UNIVERSIDADE FEDERAL DE MINAS GERAIS
ESCOLA DE CIÊNCIA DA INFORMAÇÃO

AGRUPAMENTO AUTOMÁTICO DE NOTÍCIAS DE JORNAIS *ON-LINE* USANDO TÉCNICAS DE *MACHINE LEARNING* PARA *CLUSTERING* DE TEXTOS NO IDIOMA PORTUGUÊS

Lúcia Helena de Magalhães

Belo Horizonte
2020

LÚCIA HELENA DE MAGALHÃES

AGRUPAMENTO AUTOMÁTICO DE NOTÍCIAS DE JORNAIS ONLINE USANDO TÉCNICAS DE *MACHINE LEARNING* PARA *CLUSTERING* DE TEXTOS NO IDIOMA PORTUGUÊS

Tese apresentada ao Programa de Pós-Graduação em Gestão & Organização do Conhecimento da Escola de Ciência da Informação, Universidade Federal de Minas Gerais, como requisito para obtenção do grau de Doutor.

Linha de Pesquisa: Gestão e Tecnologia

Área de Concentração: Representação do Conhecimento

Orientador: Renato Rocha Souza

Belo Horizonte

2020

M188a Magalhães, Lúcia Helena

Agrupamento automático de notícias de jornais online usando técnicas de machine learning para clustering de textos no idioma português [recurso eletrônico] / Lúcia Helena Magalhães. – 2020.
1 recurso eletrônico (188f.: il., color): pdf.

Orientador: Renato Rocha Souza.
Tese (doutorado) – Universidade Federal de Minas Gerais, Escola de Ciência da Informação.
Referências: f. 173-188
Exigências do sistema: Adobe Acrobat Reader.

1. Ciência da informação – Teses. 2. Organização da informação - Teses 3. Aprendizado do computador - Teses 4. Processamento da linguagem natural (computação) – Teses I. Título. II. Souza, Renato Rocha. III. Universidade Federal de Minas Gerais, Escola de Ciência da Informação.

CDU: 025.4.03

Ficha catalográfica: Biblioteca Profª Etelvina Lima, Escola de Ciência da Informação da UFMG.

UNIVERSIDADE FEDERAL DE MINAS GERAIS

PROGRAMA DE PÓS-GRADUAÇÃO EM GESTÃO E ORGANIZAÇÃO DO CONHECIMENTO

FOLHA DE APROVAÇÃO

AGRUPAMENTO AUTOMÁTICO DE NOTÍCIAS DE JORNAIS ON-LINE USANDO TÉCNICAS DE MACHINE LEARNING PARA CLUSTERING DE TEXTOS NO IDIOMA PORTUGUÊS

LÚCIA HELENA DE MAGALHÃES

Tese submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em GESTÃO E ORGANIZAÇÃO DO CONHECIMENTO, como requisito para obtenção do grau de Doutor em GESTÃO E ORGANIZAÇÃO DO CONHECIMENTO, área de concentração CIÊNCIA DA INFORMAÇÃO, linha de pesquisa Gestão e Tecnologia.

Aprovada em 13 de fevereiro de 2020, pela banca constituída pelos membros:

Prof(a). Renato Rocha Souza (Orientador)
FGV/RJ [por videoconferência]

Prof(a). Emerson Augusto Priamo Moraes
IF/JF

Prof(a). Luiz Cláudio Gomes Maia
FUMEC

Prof(a). Maurício Barcellos Almeida
ECI/UFMG [por videoconferência]

Prof(a). Renata Maria Abrantes Baracho Porto
Escola de Arquitetura/UFMG [por videoconferência]

Belo Horizonte, 13 de fevereiro de 2020.

UNIVERSIDADE FEDERAL DE MINAS GERAIS

PROGRAMA DE PÓS-GRADUAÇÃO EM GESTÃO E ORGANIZAÇÃO DO CONHECIMENTO

ATA DA DEFESA DE TESE DA ALUNA
LÚCIA HELENA DE MAGALHÃES

Realizou-se, no dia 13 de fevereiro de 2020, às 14:00 horas, Sala 1000 - ECI/UFMG, da Universidade Federal de Minas Gerais, a defesa de tese, intitulada *AGRUPAMENTO AUTOMÁTICO DE NOTÍCIAS DE JORNAIS ON-LINE USANDO TÉCNICAS DE MACHINE LEARNING PARA CLUSTERING DE TEXTOS NO IDIOMA PORTUGUÊS*, apresentada por LÚCIA HELENA DE MAGALHÃES, número de registro 2016662233, graduada no curso de TECNOLOGIA EM PROCESSAMENTO DE DADOS, como requisito parcial para a obtenção do grau de Doutor em GESTÃO E ORGANIZAÇÃO DO CONHECIMENTO, à seguinte Comissão Examinadora: Prof(a). Renato Rocha Souza - FGV/RJ (Orientador) [por videoconferência], Prof(a). Emerson Augusto Priamo Moraes - IF/JF, Prof(a). Luiz Cláudio Gomes Maia - FUMEC, Prof(a). Maurício Barcellos Almeida - ECI/UFMG [por videoconferência], Prof(a). Renata Maria Abrantes Baracho Porto - Escola de Arquitetura/UFMG [por videoconferência].

