 72 da Lei 4.502, de 30 de novembro de 1964, “fraude é toda ação ou omissão dolosa tendente a impedir ou retardar, total ou parcialmente, a ocorrência do fato gerador da obrigação tributária principal, ou a excluir ou modificar as suas características essenciais, de modo a reduzir o montante do imposto devido a evitar ou diferir o seu pagamento”.

Ainda de acordo com Lima e Viana (2020), “a fraude fiscal consiste na utilização de procedimentos que violem de forma direta a lei ou o regulamento fiscal, no qual o contribuinte age com objetivo de favorecer a si ou terceiros” (p.33). Nesse contexto, o auditor tem uma função extremamente nobre, que é a de atuar como agente preventivo e inibidor contra fraudes (SANTOS 2015).

No Município do Ipojuca, a Lei nº 1.502 de 12 de novembro de 2008 disciplina a concessão de incentivos fiscais a empresas do ramo da construção civil e outros relacionados para implantação, ampliação, modernização ou diversificação de um estabelecimento industrial. Esta lei atribui isenção total de ISS a uma determinada lista de serviços e uma redução a 2% da alíquota do ISS sobre os demais serviços.

Assim, não seguir essas determinações pode indicar procedimentos fraudulentos. Esses procedimentos podem ser identificados por um auditor fiscal no momento da análise de uma NFS-e. São exemplos dessas evidências: Quando o período da prestação de serviço é diferente do informado no campo da competência do serviço; quando a alíquota do ISS para o serviço prestado é menor do que o definido em legislação.

4.1.2 Formalização do Conhecimento / Construção da base de regras

Diante das informações coletadas na etapa de aquisição de conhecimento, foi identificado que grande parte das informações relevantes no processo de auditoria presentes nas NFS-e, estão disponíveis em formato de texto livre. Com isso, para a

construção do sistema Audita-NFSe, foram definidas regras para a extração dessas informações relevantes e também foram definidas regras de classificação textual responsáveis pela detecção dos indícios de fraudes, apresentadas a seguir.

4.1.2.1 Extração de Informações das NFS-e

A partir desse conhecimento adquirido, foram criadas as 6 (seis) regras descritas a seguir, para extração de informações importantes e úteis ao processo de auditoria. São elas:

Regra (1) - Extração do período de prestação do serviço: Foi identificado nas notas fiscais a informação do período de prestação do serviço, composto por uma data inicial e uma data final. Esta informação é importante para fiscalização, pois, estabelece o período de competência para cobrança do ISS. Esta regra extrai esse período, e o resultado é utilizado no processo de classificação.

Regra (2) - Extração do número do contrato: Foi identificado nas notas fiscais a informação do número do contrato firmado entre a empresa prestadora e a tomadora do serviço. Esta informação é importante, pois, no decreto nº 08 de 06 de abril de 2010, que regulamenta a Lei 1.502 de 12 de novembro de 2008 do Município do Ipojuca/PE, em seu Art. 4º diz que, para fruição de incentivo fiscal a empresa prestadora deverá apresentar requerimento acompanhado do contrato firmado entre as partes. Esta regra extrai o número do contrato. Esta informação não é utilizada para realizar nenhuma inferência no sistema, pois, os detalhes do contrato não constam na NFS-e, contudo, é útil no processo de auditoria por parte de um auditor fiscal.

Regra (3) - Extração do número do Relatório de Medição (RM): Foi identificado nas notas fiscais a informação do número do relatório de medição. Este relatório aponta o que está no planejamento da obra e o que já foi executado. Esta regra extrai o número do relatório de medição. Esta informação não é utilizada para realizar nenhuma inferência no sistema, pois os detalhes deste relatório não constam na NFS-e. Contudo, essa informação é útil no processo de auditoria, pois pode ser solicitado que a empresa apresente este relatório para ser analisado por um auditor fiscal.

Regra (4) - Extração do número da Nota de Liquidação (NL): Foi identificado nas notas fiscais a informação do número da nota de liquidação. Esta nota é um documento sem valor fiscal, é similar a um recibo. Esta regra extrai o número da Nota de Liquidação. Esta informação não é utilizada para realizar nenhuma inferência no sistema, pois, os detalhes deste relatório não constam na NFS-e, mas e útil no processo de auditoria, pois, pode ser solicitado que a empresa apresente este relatório para ser analisado por um auditor fiscal.

Regra (5) - Extração de valores: Foi identificado no campo observação da nota fiscal que são informados valores que são referentes aos serviços prestados. Esta regra extrais os valores existentes na nota. Esta informação não é utilizada para realizar nenhuma inferência no sistema, mas pode ser utilizada no processo de auditoria.

Regra (6) - Extração de serviços sem isenção: Foi identificado nas notas fiscais a descrição de serviços que não fazem parte da lista de serviços com isenção de ISS definida no Art. 2º da Lei 1502 de 12 de novembro de 2008 do município do Ipojuca/PE. Esta regra extrais o nome desses serviços. Essa informação é utilizada no processo de classificação.

4.1.2.2 Classificação de NFS-e / Detecção de Fraudes

Este processo se baseia em 4 (quatro) regras construídas a partir de conhecimento de especialistas e de leis que regulam a prestação de serviços. Elas têm o objetivo de identificar indícios de fraudes nas NFS-e, classificando-as entre “possível fraude” ou não. São elas:

Regra (1) - Competência de prestação do serviço x Competência da NFS-e: De acordo com o Art. 1º da Lei complementar Nº 116, de 31 de julho de 2003, o ISS tem como fato gerador a prestação de serviços, ou seja, o valor do imposto deve ser calculado no período que o serviço foi realizado. Assim, o período de prestação do serviço deve ser o mesmo da competência da NFS-e. Esta regra compara a data final de prestação do serviço, que foi extraída e a compara com a competência da NFS-e. Em qualquer divergência, esta regra deve acumular pontuação de fraude.

Regra (2) - Valor dos serviços x Base de Cálculo: A Lei nº 1551, de 11 de janeiro de 2010 do município do Ipojuca/PE, trata da dedução (diminuição) do valor da base de cálculo para pagamento do ISS. Para que seja permitida a dedução, é necessário que os materiais aplicados aos serviços sejam produzidos pela empresa fora do local de prestação, e não adquiridos de terceiros. Qualquer divergência entre o valor total dos serviços e o valor da base de cálculo para notas sem isenção total deve ser classificada como indício de fraude. Esta regra verifica o valor dos serviços da nota, e o compara com o valor da base de cálculo do ISS. Caso o resultado dessa comparação seja falso, esta regra acumula pontuação de fraude.

Regra (3) - Serviços sem isenção de ISS: O Art. 2º da Lei 1502, de 12 de novembro de 2008 do município do Ipojuca/PE, descreve determinados serviços que possuem isenção total do ISS. Qualquer serviço que não faça parte dessa lista for encontrado em uma NFS-e com isenção total de ISS, deve-se considerar indício de fraude.
Esta regra verifica se há um ou mais serviços que não possuem isenção total de ISS extraídos da nota. Caso o resultado dessa verificação seja verdadeiro, esta regra acumula pontuação de fraude.

Regra (4) - Notas emitidas com isenção fora do período: A Lei 1502, de 12 de novembro de 2008 do município do Ipojuca/PE, atribui isenção total de ISS ao período de implantação de uma empresa. Qualquer nota fiscal emitida com isenção total de ISS fora desse período deve ser classificada como indício de fraude. Esta regra captura a data de emissão da NFS-e, e compara com a data final do período de implantação do empreendimento, que é informado manualmente, e então verifica o valor do ISS da NFS-e. Qualquer nota emitida com isenção total de ISS após a data final do período de implantação, acumula pontuação de fraude

4.1.3 Testes e Validação

Após a criação das regras de extração e classificação, foram realizados testes com as 3080 notas emitidas entre os anos de 2010 e 2013 selecionadas. Neste processo foi possível identificar se o sistema processou corretamente todas as regras, comparando o resultado do processamento com as informações contidas nas NFS-e.

A cada informação extraída ou classificada de forma errada pelo sistema, foi necessário voltar a etapa inicial de aquisição de conhecimento, realizar diversos ajustes nas regras de extração e classificação afim de melhorar o funcionamento do sistema e realizar novos testes.

Após a conclusão da criação e ajustes das regras e dos testes de processamento, os resultados foram apresentados aos auditores fiscais para validação. O resultado desta validação foi positivo, pois, além de realizar uma avaliação do processamento das regras de extração do sistema, com as informações contidas nas NFS-e, foi possível comparar o resultado da classificação realizada pelo Audita-NFSe, com o trabalho já realizado pelos próprios auditores fiscais no *corpus* de notas selecionado.

4.2 SOLUÇÃO PROPOSTA

Considerando os fatos apresentados nas seções anteriores, foi sugerida uma solução computacional capaz de automatizar parte do trabalho manual do auditor fiscal pertinente à análise de NFS-e e à identificação de possíveis fraudes. Esta seção apresenta a visão geral e o processo proposto para a solução criada, descrevendo os principais módulos do algoritmo.

4.2.1 Visão geral e objetivos

Em busca de automatizar parte do processo manual de análise das NFS-e, foi elaborada uma solução computacional que tem como objetivo auxiliar auditores fiscais a identificar possíveis fraudes através das seguintes tarefas: Importação do corpus de notas fiscais; Extração de Informações; Classificação de possíveis fraudes; e Exportação dos resultados em um *template* de saída. Os resultados oferecidos pelo sistema podem ser usados diretamente por um auditor fiscal ou podem ser utilizados como entrada para outros processos.

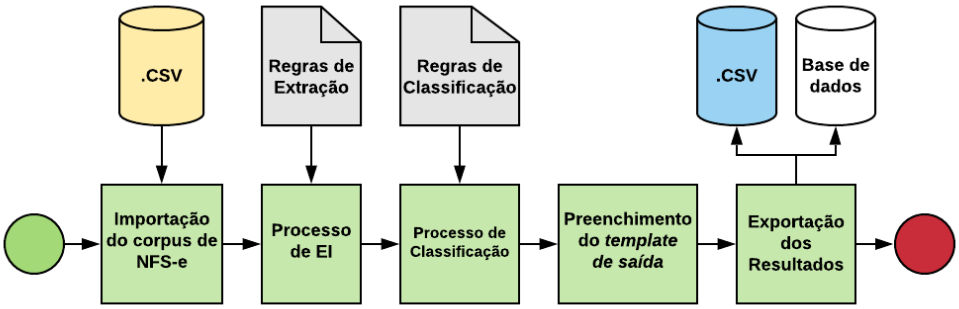
Este processo foi implementado em um sistema computacional, nomeado de “Audita-NFSe – Auditoria de notas fiscais de serviços eletrônica”. Utilizando conceitos da Mineração de Textos (MT) na aplicação de técnicas de Extração de Informação (EI) e Classificação de texto, o processamento do sistema é baseado em regras

criadas a partir do conhecimento de especialistas humanos em auditoria tributária e de leis específicas que regulam a prestação de serviços, para extrair dados importantes e classificar as possíveis fraudes existentes em um conjunto de NFS-e do município do Ipojuca/PE.

4.2.2 Etapas de processamento (arquitetura)

A Figura 17 apresenta o processo geral do protótipo do sistema proposto considerando os módulos principais: Importação do corpus de NFS-e; Processo de EI; Processo de Classificação; Preenchimento do *template*; Exportação dos Resultados.

Figura 17 - Diagrama BPMN do protótipo Audita-NFSe



Fonte: O autor (2020)

• Importação do corpus de NFS-e:

O corpus de NFS-e é importado para o sistema em um arquivo em formato csv, exportado a partir do portal web de emissão de notas fiscais do município do Ipojuca/PE.

O arquivo gerado possui um *template* definido, formado pelas principais informações contidas em NFS-e. Para este trabalho foram selecionados os campos para importação que constam no Quadro 01.

Quadro 1 - *Template* de importação das NFS-e

CAMPO	DESCRIÇÃO
cnpj_prestador	Números do CNPJ do prestador do serviço
nome_prestador	Nome da razão social do prestador de serviço
cnpj_tomador	Números do CNPJ do tomador do serviço
nome_tomador	Nome da razão social do tomaor do serviço
numero_nota	Número da NFS-e
mes_comp	Mês de competência da NFS-e
ano_comp	Ano de competência da NFS-e
data_nota	Data de emissão da NFS-e
data_competencia	Data da competência da NFS-e
valor_total	Valor total liquido da NFS-e
base_calculo	Valor da base de cálculo dos impostos da NFS-e
Aliquota	Alíquota do ISS (imposto sobre serviços) da NFS-e
Imposto	Valor do imposto do ISS
local_servico	Local de prestação do serviço
Situação	Situação da nota fiscal (ex: Normal, Isenta, Simples Nacional)
responsavel_imp	Responsável pelo pagamento do imposto
Atividade	Código da atividade econômica (CNAE) da NFS-e
descricao_servico	Descrição do serviço prestado
Observação	Observações referentes a NFS-e

Fonte: O autor (2020)

É importante destacar que os campos que formam a NFS-e, podem mudar de nomenclatura, ou possuir diversos outros campos distintos, de acordo com o sistema utilizado no município.

• Processo de EI:

Responsável por extrair informações dos campos *descricao_servico* e *observação*, este processo é realizado através da execução de 6 (seis) regras de extração baseadas em conhecimento explícito. As regras deste processo foram descritas com mais detalhes na seção 4.1.2.1 deste capítulo, descrevendo as informações importantes que cada regra deve extrair das NFS-e. A implementação é descrita no capítulo 5 deste trabalho.

• Processo de Classificação de Texto:

Responsável por classificar a possibilidade de existência de fraudes nas NFS-e, este processo é realizado após o processo de EI, com a execução de 4 regras de classificação. Essas regras foram criadas com base no conhecimento de especialistas e de leis específicas, conforme conceitos descritos na seção 4.1 deste trabalho. As regras deste processo são descritas na seção 4.1.2.2 deste capítulo. A implementação é descrita no capítulo 5 deste trabalho.

• Preenchimento do *template* de saída:

Após a execução do processo de EI e do processo de classificação, os resultados são utilizados para preencher o *template* de saída, conforme o Quadro 02. Este *template* além de incluir os campos do *template* de entrada, possui os campos comp_serv, contrato, rm, soma_valores, sem_isencao, onde são armazenados os resultados das regras de EI, e os campos classificação e indícios, onde são armazenados os resultados da classificação.

Quadro 2 - *Template* de saída do sistema Audita-NFSe

CAMPO	**DESCRIÇÃO**
Id	Número de identificação da NFS-e
cnpj_prestador	Números do CNPJ do prestador do serviço
nome_prestador	Nome da razão social do prestador de serviço
cnpj_tomador	Números do CNPJ do tomador do serviço
nome_tomador	Nome da razão social do tomador do serviço
numero_nota	Número da NFS-e
mes_comp	Mês de competência da NFS-e
ano_comp	Ano de competência da NFS-e
data_nota	Data de emissão da NFS-e
Competência_nota	Data da competência da NFS-e
valor_total	Valor total liquido da NFS-e
base_calculo	Valor da base de cálculo dos impostos da NFS-e
Alíquota	Alíquota do ISS (imposto sobre serviços) da NFS-e
Imposto	Valor do imposto do ISS
local_servico	Local de prestação do serviço
Situação	Situação da nota fiscal (ex: Normal, Isenta, Simples Nacional)

responsavel_imp	Responsável pelo pagamento do imposto
Atividade	Código da atividade econômica (CNAE) da NFS-e
descricao_servico	Descrição do serviço prestado
Observação	Observações referentes a NFS-e
comp_servico	Resultado da extração do período do serviço
Contrato	Resultado da extração do número de contrato
Rm	Resultado da extração do RM (relatório de medição)
Nl	Resultado da extração do número da Nota de Liquidação
soma_valores	Resultado da soma dos valores do campo observação
sem_isencao	Resultado da extração dos serviços sem isenção de ISS
Classificação	Resultado da classificação (valores de 0 a 4)
Indícios	Resultado da classificação com possíveis indícios de fraudes
Auditada	Informa se a NFS-e foi auditada

Fonte: O autor (2020)

• Exportação dos Resultados:

Após a execução de todas as etapas anteriores, o sistema salva o resultado em um banco de dados, possibilitando assim a exportação dos resultados do processamento em um formato .csv. Esse formato permite uma melhor visualização e também facilita a manipulação dos dados por um auditor fiscal, pois pode ser aberto nos principais softwares de planilha existentes no mercado.

4.3 CONSIDERAÇÕES FINAIS

A solução proposta teve como meta proporcionar um ganho de tempo na análise e identificação de possíveis fraudes existentes nas NFS-e, através da utilização das técnicas de Extração de Informações (EI) aplicada aos campos escritos em texto livre da Classificação de textos.

O sistema protótipo foi desenvolvido com o intuito de automatizar boa parte dos procedimentos realizados manualmente pelos auditores fiscais durante as análises de NFS-e emitidas por empresas da área de construção civil, no processo de auditoria tributária.

O protótipo possui 5 processos principais, que consistem em: importação do corpus de NFS-e; processo de EI; processo de classificação; preenchimento do

template; exportação dos resultados. As demais funcionalidades são internas, como o pré-processamento para tratamento dos dados antes de iniciar o processo de Extração de Informação.

É importante destacar que os processos definidos levaram em conta o fluxo de atividades manuais realizadas pelos auditores fiscais. Essa iniciativa foi útil para facilitar a familiaridade dos auditores com o protótipo proposto.

5 IMPLEMENTAÇÃO E TESTES

Este capítulo apresenta detalhes da implementação das funcionalidades principais do protótipo, isto é, as funcionalidades individuais dos módulos citados no capítulo 4. Também serão apresentados os testes e resultados do processo de Extração de Informações e da Classificação de texto, de modo a verificar o ganho de produtividade com ouso do protótipo na análise e identificação de possíveis fraudes em Notas Fiscais de Serviços Eletrônicas.

5.1 O PROTÓTIPO

O protótipo proposto, denominado “Audita-NFSe – Sistema Auxiliar de Autoria em Notas Fiscais de Serviços Eletrônicas”, foi desenvolvido com uso da linguagem de programação *Python* versão 3, com a utilização das bibliotecas *open-source* para pré-processamento de textos (*unicodedata.normalize* , *BeautifulSoup* e *datetime*), para extração de informação (*re*), para importação e exportação dos dados no formato de planilha (csv), desenvolvimento da Interface Gráfica do Usuário (*tkinter*), gerenciador de Banco de Dados (*sqlite3*) e criação de gráficos gerenciais (*matplotlib*).