A Comissão considerou a tese:

(X) Aprovada

( ) Reprovada

Finalizados os trabalhos, lavrei a presente ata que, lida e aprovada, vai assinada por mim e pelos membros da Comissão.
Belo Horizonte, 13 de fevereiro de 2020.

Prof(a). Renato Rocha Souza

Prof(a). Emerson Augusto Priamo Moraes

Prof(a). Luiz Cláudio Gomes Maia

Prof(a). Maurício Barcellos Almeida

Prof(a). Renata Maria Abrantes Baracho Porto

DEDICATÓRIA

Primeiramente, agradeço a Deus por mais essa realização.

Dedico a minha família e amigos, em especial ao Helder, companheiro de todas as horas e em todas as circunstâncias, pela compreensão de minhas ausências; e ao Davi Praxedes, pela motivação e pela experiência mais gratificante que algum dia senti.

Ao professor Renato por toda colaboração e paciência durante o desenvolvimento deste trabalho.

AGRADECIMENTO

Agradeço primeiramente a Deus, por sempre me iluminar e por me dar muita força e coragem para continuar mesmo passando por momentos tão difíceis.

Ao meu pai José Praxedes de Magalhães (em lembrança) pela educação e pelos valores recebidos. Pai, a saudade é grande, mas o amor é para sempre.

A minha mãe Hilda Moreira de Magalhães, um exemplo de mulher honesta e guerreira, pessoa que sempre deu o máximo de si em função de seus filhos.

Ao meu esposo Helder por compreender os meus momentos de silêncio e afastamento que se fizeram necessários para o desenvolvimento desta pesquisa.

Ao meu filho Davi, agradeço pela espontaneidade, carinho e amor incondicional que sempre me estimularam nos momentos difíceis.

Ao meu orientador, Professor Doutor Renato Rocha Souza, pelo apoio incondicional para a realização deste trabalho. As suas críticas construtivas, as discussões e reflexões foram fundamentais ao longo de todo o percurso. Serei eternamente grata por todas as orientações.

Aos professores que me deram a honra de tê-los como membros da banca examinadora.

Aos professores do PPGGOG pelos conhecimentos transmitidos, pela receptividade e por serem minha fonte de inspiração!

A todos os professores que tive a oportunidade de conviver durante toda a minha formação, pelos ensinamentos que promoveram o meu entusiasmo e o meu interesse em prosseguir com os estudos.

Ao IF Sudeste MG e aos meus colegas de trabalho que me concederam um período para eu dedicar exclusivamente ao doutorado, afastamento que foi fundamental para meu empenho.

Aos meus irmãos pelo amor e pela amizade, em especial a minha irmã Teresinha por me receber em sua residência no período que eu estava cursando as disciplinas e por todo o cuidado que teve comigo quando o cansaço me vencia.

A todos os meus familiares e amigos que muitas vezes foram privados do meu convívio para que eu pudesse estudar, mas sempre compreenderam e apoiaram as minhas iniciativas. Sou grata por vocês renovarem minhas energias para eu seguir em frente.

Um trabalho não se realiza sozinho.

Ele é fruto da união de esforços, incentivos, persistência e, acima de tudo, da amizade e do carinho daqueles que nos rodeiam.

A vida não dá e nem empresta, não se comove e nem se apieda. Tudo quanto ela faz é retribuir e transformar aquilo que nós lhe oferecemos.
Albert Einstein