A seguir serão apresentados detalhes de cada etapa do protótipo proposto e partes da aplicação “Audita-NFSe”.

5.1.1 Importação do corpus de NFS-e

Inicialmente, foram coletados dados no portal de emissão de notas fiscais da Prefeitura do Ipojuca/PE. Essas NFS-e são exportadas pelo sistema da prefeitura no formato CSV.

Esse arquivo é composto por vários campos, onde cada campo possui um tipo de informação presente na NFS-e (Quadro 01 da seção 4.2.2), e por várias linhas, sendo cada linha composta por uma nota fiscal. Alguns desses campos são utilizados apenas para exibição e outros são utilizados nos processos de Extração de Informação e Classificação de texto.

Essas informações são importadas para o sistema utilizando a biblioteca csv do Python através do código fonte exibido na Figura 18, que importa todo arquivo para a memória do sistema.

Figura 18 - Código Fonte de importação das NFS-e

```
with open('notas2014a2015.csv', newline='') as csvfile:
    spamreader = csv.DictReader(csvfile, delimiter=';')
```

Fonte: O autor (2020)

Após os dados serem carregados, se inicia o processo de Extração de Informações, descrito na próxima seção.

5.1.2 Processo de Extração de Informações

O sistema realiza a leitura de cada nota por vez. Antes de iniciar o processo de extração, os dados são pré-processados para tentar reduzir erros na EI, uma vez que os dados brutos são escritos em texto livre. Além disso, os dados possuem códigos HTML no corpo do texto, pois foram coletados de uma página na web.

O pré-processamento é realizado nos campos observação e *descricao_servico* utilizando a biblioteca *BeautifulSoup*[2], para remover os códigos HTML do texto. Em seguida, é realizada a normalização do texto utilizando a biblioteca *unicodedata.normalize*[3], para remover acentos e caracteres especiais do texto.

Após o pré-processamento, são executadas as 6 regras de extração descritas na seção 4.3 deste trabalho. Essas regras foram criadas utilizando a abordagem baseada em conhecimento explícito, com o suporte da biblioteca *re*[4] (Regular Expression) do Python, para extrair informações importantes dos textos.

As 6 regras de EI listadas abaixo são executadas nos campos de *observacao* e *descricao_servico*. Na figura 19, abaixo, temos um exemplo do campo observação de uma NFS-e, onde essas regras são aplicadas.

[2] https://pypi.org/project/beautifulsoup4/
[3] https://docs.python.org/3/library/unicodedata.html
[4] https://docs.python.org/3/library/re.html

Figura 19 - Exemplo do campo observação de uma NFS-e

Contrato No 0100.0000008.01-01; R3 1600511000
Documentos de Liberacao: RM 1, NL 0500001
Construcao Civil R$ 389.715,62 c/Reajuste (RC 20550000; PC 4505700000; FRS1009000000)
Periodo de Medicao: 26/12/13 a 25/01/14
Isencao de ISSQN conforme artigo 2o, inciso 1o, item f, da Lei no 1502 de 12 de novembro de 2008, municipio de Ipojuca - PE
Nao se aplica a retencao de INSS conforme artigo 176, inciso II da IN MPS/SRP 3/2005, exceto para atividade de montagem eletromecanica e condicionamento

Fonte: O autor (2020)

Regra (1) - Extração do período de competência da prestação do serviço: Parte do código utilizado para extrair a competência do serviço contida no campo *observação* é mostrado na Figura 20 abaixo. O sistema identifica padrões de escrita do período de competência da prestação do serviço (ex: 26/11/13 a 25/12/13) e armazena em uma variável que é utilizada pelo sistema para exportar seu conteúdo para uma base de dados ou como planilha no formato *csv*.

Figura 20 - Código Fonte para extração do período de competência de prestação do serviço

```
comp_servico = re.findall('\d{2}\.\d{2}\.\d{2,4}.a.\d{2}\.\d{2}\.\d{2,4}', text,re.I)
if comp_servico == []:
    comp_servico = re.findall('\d{2}\/\d{2}\/\d{2}.a.\d{2}\/\d{2}\/\d{2}', text, re.I)
    if comp_servico == []:
        comp_servico = re.findall('\d{2}\/\d{2}\/\d{2}.à.\d{2}\/\d{2}\/\d{2}', text, re.I)
```

Fonte: O autor (2020)

Regra (2) - Extração do número do contrato: Parte do código utilizado para extrair o número do contrato contido no campo observação é mostrado na Figura 21 abaixo. O sistema identifica padrões de escrita do número de contrato (ex: 0100.0000008.01-01) e armazena em uma variável que é utilizada pelo sistema para exportar seu conteúdo para uma base de dados ou como planilha no formato csv.

Figura 21 - Código fonte para extrair o número do contrato

```
contrato = re.findall('\d{4}\.\d{7}\.\d{2}\.\d{1}',text)
if contrato == []:
    contrato = re.findall('\d{1}\.\d{3}\.\d{7}\.\d{2}\-\d{1}',text)
    if contrato == []:
        contrato = re.findall('\d{4}\.\d{7}\.\d{2}\-\d{2}', text)
```

Fonte: O autor (2020)

Regra (3): Extração do número do Relatório de Medição: Parte do código utilizado para extrair o número do relatório de medição é mostrado na Figura 22. O sistema identifica padrões de escrita do número do Relatório de Medição (ex: RM 1) e armazena em uma variável que é utilizada pelo sistema para exportar seu conteúdo para uma base de dados ou como planilha no formato csv.

Figura 22 - Código fonte para extrair o número do RM (Relatório de Medição)

```
rm = re.findall('RM.\d{1,3}|RM..\d{1,3}|RM.no.\d{1,3}',norm, re.I)
```

Fonte: O autor (2020)

Regra (4) - Extração do número da Nota de Liquidação: Parte do código utilizado para extrair o número da nota de liquidação é mostrado na Figura 23 abaixo. O sistema identifica padrões de escrita do número da Nota de Liquidação (ex: NL 0500001) e armazena em uma variável que é utilizada pelo sistema para exportar seu conteúdo para uma base de dados ou como planilha no formato csv.

Figura 23 - Código fonte para extrair o número da NL (Nota de Liquidação)

```
nl = re.findall('NL.00000\d{6,7}|NL..00000\d{6,7}|NL.:.00000\d{6,7}',text)
if nl == []:
    nl = re.findall('NL.\d{6,7}|NL..\d{6,7}|NL.:.\d{6,7}',text)
    if nl == []:
        nl = re.findall('NL.no.\d{6,7}', text, re.I)
```

Fonte: O autor (2020)

Regra (5) - Extração dos valores dos serviços: Parte do código utilizado para extrair os valores dos serviços é mostrado na Figura 24 abaixo. O sistema identifica padrões de escrita dos valores dos serviços (ex: R$ 389.715,62), soma todas as ocorrências e armazena em uma variável que é utilizada pelo sistema para exportar seu conteúdo para uma base de dados ou como planilha no formato csv.

Figura 24 - Código fonte para extrair os valores dos serviços

```
vlrservicoobs = re.findall('\d{1,3}.\d{3}.\d{3},\d{2}|\d{1,3}.\d{3},\d{2}|\d{1,3},\d{2}', descricaoServico, re.I)
```

Fonte: O autor (2020)

Regra (6) - Extração dos serviços sem Isenção Total de ISS: Parte do código utilizado para extrair os serviços sem isenção total é mostrado na Figura 25 abaixo. O sistema identifica padrões de escrita dos serviços sem isenção total de ISS (ex: Condicionamento) e armazena em uma variável que é utilizada pelo sistema para exportar seu conteúdo para uma base de dados ou como planilha no formato csv.

Figura 25 - Código fonte para extrair os serviços sem isenção total de ISS

```
sem_isencao = re.findall('condicionamento', text, re.I)
sem_isencao += re.findall('desmobiliza...', text, re.I)
sem_isencao += re.findall('.assistencia.pre.operacional',text, re.I)
```

Fonte: O autor (2020)

5.1.3 Processo de Classificação de Textos

Após o processo da Extração de Informações, o sistema realiza a processo de Classificação de texto, onde são realizadas inferências entre as informações extraídas e também campos da nota fiscal.

Antes da classificação é realizado o pré-processamento para transformar informações não estruturadas extraídas, em formato de texto, para informações estruturadas em datas, números, valores. São usadas as próprias funções da linguagem *Python* durante este processo.

Após o pré-processamento é realizado o processo de classificação com a execução das 4 regras descritas na seção 4.4 deste trabalho. Essas regras foram criadas utilizando a abordagem baseada em conhecimento explícito, (seção 3.2.2).

A seguir são listadas as 4 regras de classificação das NFS-e:

Regra (1) - Competência de prestação do serviço x Competência da NFS-e: Esta regra de classificação utiliza o resultado da regra de extração da competência do serviço como entrada, e a compara com a competência da nota fiscal. A Figura 26 abaixo mostra parte do código utilizado neste processo.

Figura 26 - Código fonte para classificar Competência do Serviço x NFS-e

```
if comp_servico == []:
    classificacao += 0
else:
    if (row['mes_comp'], row['ano_comp']) == selecionaCompetencia(comp_servico[len(comp_servico) - 1]):
        classificacao += 0
    else:
        classificacao += 1
```

Fonte: O autor (2020)

Regra (2) - Valor dos serviços x Base de Cálculo para notas sem isenção total: Esta regra de classificação utiliza os campos valor_total e base_calculo da própria NFS-e no processamento. A Figura 27 abaixo mostra parte do código utilizado neste processo.