RESUMO

Clusterização é uma técnica de organizar dados em grupos cujos membros apresentam alguma semelhança. Assim, a proposta desta pesquisa é utilizar as tecnologias de Mineração de Textos, Processamento de Linguagem Natural, *Machine Learning* e *Clustering*, para criar grupos de informes semelhantes a partir de uma amostra recuperada dos principais jornais *on-line*, uma vez que existem poucos estudos relacionados ao tema *clustering* de notícias publicadas no idioma português. Dessa forma, a lacuna de pesquisas nessa área acaba por reforçar e aprofundar a escassez de informação relacionada ao desenvolvimento de soluções automatizadas, capazes de recuperar e comparar as matérias em destaque na mídia, publicadas na língua brasileira, e agrupá-las por similaridade. Assim, este estudo tem como objetivo utilizar uma metodologia de aprendizado não supervisionado, que seja capaz de agrupar, automaticamente, notícias publicadas no idioma do Brasil, postadas na grande mídia. Além disso, busca identificar quais são os principais métodos utilizados no processo de *clustering* de textos; aplicar essas técnicas em uma coleção de notícias publicadas na língua portuguesa e verificar o desempenho dos algoritmos de clusterização ao serem alimentados por um corpus de textos; aplicar a metodologia em diferentes corpora e discutir o sucesso da técnica em cada caso; averiguar a possibilidade efetiva de clusterização dos documentos e analisar as dificuldades encontradas para diferentes amostras. Para tanto, são apresentados os conceitos e as áreas relacionadas com o tema, bem como a revisão bibliográfica dos trabalhos correlatos, a metodologia proposta e alguns experimentos que permitem desenvolver determinados argumentos e comprovar algumas hipóteses. Para as experimentações, primeiramente, coletaram-se as notícias e, em seguida, realizou-se o pré-processamento dos informes, etapa em que as *stop words* foram removidas e as técnicas de tokenização e *stemming* foram aplicadas. Assim, com o corpus preparado, extraíram-se as principais características dos textos e os documentos foram representados em um modelo de espaço vetorial. A semelhança entre as matérias foi encontrada através do cálculo da similaridade, imediatamente a técnica de *clustering* foi aplicada e consequentemente os grupos foram formados. Para melhor visualização, validação e interpretação dos resultados, apresentaram-se os *clusters* em dendogramas e em diagramas de dispersão. As conclusões principais desta pesquisa indicaram que a etapa de pré-processamento exige um esforço especial para garantir a qualidade dos dados. Assim como a complexidade da língua portuguesa, a necessidade de atualização da lista de *stop words,* a detecção de quais características são mais importantes e, em geral, a complexidade dos problemas relacionados à alta dimensionalidade dos dados foram evidenciados durante todo o processo deste estudo. As medidas de distância também desempenharam um papel importante na análise de *clustering*, porém não existe uma que melhor se adapte a todos os problemas de agrupamento. O algoritmo *k-means* obteve os melhores resultados para esse tipo de informação e o *Hierarchical Clustering* apresentou dificuldades para corpus grande, visto que documentos semelhantes foram alocados em grupos diferentes. Já o algoritmo *Affinity Propagation* apresentou divergência quanto ao número ideal de *clusters*, mas conseguiu bom desempenho ao agrupar por similaridade.

Palavras-chave: Agrupamento de notícias. Mineração de Textos. Aprendizado de Máquina. Processamento de Linguagem Natural.

ABSTRACT

Clustering is the technique of organization of data into groups whose members are somewhat similar. The purpose of this research is to use the techniques of Text Mining, Natural Language Processing, Machine Learning and Clustering, to create groups of similar reports from a sample retrieved from online newspapers, considering that there are few studies related to the clustering theme of news published in Portuguese. The lack of research in this area ends up reinforcing the scarcity of information, which interferes in the development of automated solutions capable of retrieving and comparing the articles featured in the media, published in Portuguese, and grouping them by similarity. Thus, this study aims to use an unsupervised learning methodology, which is capable of automatically grouping news published in the Brazilian Portuguese language, posted in the mainstream media. In addition, it also seeks to identify which are the main methods used in the text clustering process; apply these techniques to a collection of news published in the Portuguese language and verify the performance of the clustering algorithms when fed by a corpus of texts; apply the methodology in different corpora and discuss the success of the technique in each case; to investigate the effective possibility of document clustering and to analyze the difficulties encountered for different samples. For that, the concepts and areas related to the theme are presented, as well as the bibliographic review of related works, the proposed methodology and some experiments that allow developing certain arguments and proving some hypotheses. For the experiments, first, the news were collected and then, the pre-processing of the reports was carried out, a stage in which the stop words were removed and the tokenization and stemming techniques were applied. Thus, with the corpus prepared, the main characteristics of the texts were extracted and the documents were represented in a vector space model. The similarity between the materials was found by calculating the similarity, immediately the clustering technique was applied and consequently the groups were formed. For better visualization, validation and interpretation of results, clusters were presented in dendograms and in dispersion diagrams. The main conclusions of this research indicated that the pre-processing stage requires a special effort to guarantee the quality of the data. As well as the complexity of the Portuguese language, the need to update the list of stop words, the detection of which characteristics are most important and, in general, the complexity of the problems related to the high dimensionality of the data were evidenced throughout the process of this study. Distance measurements also played an important role in clustering analysis, but there is no one that best suits all clustering problems. The k-means algorithm obtained the best results for this type of information and Hierarchical Clustering presented difficulties for larger corpus, since similar documents were allocated to different groups. The Affinity Propagation algorithm, on the other hand, diverged as to the ideal number of clusters, but achieved good performance when grouping by similarity.

Keywords: Clustering of news. Text Mining. Machine Learning. Natural Language Processing.