Figura 27 - Código fonte para classificar Valor dos serviços x Base de Cálculo

```
if MoedaToFloat(row['valor_total']) == MoedaToFloat(row['base_calculo']):
    classificacao += 0
else:
    if row['aliquota'] != 0:
        classificacao += 1
```

Fonte: O autor (2020)

Regra (3) - Serviços sem isenção total de ISS: Esta regra de classificação utiliza o resultado da extração dos serviços sem isenção total no processamento. A Figura 28 mostra parte do código utilizado neste processo.

Figura 28 - Código fonte para classificar Serviços sem isenção de ISS

```
if sem_isencao != []:
    classificacao += 1
```

Fonte: O autor (2020)

Regra (4) - Notas emitidas com isenção fora do período: Esta regra de classificação utiliza os campos da própria NFS-e e a data final informada manualmente no processamento. A Figura 29 mostra parte do código utilizado neste processo.

Figura 29 - Código fonte para classificar as NFS-e emitidas com isenção fora do período

```
datafinal = '15/02/2015'
if StringToDate(row['data_nota']) >= StringToDate(datafinal):
    if int(row['aliquota']) == 0:
        classificacao += 1
```

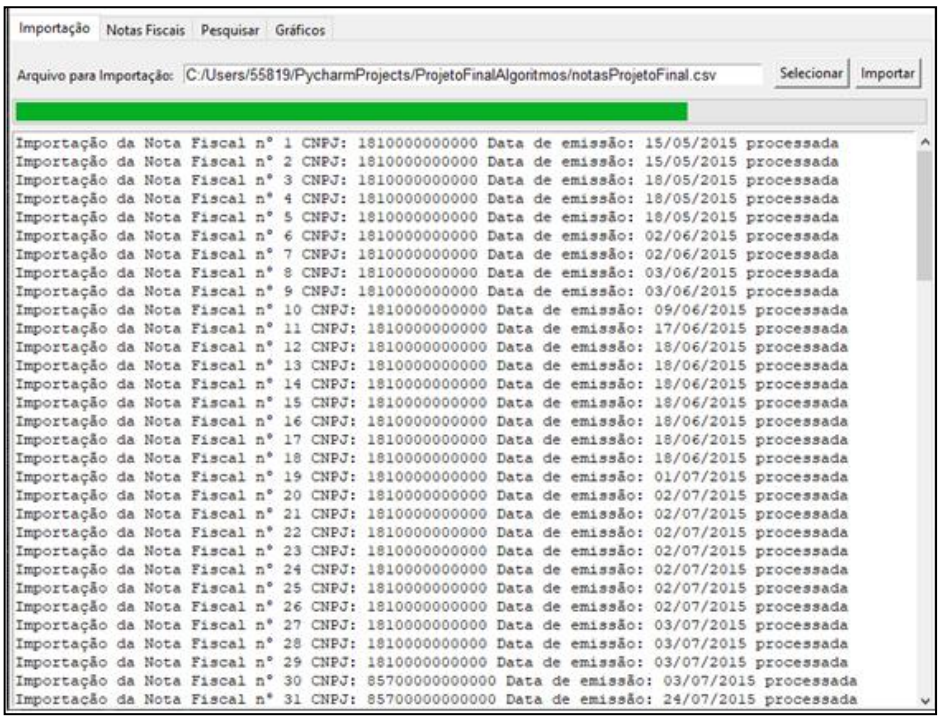
Fonte: O autor (2020)

5.1.4 Aplicação Audita-NFSe

A aplicação desenvolvida como protótipo deste trabalho é composta por 4 telas principais distribuídas da seguinte forma: Importação, Notas Fiscais, Pesquisar e Gráficos.

(1) Importação: Nesta área, é realizada a importação, selecionando o arquivo .csv com as notas fiscais (Figura 30). Na importação, são realizados os processos de Extração de Informação e Classificação de texto descritos neste capítulo. Após o processamento, as notas fiscais são armazenadas em um banco de dados, e podem ser acessadas na área de Notas Fiscais do sistema Audita-NFSe.

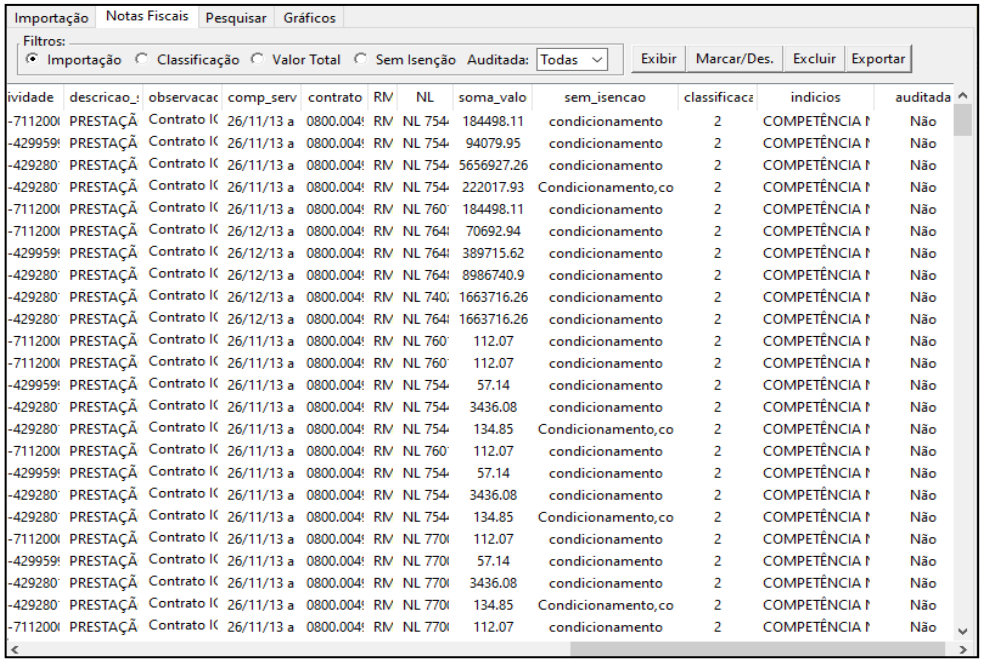
Figura 30 - Tela de importação das NFS-e do sistema Audita-NFSe



Fonte: O autor (2020)

(2) Notas Fiscais: Nesta área é possível visualizar todas as notas importadas para o sistema, marcar as notas que já foram auditadas, excluir notas e exportar notas selecionadas em formato de arquivo .csv (Figura 31). Essa tela mostra a classificação das NFS-e quanto aos indícios de fraude identificados (coluna classificação). Além disso, a tela também informa que notas já foram auditadas e a descrição das possíveis fraudes encontradas (coluna indícios), facilitando o controle do trabalho dos auditores.

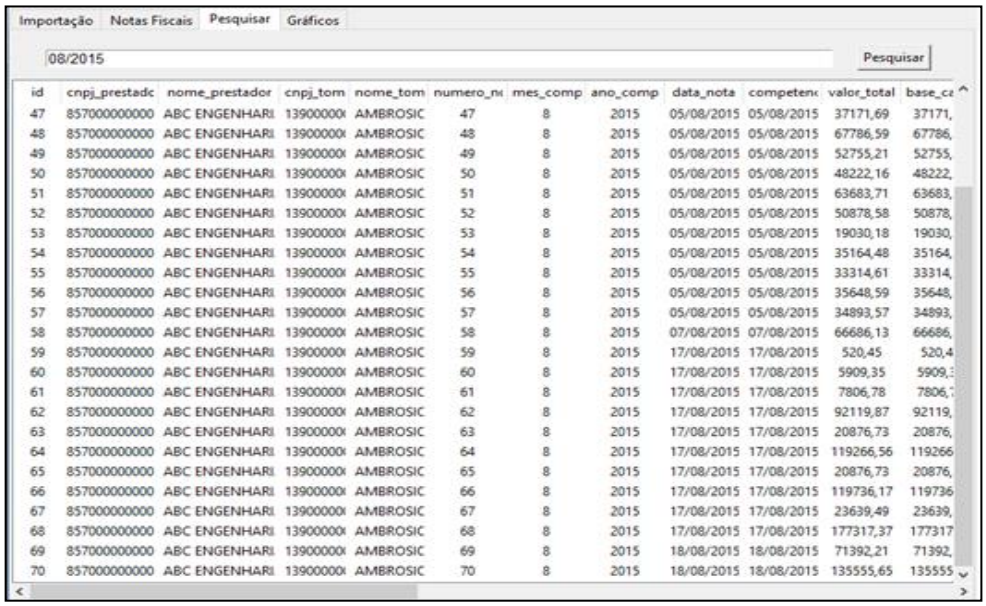
Figura 31 - Tela e exibição das NFS-e importadas



Fonte: O autor (2020)

(3) Pesquisar: Nesta tela é possível realizar uma pesquisa mais detalhada em toda base de dados de notas fiscais já importadas para o sistema (Figura 32). Esta busca facilita a localização de alguma informação específica, como exemplo, um número de contrato.