LISTA DE FIGURAS

LISTAS DE TABELAS

LISTAS DE QUADROS

LISTAS DE ABREVIATURAS

ACP	Análise de Componentes Principais
AM	Aprendizado de Máquina
AP	*Affinity Propagation*
BoW	*Bag-of-Words*
DF	*Document Frequency*
EM	*Expectation Maximization*
FD	Frequência do Documento
Fabs	Frequência absoluta
Frel	Frequência relativa
IA	Inteligência Artificial
IDF	*Inverse Document Frequency*
kNN	*K-Nearest Neighbor*
LDA	*Latent Dirichlet Allocation*
LSI	*Latent Semantic Indexing*
MDS	*Multidimensional Scaling*
MIT	*Massachussets Institute of Technology*
NLP	*Natural Language Processing*
NLTK	*Natural Language Toolkit*
NMF	*Non-Negative Matrix Factorization*
OCR	Reconhecimento Ótico de Caracteres
PCA	*Principal Component Analysis*
PLN	Processamento de Linguagem Natural
PLSA	*Probabilistic Latent Semantic Analysis*
POS	*Part-Of-Speech Tagging*
RI	Recuperação da Informação
RSLP	Removedor de Sufixos da Língua Portuguesa
SSE	*Sum of Squared Error*
SVM	*Support Vector Machines*
TF	*Term Frequency*
TF-IDF	*Term-frequency-inverse document frequency*
VSM	*Vector Space Model*
WSS	*Within Sum of Square*

SUMARIO

1 INTRODUÇÃO

O amplo volume de dados que é gerado e publicado em inúmeras páginas da internet, nas quais se encontram informações das mais diversas áreas do conhecimento, facilitou a coleta, disseminação, produção e junção de uma grande quantidade de informação em um ambiente digital. Além disso, as pessoas passaram a produzir mais dados e, ao mesmo tempo, coletar mais informações para que elas ficassem disponíveis e de fácil acesso para auxiliar na tomada de decisão de forma rápida e segura. Porém, esse volume de dados, que cresce de forma exponencial, dificultou a localização e a leitura das informações em tempo hábil. Segundo Sampaio (2020), são produzidos mais de 2,5 quintilhões de bytes de dados diariamente, o que ocasiona um grande aumento de informações disponíveis.

Em consequência disso, surgiu uma grande demanda pela busca de informações relevantes neste emaranhado de dados, fazendo surgir importantes pesquisas relacionadas à Recuperação e Extração da Informação, áreas inspiradoras para a Ciência da Informação (CI). Segundo Borko (1968) a CI é responsável por investigar as propriedades e o comportamento da informação, as forças que governam seu fluxo e os meios de processá-la para otimizar sua acessibilidade e seu uso.

Mas, para tanto, ainda se fazem necessários mecanismos eficientes para extração de conhecimentos a partir de textos da internet, pois a busca por informação relevante e em tempo útil torna-se cada vez mais indispensável. De acordo com Souza *et al*. (2014), é primordial desenvolver metodologias e tecnologias associadas que possam enfrentar os muitos desafios que surgem ao lidar com grandes quantidades de dados textuais, “como nas bibliotecas e arquivos digitais, ou na *World Wide Web*, notadamente quando estes precisam ser regularmente organizados e pesquisados, visando à recuperação em tempo hábil de informações relevantes para algum objetivo específico.” (SOUZA *et al*., 2014, p.2).

Para Park e Paraubek (2010), os ambientes virtuais contêm fontes ricas de dados que podem ser utilizadas de forma eficiente. Esse uso eficaz é que fundamenta a importância da geração de sistemas capazes de analisar textos, ou seja, o desenvolvimento de aplicações computacionais eficientes e capacitadas para organizar e agrupar textos da internet de forma que o usuário possa manipular somente o que o interessa, descartando os grupos de textos irrelevantes e, desta forma, facilitando a leitura e a busca pelo conhecimento.

Desse modo, torna-se imprescindível o desenvolvimento de métodos automatizados de análise de texto para extrair conhecimento de documentos não estruturados da web e criar versões condensadas de informações que possam melhorar, por exemplo, a percepção do leitor na busca sobre um determinado assunto, sem a necessidade da leitura de centenas de documentos postados na web.

De acordo com Baeza-Yates e Ribeito-Neto (1999), o campo de desenvolvimento de ferramentas e métodos para organização de informações está em constante evolução, uma vez que grandes volumes de dados exigem processos tecnológicos de recuperação cada vez mais sofisticados.

Apoiando-se nesse contexto, esta pesquisa propõe o agrupamento automático de notícias coletadas de jornais *on-line*, utilizando algoritmos de *Machine Learning* e técnicas de Processamento de Linguagem Natural, com o objetivo de tornar o processo de descoberta de conhecimento mais eficiente. Neste estudo, será compilado, mapeado e analisado um conjunto de informes da atualidade. Para captura da coleção, será usado o *Media Frame*[1], uma ferramenta de código aberto, desenvolvida na Fundação Getúlio Vargas, que permite capturar um grande número de notícias das mídias *on-line*, porém, para utilizá-la, é necessário cadastro prévio.

Segundo Souza *et al*. (2014), o objetivo do projeto *Media Frame* é a estruturação de um processo contínuo de captação de uma infinidade de dados de cunho textual (jurídicos, legislativos, midiáticos, acadêmicos etc.) visando à realização de análises de natureza acadêmica, semântica, estatística e orgânica, permitindo a construção de modelos para percepção e inferência sobre a conjuntura brasileira e análises preditivas. Por meio do sistema é possível escolher as fontes de mídia de interesse, capturar coleções de notícias e a partir delas extrair características semânticas dos textos, fazer análise de assunto e conteúdo, construir estatísticas sobre frequências de palavras através de Processamento de Linguagem Natural e analisar os resultados. (SOUZA et, al. 2014).