Figura 32 - Tela de pesquisa do sistema Audita-NFSe

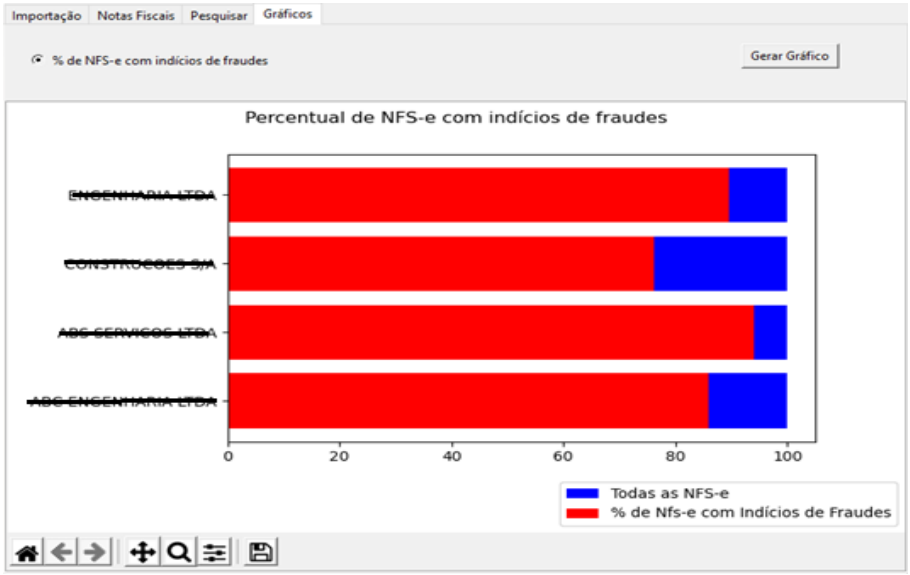


id	cnpj_prestadc	nome_prestador	cnpj_tom	nome_tom	numero_ni	mes_comp	ano_comp	data_nota	competenc	valor_total	base_ca
47	857000000000	ABC ENGENHARI	139000000	AMBROSIC	47	8	2015	05/08/2015	05/08/2015	37171,69	37171,
48	857000000000	ABC ENGENHARI	139000000	AMBROSIC	48	8	2015	05/08/2015	05/08/2015	67786,59	67786,
49	857000000000	ABC ENGENHARI	139000000	AMBROSIC	49	8	2015	05/08/2015	05/08/2015	52755,21	52755,
50	857000000000	ABC ENGENHARI	139000000	AMBROSIC	50	8	2015	05/08/2015	05/08/2015	48222,16	48222,
51	857000000000	ABC ENGENHARI	139000000	AMBROSIC	51	8	2015	05/08/2015	05/08/2015	63683,71	63683,
52	857000000000	ABC ENGENHARI	139000000	AMBROSIC	52	8	2015	05/08/2015	05/08/2015	50878,58	50878,
53	857000000000	ABC ENGENHARI	139000000	AMBROSIC	53	8	2015	05/08/2015	05/08/2015	19030,18	19030,
54	857000000000	ABC ENGENHARI	139000000	AMBROSIC	54	8	2015	05/08/2015	05/08/2015	35164,48	35164,
55	857000000000	ABC ENGENHARI	139000000	AMBROSIC	55	8	2015	05/08/2015	05/08/2015	33314,61	33314,
56	857000000000	ABC ENGENHARI	139000000	AMBROSIC	56	8	2015	05/08/2015	05/08/2015	35648,59	35648,
57	857000000000	ABC ENGENHARI	139000000	AMBROSIC	57	8	2015	05/08/2015	05/08/2015	34893,57	34893,
58	857000000000	ABC ENGENHARI	139000000	AMBROSIC	58	8	2015	07/08/2015	07/08/2015	66686,13	66686,
59	857000000000	ABC ENGENHARI	139000000	AMBROSIC	59	8	2015	17/08/2015	17/08/2015	520,45	520,4
60	857000000000	ABC ENGENHARI	139000000	AMBROSIC	60	8	2015	17/08/2015	17/08/2015	5909,35	5909,
61	857000000000	ABC ENGENHARI	139000000	AMBROSIC	61	8	2015	17/08/2015	17/08/2015	7806,78	7806,
62	857000000000	ABC ENGENHARI	139000000	AMBROSIC	62	8	2015	17/08/2015	17/08/2015	92119,87	92119,
63	857000000000	ABC ENGENHARI	139000000	AMBROSIC	63	8	2015	17/08/2015	17/08/2015	20876,73	20876,
64	857000000000	ABC ENGENHARI	139000000	AMBROSIC	64	8	2015	17/08/2015	17/08/2015	119266,56	119266
65	857000000000	ABC ENGENHARI	139000000	AMBROSIC	65	8	2015	17/08/2015	17/08/2015	20876,73	20876,
66	857000000000	ABC ENGENHARI	139000000	AMBROSIC	66	8	2015	17/08/2015	17/08/2015	119736,17	119736
67	857000000000	ABC ENGENHARI	139000000	AMBROSIC	67	8	2015	17/08/2015	17/08/2015	23639,49	23639,
68	857000000000	ABC ENGENHARI	139000000	AMBROSIC	68	8	2015	17/08/2015	17/08/2015	177317,37	177317
69	857000000000	ABC ENGENHARI	139000000	AMBROSIC	69	8	2015	18/08/2015	18/08/2015	71392,21	71392,
70	857000000000	ABC ENGENHARI	139000000	AMBROSIC	70	8	2015	18/08/2015	18/08/2015	135555,65	135555

Fonte: O autor (2020)

(4) Gráficos: Nesta tela é exibido um gráfico que mostra a porcentagem de NFS-e que foram classificadas com a possibilidade de existir fraudes (Figura 33). Esse gráfico é de grande valia para os auditores, pois eles conseguem identificar as empresas que mais comentem fraudes em suas NFS-e, mostrando a porcentagem de notas com indícios de fraudes.

Figura 33 - Tela de exibição de gráficos do sistema Audita-NFSe



Fonte: O autor (2020)

5.2 TESTES E RESULTADOS OBTIDOS

O protótipo foi testado utilizando-se um corpus de NFS-e que dispõe de 300 novas notas dos anos de 2014 e 2015, das mesmas empresas selecionadas entre as 3080 utilizadas na etapa de aquisição de conhecimento. Foram utilizadas as métricas de *Precisão*, *Cobertura* e *F-measure* para avaliar os resultados obtidos com o processo de EI dos campos de “observação” e “discriminação dos serviços”. O processo de classificação foi avaliado através da Matriz de Confusão e da métrica de Acurácia.

5.2.1 Resultados do processo de Extração de Informações

Abaixo são apresentados os resultados de *Cobertura*, *Precisão* e *F-measure* para o processo de Extração de Informação visto na seção 5.1.2 deste capítulo. Os valores são calculados seguindo os conceitos apresentados na seção 2.2.5 deste documento.

Regra (1) - Extração do período de competência da prestação do serviço (Quadro 03): Todas as 300 notas processadas possuem o período do serviço. Contudo, apenas 193 notas tiveram esse dado extraído corretamente. O sistema não conseguiu extrair os seguintes formatos de texto: “26/12/2014A 25/01/2015”, “01 a 25/09/2015”, “26/08 a 31/08/2015” e “26.06.2014 À 25.07.2014”. Este problema ocorreu porque esses formatos não estavam previstos na regra de extração. Como o campo de observação é escrito em texto livre, seu conteúdo pode sofrer alteração no formato, e também pode conter erros, como no caso de “26/12/2014A 25/01/2015”, onde a letra “A” está colada à data. Esses problemas podem ser corrigidos através de uma etapa de pré-processamento mais sofisticada.

Quadro 3 - Resultados da extração do período de competência de prestação do serviço

Nº extraídas corretamente	Nº a extrair	Nº extraídas	PRECISÃO	COBERTURA	F-MEASURE
193	300	300	64%	64%	64%

Fonte: O autor (2020)

Regra (2) - Extração do número do contrato (Quadro 04): Para esta regra de extração, das 300 notas processas, 299 possuem o número do contrato. Contudo, apenas 239 ocorrências foram extraídas corretamente. O sistema não conseguiu extrair o seguinte formato de texto: “0800.0049742.09-2”. Este formato não estava previsto na regra de extração criada, podendo ser incluído na próxima versão do sistema.

Quadro 4 - Resultados da extração do número do contrato

Nº extraídas corretamente	Nº a extrair	Nº extraídas	PRECISÃO	COBERTURA	F-MEASURE
239	299	239	100%	80%	89%

Fonte: O autor (2020)

Regra (3) - Extração do número do Relatório de Medição (Quadro 05): Para esta regra de extração, das 300 notas processadas, 299 possuem o número do relatório de medição, e todas as ocorrências foram extraídos corretamente. Esta regra possui uma maior precisão devido ao formato da informação ser mais regular.

Quadro 5 - Resultado da extração do Relatório de Medição

Nº extraídas corretamente	Nº a extrair	Nº extraídas	PRECISÃO	COBERTURA	F-MEASURE
299	299	299	100%	100%	100%

Fonte: O autor (2020)

Regra (04) - Extração do número da Nota de Liquidação (Quadro 06): Para esta regra de extração, das 300 notas processadas, 300 possuem o número do relatório de medição e 297 foram extraídos corretamente. O sistema não conseguiu extrair o formato “NL: 09610414” com 8 caracteres numéricos, extraindo apenas o formato “NL: 0961041” com 7 caracteres numéricos. Este formato não estava previsto na regra de extração criada, podendo ser incluído na próxima versão do sistema.