Por conseguinte, optou-se por utilizar *clustering* nesta pesquisa pelo fato da classificação exigir uma base de dados rotulada previamente para treino. Mesmo possuindo uma base de dados rotulada, adquirida a partir de notícias anteriores, há poucas chances das mesmas classes, obtidas anteriormente por um classificador, em um processo de aprendizado supervisionado, continuarem sendo válidas para uma nova amostra, o que inviabiliza a classificação de novas notícias não rotuladas (NASSIF, 2013).

Desta forma, os métodos de aprendizado não supervisionado passam a ser mais interessantes neste estudo, pois através deles é possível descobrir padrões existentes em dados sem rótulos. Assim, com a utilização de algoritmos de clusterização, notícias previamente desconhecidas, mas com o mesmo padrão de conteúdo, são alocadas no mesmo grupo, facilitando, dessa forma, a análise dos textos. Nesse sentido, o leitor poderá analisar as notícias dos grupos que o interessa e descartar os grupos irrelevantes, evitando, assim, a leitura de uma grande quantidade de documentos.

[1] https://mediaframe.io

1.1 Apresentação e contextualização do tema

O crescimento de recursos e serviços digitais, tais como e-commerce, bibliotecas digitais, redes sociais, teve como consequência o aumento de informações disponibilizadas na web, que podem ser usadas para diversos fins. Esse fenômeno intensifica a conscientização para a exigência de técnicas efetivas que possam ajudar durante a busca e recuperação de textos divulgados na internet. (ABDULSAHIB; KAMARUDDIN, 2015).

Além do mais, a web é, atualmente, o maior, mais aberto e mais democrático sistema editorial do mundo. Sua crescente popularidade possibilitou a disponibilização de notícias, tais como as publicações postadas pelos jornais *on-line*. Com isso, novas oportunidades e desafios surgem na área de recuperação, organização, agrupamento e classificação de textos com o objetivo de facilitar o acesso ao conhecimento de forma mais eficiente. De acordo com Bonette (2011), o uso de dados disponíveis na web, alinhados com ferramentas de extração e descoberta de conhecimento, pode auxiliar as organizações em ambientes competitivos. Porém, apesar do acesso às informações relevantes ser imprescindível, obtê-las normalmente não é uma tarefa simples.

Desta forma, a necessidade de métodos que sejam capazes de melhorar o processo de obter informações úteis, intrínsecas e que estão ocultas devido à quantidade de dados, fez surgir várias pesquisas na área de análise de texto e tratamento de linguagem natural. Porém, ainda são necessários mais estudos que abordem os novos desafios para a área de indexação, sumarização, clusterização e classificação de textos escritos no idioma português.

A técnica de Mineração de Textos não é desafiadora apenas por causa do processamento da linguagem natural, mas também pela sua utilidade prática, por possibilitar a extração do conhecimento a partir de uma grande quantidade de documentos. Contudo, torna-se cada vez mais necessário implementar métodos automatizados capazes de extrair conhecimento e obter melhores resultados a partir de informações disponibilizadas na internet. Segundo Goldschmidt, Passos e Bezerra (2015), a Mineração de Textos se apresenta como uma área de grande expansão, visto que a maior parte do conhecimento humano na atualidade se encontra em formato textual, uma vez que os mecanismos de busca na web têm contribuído para amplificar a sobrecarga informacional, pois tornam novos documentos textuais rapidamente disponíveis.

Assim, o agrupamento de textos passa a ser uma tarefa necessária, pelo fato de permitir categorizar documentos automaticamente em *clusters* significativos, facilitando, desta forma, o acesso aos grupos de textos similares.

Por conseguinte, esta pesquisa pode ser considerada relevante na área de agrupamento automático de notícias da internet, uma vez que os trabalhos relacionados à

extração, agrupamento e classificação de informações publicadas na web e escritas no idioma português ainda são restritos.

Portanto, novas propostas de pesquisas, com o intuito de organizar o conteúdo da internet e encontrar maneiras para tornar as informações mais facilmente acessíveis, são necessárias.

1.2 Problema de pesquisa

A web é um dos maiores repositórios de informação do mundo contemporâneo (LEITÃO, 2004) e a quantidade de dados no formato digital tem aumento de forma exponencial (MAÇADA; CANARY, 2014). Segundo Dobre & Xhafa (2014), são gerados no mundo aproximadamente um bilhão de gigabytes, sendo aproximadamente 90% desses dados oriundos de fontes não estruturadas, e que em 2020 o universo terá gerado 40 trilhões de gigabytes (SIVARAJAH *et al*., 2017). Mas a grande questão é como encontrar o conhecimento no meio de tantas informações disponíveis? Encontrar informação relevante em ambiente de *big data* impõe limites e condições. A mesma dificuldade acontece quando se deseja obter mais clareza sobre uma notícia específica da atualidade nos jornais *on-line*. O leitor pode pesquisar em um portal específico, mas também pode ter interesse em ler informes semelhantes em outros sites para melhor compreensão do assunto (LAMA, 2013).