Quadro 6 - Resultado da extração do número da nota de liquidação

Nº extraídas corretamente	Nº a extrair	Nº extraídas	PRECISÃO	COBERTURA	F-MEASURE
297	300	300	99%	99%	99%

Fonte: O autor (2020)

Regra (5) - Extração dos valores dos serviços (Quadro 07): Para estra regra de extração, das 300 notas processadas, 292 possuem os valores dos serviços, mas apenas 232 foram extraídos corretamente. O sistema não conseguiu diferenciar valores no formato de moeda e valores no formato de porcentagem (%), por exemplo: R$150,00 e 35,98%. Dessa forma, tivemos um número maior de informações extraídas do que a quantidade correta de informações a serem extraídas. Esse problema também pode ser contornado via pré-processamento.

Quadro 7 - Resultado da extração dos valores dos serviços

Nº extraídas corretamente	Nº a extrair	Nº extraídas	PRECISÃO	COBERTURA	F-MEASURE
232	292	296	78%	79%	79%

Fonte: O autor (2020)

Regra (6) - Extração dos serviços sem Isenção Total de ISS (Quadro 08): Para estra regra de extração, das 300 notas processadas, 97 possuem serviços sem isenção total de ISS, mas apenas 94 informações foram extraídas corretamente. O sistema não conseguiu identificar corretamente o formato de texto “Assistência Pré Operacional” por motivo ainda não identificado.

Quadro 8 - Resultado da extração dos serviços sem isenção total de ISS

Nº extraídas corretamente	Nº a extrair	Nº extraídas	PRECISÃO	COBERTURA	F-MEASURE
94	97	94	100%	97%	98%

Fonte: O autor (2020)

No Quadro 09 a seguir é exibida a média dos cálculos de *Precisão*, *Cobertura* e *F-measure* dos campos extraídos.

Quadro 9 - Média do resultado das 6 regras de EI

Regra de extração	Precisão	Cobertura	F-measure
(1) Extração do período de competência da prestação do serviço	64%	64%	64%
(2) Extração do número do contrato	100%	80%	89%
(3) Extração do número do Relatório de Medição:	99%	100%	100%
(4) Extração do número da Nota de Liquidação	99%	99%	99%
(5) Extração dos valores dos serviços	78%	79%	79%
(6) Extração dos serviços sem Isenção Total de ISS	100%	97%	98%
MÉDIA	**90%**	**86%**	**88%**

Fonte: O autor (2020)

5.2.2 Resultados do processo de Classificação de Textos

Abaixo são apresentados os resultados do cálculo da acurácia de cada regra do processo de classificação apresentado na seção 5.1.3 deste capítulo, utilizando-se a Matriz de Confusão, de acordo com os conceitos apresentados na seção 2.2.5 deste trabalho.

Regra (1) - Competência de prestação do serviço x Competência da NFS-e (quadro 10): Para esta regra de classificação, apenas 193 das 300 notas processadas foram classificadas corretamente. Um dos campos utilizado no processamento como input da regra é o campo comp_servico, resultado da regra 1 de extração da competência do serviço. Como 107 notas retornaram resultados errados, o sistema classificou erroneamente essas notas. O resultado obtido foi uma acurácia de 64%.

Quadro 10 - Competência de prestação do serviço x Competência da NFS-e

	Classificação Manual	
Classificação Automática	**Positivas**	**Negativas**
Positivas	193 (VP)	0 (FP)
Negativas	107 (FN)	0 (VN)

Fonte: O autor (2020)

Regra (2) - Valor dos serviços x Base de Cálculo para notas sem isenção total (Quadro 11): Nenhuma das 300 notas utilizadas no processamento desta regra de classificação apresentou divergência entre o valor dos serviços e a base de cálculo. Assim, neste caso o sistema obteve 100% de acurácia.

Quadro 11 - Valor dos serviços x Base de Cálculo para notas sem isenção total

	Classificação Manual	
Classificação Automática	**Positivas**	**Negativas**
Positivas	0 (VP)	0 (FP)
Negativas	0 (FN)	300 (VN)

Fonte: O autor (2020)

Regra (3) - Serviços sem isenção total de ISS (Quadro 12): O cálculo da acurácia desta regra de classificação foi feito com base em 300 notas. O campo sem_isencao, resultado da extração dos serviços sem isenção total de ISS, é utilizado como input para o processamento desta regra. O sistema acertou a classificação de 241 notas, obtendo assim uma acurácia de 80%.

Quadro 12 - Serviços sem isenção total de ISS

	Classificação Manual	
Classificação Automática	**Positivas**	**Negativas**
Positivas	38 (VP)	59 (FP)
Negativas	0 (FN)	203 (VN)

Fonte: O autor (2020)

Regra (4) - Notas emitidas com isenção fora do período (Quadro 13): Para esta regra de classificação, das 300 notas processadas, todas foram classificadas corretamente. Assim, o sistema obteve uma acurácia de 100%.

Quadro 13 - Notas emitidas com isenção fora do período

	Classificação Manual	
Classificação Automática	**Positivas**	**Negativas**
Positivas	31 (VP)	0 (FP)
Negativas	0 (FN)	269 (VN)

Fonte: O autor (2020)

No Quadro 14 a seguir é exibida a média do cálculo das 4 regras de Classificação de texto.

Quadro 14 - Média do resultado das 4 regras de Classificação de Texto

Regra de Classificação	**Acurácia**
Regra (1) - Competência de prestação do serviço x Competência da NFS-e	64%
Regra (2) - Valor dos serviços x Base de Cálculo para notas sem isenção total	100%
Regra (3) - Serviços sem isenção total de ISS	80%
Regra (4) - Notas emitidas com isenção fora do período	100%
MÉDIA	86%

Fonte: O autor (2020)

5.2.3 Avaliação Geral dos Resultados Obtidos

Diante dos resultados obtidos nos processos de Extração de Informação, foi identificado que quanto mais complexos os dados a serem processados menor é precisão do protótipo desenvolvido. Isto é resultado da grande diversidade dos dados que são escritos em texto livre.

Outro dado observado, nos resultados dos processos de Classificação de Textos, é que duas das quatro regras existentes, dependem de informações extraídas corretamente. Dessa forma, erros no processo de extração, afetam o desempenho do módulo de classificação.

Apesar da baixa complexidade das regras de classificação, estes resultados são de grande importância para o trabalho proposto, pois, com a execução dessas regras é possível identificar possíveis indícios de fraudes nas NFS-e.

5.3 CONSIDERAÇÕES FINAIS

O objetivo deste capítulo foi apresentar detalhes dos principais processos do protótipo desenvolvido, bem como a apresentação dos resultados obtidos através da utilização de métricas que calculam a eficácia da aplicação para o que foi proposta.

Neste capítulo também foram apresentados detalhes da tecnologia usada para desenvolver o protótipo, como a linguagem de programação, bibliotecas extras que foram utilizadas e também parte do código fonte desenvolvido nos principais processos do protótipo.

6 CONCLUSÃO

Neste capítulo, temos a conclusão do trabalho realizado. Aqui listamos as principais contribuições, bem como sugestões de trabalhos futuros. Como mostrado anteriormente, o objetivo principal deste trabalho foi desenvolver um sistema que auxilie os Auditores Fiscais Tributários durante o processo de análise e identificação de possíveis fraudes em Notas Fiscais de Serviços Eletrônicas.

6.1 PRINCIPAIS CONTRIBUIÇÕES

A principal contribuição deste trabalho foi ter como foco a utilização da tecnologia para melhoria na prestação do serviço público, com o desenvolvimento de uma aplicação que auxilia parte do trabalho de análise de notas fiscais do setor de fiscalização tributária. Também deve ser levado em consideração que esta pesquisa foi desenvolvida em um ambiente real, utilizando dados reais do Município do Ipojuca/PE.

De acordo com revisões bibliográficas realizadas, apesar de existirem muitos trabalhos na área de Extração de Informações e Classificação de texto, não foram encontradas iniciativas semelhantes à deste trabalho. Em outras palavras, não foram encontrados trabalhos que apresentem propostas para análise de Notas Fiscais de Serviços Eletrônicas utilizando as mesmas técnicas abordadas neste trabalho.

As técnicas utilizadas neste projeto para a análise do ISS, que é um imposto municipal, podem ser adaptadas para análise e identificação de possíveis fraudes também em outros impostos em uma NFS-e, como o INSS (Instituto Nacional do Seguro Social), PIS/PASEP (Programa de Integração Social e Programa de Formação do Patrimônio do Servidor Público), COFINS (Contribuição para o Financiamento da Seguridade Social), CSLL (Contribuição Social sobre o Lucro Líquido), IRPJ (Imposto de Renda Pessoa Jurídica), que são impostos federais, e o ICMS (Imposto sobre Operações relativas à Circulação de Mercadorias e Prestação de Serviços de Transporte Interestadual e Intermunicipal e de Comunicação), que é um imposto estadual.

Com este trabalho, foi possível demostrar como a utilização de técnicas da Inteligência Artificial tem um forte potencial para agilizar parte das atividades dos setores de fiscalização tributária de governos municipais, estaduais e federais, pois

torna possível o processamento de uma grande quantidade de dados, trazendo benefícios em relação aos procedimentos manuais de análise de NFS-e realizados por auditores fiscais.

6.2 TRABALHOS FUTUROS

Nesta seção, veremos algumas sugestões para trabalhos futuros.

Regras de EI: Para este trabalho, foi identificado que o módulo de EI não conseguiu extrair corretamente todas as informações disponíveis dos campos de observação e descrição do serviço da NFS-e. Como sugestão, é interessante incluir mais opções de padrões de escrita utilizando tanto as NFS-e do conjunto de testes deste projeto quanto novos conjuntos de NFS-que ainda não foram analisadas, a fim de identificar novos padrões de escrita.

Aprendizagem de Máquina: Para este trabalho, foram utilizadas regras baseadas em conhecimento explícito tanto no processo de Extração de Informações como no processo de Classificação de Texto. Como sugestão, pode ser interessante a utilização de técnicas de aprendizagem de máquina nestes processos, utilizando para treinamento, as NFS-e com 100% de acerto processadas neste sistema.

Novos setores de fiscalização: Para este trabalho foram utilizadas notas fiscais de serviços eletrônicas de empresas do setor de construção civil. Como sugestão, é interessante utilizar as técnicas empregadas neste trabalho e adaptá-las em outros setores como hotelaria e serviços portuários.

Análise de outros impostos: Para este trabalho foram utilizadas técnicas para análise do ISS (Imposto Sobre Serviços), Como sugestão, é interessante utilizar técnicas empregadas neste trabalho e adaptá-las para análise de impostos federais e estaduais.

REFERÊNCIAS

ABEL, M.; FIORINI, S. **Uma revisão da Engenharia do Conhecimento: Evolução, Paradigmas e Aplicações**. International Journal of Knowledge Engineering and Management (IJKEM). 2. 1-35. 2013. Disponível em: <https://www.researchgate.net/publication/264156426_Uma_revisao_da_Engenharia_do_Conhecimento_Evolucao_Paradigmas_e_Aplicacoes>. Acesso em: 10 Jan. 2020.

ABRASF. **NFS-e Modelo Conceitual. Versão 1.0**. Associação Brasileira das Secretarias de Finanças das Capiais, 2008. Disponível em: <http://www.abrasf.org.br/arquivos/publico/NFS-e/Versao_1.00/NFSe_ModeloConceitual_2008dez29.pdf>. Acesso em: 20 Mar. 2020

AGGARWAL, C.C. **Data Mining: The Textbook**. Springer International Publishing, Switzerland, 2015.

ÁLVAREZ, A.C. **Extração de informação de Artigos Científicos uma abordagem baseada em indução de regras de etiquetagem**. Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, São Carlos, 2007.

ANDRADE, P.H.M.A. **Aplicação de Técnicas de Mineração de Textos para Classificação de Documentos: um Estudo da Automatização da Triagem de Denúncias na CGU**. Dissertação de Mestrado Profissional. UnB, Brasília, 2015. Disponível em: <https://repositorio.unb.br/bitstream/10482/21004/1/2015_Patr%C3%ADciaHelenaMaiaAlvesdeAndrade.pdf>. Acesso em: 15 Jan. 2020.

ARANHA, C.; PASSOS, E. **A Tecnologia de Mineração de Textos**. RESI - Revista Eletrônica de Sistemas de Informação. v. 5. n. 2. Rio de Janeiro, 2006. Disponível em: <http://www.periodicosibepes.org.br/index.php/reinfo/article/view/171/66>. Acesso em: 06 Dez. 2019.

ARAUJO, L.S. **Nota Fiscal Eletrônica**. Monografia para obtenção do grau de especialista em Gestão e Planejamento Tributário. Universidade Candido Mendes. Niterói, 2019.

BAEZA-YATES, R.; RIBEIRO-NETO, B. **Modern Information Retrieval**. ACM Press, New York, 1999.

BARION, E.C.N.; LAGO, D. **Mineração de textos**. Revista de Ciências Exatas e Tecnologia. Vol.III, nº3. Sao Paulo: Anhanguera Educacional S.A, 2008.

BARROS, F. **IN1152 - Recuperação Inteligente de Informação**. Slides das aulas do curso de mestrado. CIN/UFPE. Recife, 2017. Disponível em: <https://www.cin.ufpe.br/~in1152/2017/>. Acesso em: 03 Dez. 2020.

BERTOZZO, R.J. **Aplicação de Machine Learning em Dataset de Consultas Médicas do SUS**. Dissertação de Graduação em Sistemas de Informação, UFSC. Florianópolis, 2019

BRASIL. **Casa Civil. Lei nº 8846. Dispõe sobre a emissão de documentos fiscais e o arbitramento da receita mínima para efeitos tributários, e dá outras providências**. Brasilia, 1994. Disponível em: <http://www.planalto.gov.br/ccivil_03/leis/L8846.htm>. Acesso em: 02 Fev. 2021.

BRITO, E. M. **Mineração de Textos: Detecção automática de sentimentos em comentários nas mídias sociais**. Dissertação (Mestrado Profissional em Sistemas da Informação e Gestão do Conhecimento) – Universidade Fundação Mineira de Educação e Cultura. Belo Horizonte, 2016.

CORRÊA, G.M.; MARCACINI, R.M.; REZENDE, S.O. **Uso da mineração de textos na análise exploratória de artigos científicos**. Relatórios Técnicos do Instituto de Ciências Matemáticas e Computação, São Carlos, 2012. Disponível em: <http://repositorio.icmc.usp.br/handle/RIICMC/6631>. Acesso em: 19 Jan. 2020

CÔRTES, S.C.; PORCARO, R.M.;LIFSCHITZ, S. **Mineração de Dados - Funcionalidades, Técnicas e Abordagens**. PUC-RioInf.MCC10/02, Rio de Janeiro, 2002. Disponível em: <ftp://obaluae.inf.puc-rio.br/pub/docs/techreports/02_10_cortes.pdf>. Acesso em: 05 Fev. 2020.

DEAN, J. **Big Data, Data Mining, and Machine Learning: Value Creation for Business Leaders and Practitioners**. New Jersey: Wiley, 2014.

DOMINGUES, M.L.C.S. **Mineração de Dados Utilizando Aprendizado Não-Supervisionado: um estudo de caso para bancos de dados de saúde**. Porto Alegre: PPGC da UFRGS, 2003. Disponível em: <https://www.lume.ufrgs.br/bitstream/handle/10183/2702/000375416.pdf?sequence=1>. Acesso em: 09 Mar.2020.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. **From Data Mining to Knowledge Discovery in Databases**. AI Magazine Volume 17 Number 3, 1996.

FELDMAN, R. SANGER, J. **Mining Handbook - Advanced Approaches in Analyzing Unstructured Data**. Cambridge University Press. New York, 2006.

FELFERING, A. KOSTYANTYN, S. **Debugging user interface descriptions of knowledge-based recommender applications**. Proceding of the 11th International Conference on Intelligent user interfaces, pp. 234-241, Sydney, Jan.-Feb, 2006.

FINIZOLA, A.B.S. **Dogfooding Analysis System: um sistema de análise de feedbacks de dogfooding para auxiliar as atividades de Testes Exploratórios**. Dissertação de Mestrado em Ciencia da Computação, UFPE. Recife, 2019

FRANK, E.; HALL, M.A.; WITTEN, I.H. **The WEKA workbench - Online Appendix for "Data Mining: Pratical Machine Learning Tools and Techniques"**. Forth Edition, 2016. Disponível em: <https://www.cs.waikato.ac.nz/ml/weka/Witten_et_al_2016_appendix.pdf>, Acesso em: 11 Mai, 2020.

GOLÇALVES, E.C. **Mineração de Texto: Conceitos e Aplicações Práticas**. SQL Magazine, v.105, p.31-44, 2012. Disponível em: <https://www.researchgate.net/publication/317912973>, Acesso em: 06 Mai. 2020.

GONÇALVES, T.; SILVA, C.; QUARESMA, P.; VIEIRA, R**. Analysing Part-of-Speech for Portuguese Text Classification. In: Gelbukh A. (eds) Computational Linguistics and Intelligent Text Processing. CICLing 2006**. Lecture Notes in Computer Science, vol 3878. Springer, Berlin, Heidelberg.

GRUS, J. **Data Science do Zero: Primeiras Regras com Python**. Rio de Janeiro: Alta Books, 2016.

HAN, J. KAMBER, M. PEI, J. **Data Mining: Concepts and Techniques, 3rd ed**. Waltham: Elsevier, 2011.

HARRINGTON, P. **Machine Learning in Action. Shelter Island**: Manning Publication Co, 2012.

JARGAS, A.M. **Expressões Regulares: Uma abordagem divertida**. 4ª Edição. São Paulo: Novatec, 2012.

KUSHMERICK, N.; THOMAS, B. **Adaptive information extraction: Core technologies for information agents. Lecture Notes in Computer Science**. Springer, Berlin, 2003.

LIMA, K.S.;VIANA, J.M. **Crime de Sonegação Fiscal: Lei 8.137/90**. Âmbito Jurídico, 2020. Disponível em: <https://ambitojuridico.com.br/cadernos/direito-tributario/crime-de-sonegacao -fiscal-lei-8-137-90/>, Acesso em: 28 Abr. 2020.

LOPES, M.C.S. **Mineração de Dados Textuais Utilizando Técnicas de Clustering para o Idioma Português**. Tese de doutorado em Ciências em Engenharia Civil, UFRJ. Rio de Janeiro, 2004.