Porém, é desafiante localizar quais portais trazem as notícias desejadas. Normalmente, o usuário visita os sites mais prováveis de encontrar o que almeja e, em seguida, busca pelas notícias para descobrir se o assunto que ele procura está presente ou não. De acordo com Lama (2013), esta tarefa é problemática, demorada e tediosa. Pois, com a abundância de informações disponíveis na web, encontrar apenas o que é relevante não é tarefa fácil. Segundo Davenport (1998), a atenção humana tem sua capacidade limitada e alerta que para analisar e filtrar as informações, as pessoas despendem muita energia e quando a sobrecarga informacional sobrevém, o desenvolvimento de suas atividades pode sofrer sérios prejuízos.

Por outro lado, as tecnologias inteligentes, definidas por Lévy (1998) como ferramentas ou dispositivo que ampliam, modificam e exteriorizam as funções cognitivas dos sujeitos e reforçam suas habilidades intelectuais, ao tempo em que, internalizadas, passam a se constituir como suportes ao desenvolvimento de outras tecnologias, estão disponíveis e podem ser utilizadas para satisfazer o interesse dos usuários, com pouco esforço e tempo.

Diante disso, esta pesquisa busca por uma solução que possa ajudar o leitor a encontrar as notícias de seu interesse, com o mínimo de trabalho. A proposta é utilizar as técnicas de Processamento de Linguagem Natural, *Machine Learning* e *Clustering* para criar grupos de notícias semelhantes a partir da amostra de notícias capturadas pelo *Media Frame*.

Isso possibilitará ao ledor encontrar todas as notícias similares agrupadas, sem necessidade de navegar em vários sites de notícias à procura das informações almejadas. Além disso, existem poucos estudos relacionados ao tema *clustering* de notícias publicadas no idioma português. A lacuna de pesquisas nessa área acaba por reforçar e aprofundar a escassez de informação relacionada às seguintes questões: como desenvolver uma solução automatizada, capaz de recuperar e comparar as notícias em destaque na mídia, publicadas no idioma português em diferentes portais, e agrupá-las por semelhança? As soluções existentes apresentam o mesmo desempenho quando alimentadas por um corpus de notícias publicadas no idioma português, uma vez que essa língua apresenta muitas particularidades? Diante disso, surgem outras indagações: Qual tecnologia apresenta melhor resultado para uma coleção de notícias publicadas no idioma português? Os algoritmos apresentam o mesmo desempenho ao ser alimentados com corpora diversificados?

1.3 Justificativa e relevância do tema

As notícias de jornais *on-line* contêm uma grande quantidade de dados em formato não estruturado que pode ser extraída e transformada em informações valiosas, de acordo com a exigência do usuário (LAMA, 2013). Essa massa de dados tem aumentado cada vez mais, pois todos os dias, quintilhões de bytes de dados são gerados provenientes do uso de mídia social e interações digitais (DAS, 2017).

Essa questão é cada vez mais importante no mundo dos negócios e da sociedade. Por isso que a internet tem se tornado um interessante objeto de estudo, pois com a riqueza de conteúdo disponível na web, se tornam cada vez mais necessários mecanismos capazes de analisar essa abundância de dados e revelar informações de uma forma que os indivíduos dificilmente seriam capazes de identificar.

Na internet existem informações imensuráveis e muitas possibilidades ainda não exploradas, como por exemplo, os artigos de notícias publicados no idioma português. Essas publicações são fontes importantes de informação que mantêm as pessoas atualizadas com os acontecimentos atuais do mundo (LAMA, 2013). Porém, muitas vezes, a notícia de um único portal não é suficiente para obter todo o conhecimento desejado. Assim, é necessário recorrer a vários sites em busca de manchetes semelhantes, todavia, essa tarefa não é tão simples.

Segundo Liu (2007), os noticiários *on-line* geram diariamente uma grande quantidade de informes e para fornecer um serviço integrado de notícias, os artigos coletados deveriam ser dispostos em uma hierarquia de tópicos. Mas como essas informações podem ser organizadas? Uma das possibilidades seria empregar um grupo de editores humanos para fazer o trabalho. No entanto, a organização manual é dispendiosa e demorada, o que se torna

inadequada para notícias e para outras informações que são sensíveis ao tempo. E disponibilizar todas as notícias para os leitores, sem nenhuma organização, também não seria uma boa alternativa.

Assim, embora a classificação seja capaz de categorizar notícias de acordo com tópicos predefinidos, essa solução não é aplicável em todos os casos porque a classificação precisa de dados de treinamento, que devem ser rotulados manualmente com as classes referentes às subdivisões. Todavia, os assuntos das notícias mudam constantemente, tornando a rotulagem manual inviável. Logo, *clustering* é claramente uma solução para esse problema porque agrupa automaticamente um fluxo de documentos com base na similaridade do conteúdo (LIU, 2007, p. 119).