MADEIRA, R. O. C. **Aplicação de técnicas de mineração de texto na detecção de discrepâncias em documentos fiscais**. Dissertação de Mestrado em Modelagem Matemática da Informação (FGV), Escola de Matemática Aplicada. Rio de Janeiro, 2015.

MAIMON, O.; ROKACH, L. **Chapter 1 - Introduction to Knowledge Discovery in Databases**. 10.1007/0-387-25465-X_1. 2005. Disponível em: <https://www.researchgate.net/ publication/225835494_Chapter_1_-_Introduction_to_Knowledge_Discovery_in_Databases>, Acesso em: 05 Fev. 2020.

MANNING, C. D.; RAGHAVAN, P.; SCHUTZE, H. **An Introduction to Information Retrieval**. Cambridge University Press, 2008.

MARTINS, Claudia Aparecida et al. **Uma Experiência em Mineração de Textos Utilizando Clustering Probabilístico Clustering Hierárquico**. Relatórios Técnicos do Instituto de Ciências Matemáticas e de Computação, São Carlos, 2003. Disponível em: <http://conteudo.icmc.usp.br/CMS/Arquivos/arquivos_enviados/BIBLIOTECA_113_RT_205.pdf>, Acesso em: 05 Mar. 2020.

MATOS, E.J.V. **DRT TC Classifier: Classificação de casos de teste para automação de testes de dispositivos móveis**. Programa de Pós-graduação em Ciência da Computação do Centro de Informática da UFPE. Recife, 2020.

MELO, T.F.O.; OLIVEIRA, M.S. **Controle Fiscal: Análise da Sonegação na Sociedade Brasileira**. Artigo Apresentado no XVII - Simpósio de Excelência em Gestão e Tecnologia. Rio de Janeiro: AEDB, 2017. Disponível em: <https://www.aedb.br/seget/arquivos/artigos17/ 13325117.pdf>, Acesso em: 23 Abr. 2020.

MICROSOFT. **O que é o Azure Machine Learning?** Documentação do Azure Machine Learning. Disponível em:<https://docs.microsoft.com/pt-br/azure/machine-learning/overview-what-is-azure-ml>. Acesso em: 16 Dez. 2020.

MOHRI, M.; ROSTAMIZADEH, A.; TALWALKAR, A. **Foudation of Machine Learning**. Second Edition. Cambridge: MIT Press, 2018.

MORAIS, E. A. M.; AMBRÓSIO, A. P. L. **Mineração de Textos**. Instituto de Informática da Universidade Federal de Goiás. Goiás, 2007. Disponível em: <http://www.inf.ufg.br/sites/default/files/uploads/relatorios-tecnicos/RT-INF_005-07.pdf> Acesso em: 08 Mar. 2020.

MUNHOZ, J.P. **Fiscalização do ISS na prática passo a passo**. Curitiba/PR. Disponíve em:<https://www.unipublicabrasil.com.br/uploads/materiais/2f548bea3506acde03d039a88b48caeb17032017133040.pdf>, Acesso em: 27 Abr. 2020.

NASCIMENTO, J.C. **Avaliação de Desempenho de Algoritmos de Classificação em Mineração de Opinião em Textos em Português**. Monografia para obtenção do grau de bacharel em Sistemas de Informação da Universidade Federal do Acre. Rio Branco, 2019

PELLUCCI, P.R.S. et al. **Utilização de Técnicas de Aprendizagem de Máquina no Reconhecimento de Entidades Nomeadas no Português**. e-xacta, UniBH, Belo Horizonte, v. 4, n. 1, p. 73-81. 2011.

PEREIRA, C.R.M.P. **Engenharia do Conhecimento**. Editora Unicentro. Paraná,2015. Disponível em: <http://repositorio.unicentro.br:8080/jspui/bitstream/123456789/935/5/Engenharia%20do%20conhecimento.pdf>. Acesso em: 24 Jan. 2020

PETRI, S.M. **Funcionamento do Processo de Fiscalização do ISS do Município de Palhoça**. Trabalho de Conclusão de Curso de Ciências Contábeis da UFSC. Florianópolis, 2016 apud TAUIL, Roberto Adolfo. O PLANEJAMENTO DA FISCALIZAÇÃO TRIBUTÁRIA. 2003. Disponível em: <http://consultormunicipal.adv.br/artigo/fiscalizacao-municipal/oplanejamento-da-fiscalizacao-tributaria/>. Acesso em: 05 dez. 2020.

PEZZINI, Anderson. **Mineração de Textos Conceito, Processo e Aplicações**. 2016. Disponível em <http://www.revistas.udesc.br/index.php/reavi/article/view/6750/6415>. Acesso em 14 Fev. 2020.

PREFEITURA DO IPOJUCA. **Portal do Contribuinte**. Secretaria de Finanças. Disponível em: <: https://www.ipojuca.pe.gov.br/contribuinte/>. Acesso em: 24 Ago. 2020.

PREFEITURA DO RECIFE. **NFSE - Nota Fiscal de Serviços Eletrônica**. Disponível em: <https://nfse.recife.pe.gov.br/>. Acesso em: 27 Abr. 2020.

PREFEITURA DO RIO DE JANEIRO. **Nota Carioca**. Disponível em: <https://notacarioca.rio.gov.br/>. Acesso em: 27 Abr. 2020.

PREFEITURA DE SÃO PAULO. **NF-e - Nota do Milhão**. Secretaria Municipal da Fazenda. Disponível em: <http://notadomilhao.prefeitura.sp.gov.br/>. Acesso em: 24 Abr. 2020.

RIBEIRO, L.P. **Infomovie: Sistema de Extração de Informação com interface para linguagem natural**. Projeto de Graduação em Engenharia de Computação da UFRJ. Rio de Janeiro, 2013.

RODRIGUES, J.P. **Sistemas Inteligentes híbridos para classificação de texto**. Dissertação de Mestrado em Ciência da Computação, UFPE. Recife, 2009.

ROSA, F.S. **Aprendizagem de máquina para apoio à tomada de decisão em vendas do varejo utilizando registros de vendas**. Monografia do curso de Engenharia de Controle e Automação. UFSC, Florianópolis, 2016.

RUSSELL,S. NORVING, P. **Inteligência Artificial**. Tradução da 3ª Edição. Rio de Janeiro: Elsevier, 2013.

SANTOS, C. **Auditoria Fiscal e Tributária**. 3ª Edição. Sao Paulo: IOB|SAGE, 2015.

SANTOS, H.A. **Utilização de um Sistema Especialista para diagnóstico de patologias ortopédicas dos membros inferiores**. Trabalho de Conclusão do curso de Sistemas de Informação do Centro Universitário Luterano de Palmas – CEULP/ULBRA. Palmas, 2011.

SANTOS, M. **Auditoria Tributária**. São Paulo: Editora Senac, 2020.

SANTOS, R.E.S *et al*. **Técnicas de processamento de liguagem natural aplicadas ao processo de mineração de textos: Resultados preliminares de um mapeamento sistemático**. Revista de Sistemas e Computação, Salvador, v. 4, n. 2, p. 116-125, jul./dez. 2014.

SCARINCI, R.G. **SES - Sistema de Extração Semântica de Informações**. Dissertação de Mestrado em Ciência da Computação. Porto Alegre: UFRGS, 1997.

SEGARAN, T. **Programing Collective Inteligence**. Sebastopol: O'Reilly, 2007

SETTE, B.S.; MARTINS, C.A. **Pré-processamento textual para a extração de informação em bases de patentes**. Instituto de Computação - Universidade Federal de Mato Grosso (UFMT). Cuiabá, 2016.

SILVA, T.M.S. **Extração de informações para busca semântica na web baseada em ontologias**. Dissertação do curso de pós-graduação em engenharia elétrica, UFSC. Florianópolis, 2003.

SOARES, F. A. **Mineração de textos na coleta inteligente de dados na web**. Dissertação de Mestrado em Engenharia Elétrica, Departamento de Engenharia Elétrica do Centro Técnico Científico da PUC-Rio. Rio de Janeiro, 2008.

TAN, A-H. **Text mining the state of the art and the challenges**. In Proceddings of the PAKDD99 workshop on Knowledge Discovery from Advanced Databases. Beijing, 1999.

TICOM, A.A.M. **Aplicação das Técnicas de Mineração de Textos e Sistemas Especialistas na Liquidação de Processos Trabalhistas**. Dissertação (Mestrado em Engenharia Civil) - Programa de Pós-graduação em Engenharia, UFRJ, Rio de Janeiro, 2007.

THARWAT, A. **Classification assessment methods**. Applied Computing and Informatics, Vol. ahead-of-print No. ahead-of-print. https://doi.org/10.1016/j.aci.2018.08.003. Frankfurt, 2020.

WEKA. **O ambiente de trabalho para aprendizado de máquina**. Disponível em: <https://www.cs.waikato.ac.nz/ml/weka/>, Aceso em: 10 Mai. 2020.

XU, R.; WUNSCH, D. **Clustering**. Wiley-IEEE Press, IEEE Press Series on Computational Intelligence, 2008.

ZAMBENEDETTI, C. **Extração de Informação sobre Bases de Dados Textuais**. Dissertação (mestrado) - Programa de Pós-Graduação em Computação, Porto Alegre, 2002.