Segundo Han e Kamber (2001), a grande vantagem do uso das técnicas de *Clustering* é que ao agrupar textos, pode-se descrever de forma mais eficaz as características dos diversos grupos, o que permite um maior entendimento da coleção original, além de possibilitar o desenvolvimento de esquemas de classificação para novos documentos.

Por conseguinte, buscar novos recursos, que sejam capazes de recuperar manchetes similares, de vários portais de notícias, e disponibilizá-las em *clusters*, seria uma possibilidade muito eficiente e sofisticada para explorar informações sobre assuntos semelhantes. Por isso, a importância de um estudo na área de agrupamento de textos, com o objetivo de desenvolver estratégias para agrupar, automaticamente e de forma inteligente, informações importantes recuperadas dos principais jornais *on-line* e representá-las em grupos, facilitando, deste modo, o acesso aos informes atuais.

Assim, a área de *clustering* de textos é um objeto de estudo fascinante que busca facilitar o acesso à informação, pois o rápido aumento do volume de dados presente na internet dificultou ao ser humano acompanhar e explorar todo o conhecimento disponível. Portanto, desenvolver uma metodologia apropriada para agrupamento que possa emitir de forma prática *clusters* de notícias semelhantes e minimizar o tempo do leitor na procura da informação desejada, é de extrema relevância.

Além de tudo, há também outras motivações que impulsionam o desenvolvimento desta pesquisa, como a vontade de dar continuidade aos estudos realizados durante o mestrado, no qual foi realizada uma análise de ferramentas para Mineração de Conteúdos de páginas web. Portanto, o interesse para *clustering* é ainda maior, ao perceber a necessidade de minimizar o esforço dos usuários na busca por informações relevantes a partir da extensa quantidade de dados disponibilizados na internet.

Ademais, na Mineração de Textos, trabalhos relacionados à língua portuguesa são escassos, pois a maioria dos esforços é direcionada para a língua inglesa (WEISS *et al*., 2005; FELDMAN; SANGER, 2006; KONCHADY, 2006; LIU, 2011; SRIVASTAVA; SAHAMI, 2009;

SARKAR, 2016; BENGFORT; BILBRO; OJEDA, 2018). Contudo, em decorrência da existência de poucas ferramentas para a análise automática de textos escritos no idioma brasileiro (SOUZA; OLIVEIRA; MOREIRA, 2018), e pelo fato “do português do Brasil apresentar características peculiares que exigem experimentações específicas nas tarefas de processamento de textos” (AFONSO, 2013, p.28), estudos nesse campo ainda são necessários.

Além disso, muitos dos algoritmos de agrupamento utilizados nas pesquisas existentes, tais como os de particionamentos, agrupamentos hierárquicos e probabilísticos, foram testados com corpus em outros idiomas. Também não foram encontrados na literatura trabalhos em que os algoritmos de *clustering* fossem alimentados com notícias publicadas no idioma português e coletas de vários jornais *on-line*. Assim, “quando se trata especificamente da recuperação da informação textual em meio digital, o fator ‘língua’ acaba por exigir novas pesquisas e experimentações” (AFONSO, 2013, p. 21), o que impulsionou ainda mais o interesse por buscar novos conhecimentos nessa área de saber.

Dessa forma, as técnicas de Mineração de Textos, *Clustering, Machine Learning*, Processamento de Linguagem Natural, bem como Estudos Linguísticos e outros métodos pesquisados durante o levantamento teórico foram as questões norteadoras desta pesquisa, que tem a finalidade de avaliar as técnicas de agrupamento existentes, porém aplicadas em textos no idioma português, e, assim, desenvolver novas estratégicas, capazes de agrupar automaticamente notícias da internet e superar os desafios impostos pelo avanço e popularidade da web.

Por conseguinte, espera-se que este estudo traga uma contribuição no ponto de vista acadêmico, por colaborar com o desenvolvimento de estudos relacionados ao agrupamento de informações textuais no idioma português do Brasil, uma vez que a quantidade de informação produzida nesta língua é ampla. Além disso, a pesquisa trará uma contribuição também para o usuário final, uma vez que a solução auxiliará o leitor a encontrar as notícias de seu interesse.

1.4 Objetivo geral

O objetivo da pesquisa é testar e aprimorar uma metodologia de aprendizado não supervisionado, capaz de agrupar, automaticamente, notícias publicadas no idioma português do Brasil, postados na grande mídia e coletadas pelo *Media Frame.*

1.5 Objetivos específicos

Para cumprir o objetivo geral de uma pesquisa é necessário delimitar metas mais peculiares. Assim, este trabalho tem como objetivos específicos:

- Identificar métodos e técnicas utilizadas no processo de *clustering* de textos;

- Aplicar as técnicas de *clustering* em uma coleção de notícias publicadas na língua portuguesa.

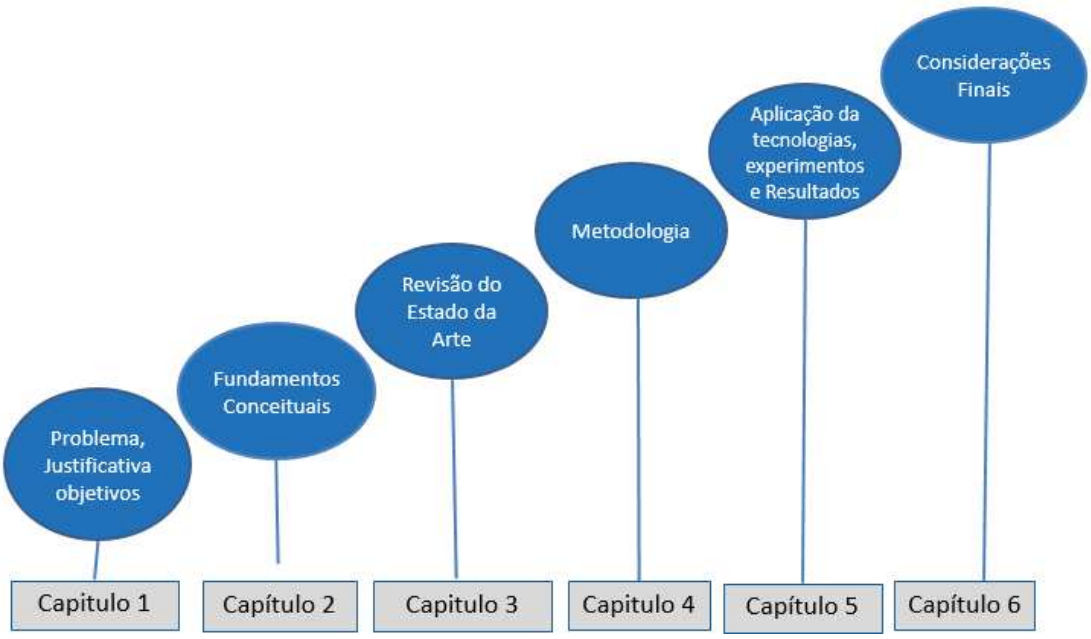
- Verificar o desempenho dos algoritmos de *clustering* (*K-Means*, *Affinity Propagation* e os algoritmos hierárquicos *Single Linkage*, *Average Linkage*, *Complete Linkage* e o *Ward's method)* ao serem alimentados por uma coleção de notícias publicadas no idioma português.

- Aplicar a metodologia em diferentes corpora, discutir o sucesso da técnica em cada caso, averiguar a possibilidade efetiva de clusterização das notícias e analisar as dificuldades encontradas para diferentes amostras.

1.6 Estrutura da tese

A tese está estruturada em seis capítulos, conforme ilustrado na Figura 1:

FIGURA 1: Estrutura da tese



Fonte: elaborada pela autora

O capítulo 1 apresenta a introdução do trabalho, bem como a contextualização do tema, além de expor a justificativa e a relevância do assunto. A problemática é apresentada em detalhes e relacionada aos objetivos desta pesquisa.

No capítulo 2 são apresentados os fundamentos conceituais relacionados aos temas: Processamento de Linguagem Natural, Mineração de Textos, Aprendizado de Máquina, Clusterização e Modelagem de Tópicos. Este capítulo também expõe o processo de *Text Mining*, sendo descritas as suas principais etapas: pré-processamento, extração de conhecimento e validação. Além disso, descreve as principais técnicas e algoritmos utilizados no processo de agrupamento de textos.

O capítulo 3 faz a revisão do estado da arte, em que são apresentados os trabalhos relacionados ao tema *Clustering* de Textos. O capítulo 4 descreve sobre a Metodologia utilizada para a realização do experimento desta tese, sendo apresentadas as tecnologias e os procedimentos realizados. O capítulo 5 relata os resultados dos experimentos e, por fim, é dissertada a conclusão desta pesquisa e as referências bibliográficas relevantes são enunciadas.

2 FUNDAMENTOS CONCEITUAIS

Neste capítulo serão abordados os principais fundamentos teóricos necessários para a compreensão desta tese de natureza interdisciplinar, bem como a própria Ciência da Informação que, segundo Moraes e Carelli (2016), é uma área que trata da interdisciplinaridade, ou seja, um movimento integrador dos saberes que busca resolver os problemas relacionados ao crescente volume de informação, ao aparecimento de novas aplicações tecnológicas e também ao novo patamar de importância que a informação, o conhecimento e o saber adquiriram com a explosão maciça dos dados.

A seção foi dividida nas seguintes subseções: Processamento de Linguagem Natural, Mineração de Textos, Aprendizado de Máquina e Clusterização, que serão descritas a seguir:

2.1 Processamento de Linguagem Natural (PLN)

Este tópico tem por objetivo expor o conceito de Processamento de Linguagem Natural (PLN) e apresentar as suas principais técnicas, tais como análise fonética, sintática, semântica, pragmática e o Rotulador de Função Gramatical. Esses recursos podem ser usados na fase de pré-processamento e visam melhorar o desempenho dos algoritmos que irão executar as tarefas de análise de textos. Além disso, este subcapítulo apresenta as diversas aplicações do PLN, o que justifica a importância da compreensão dessa ampla área.

Para entender a análise de texto e o Processamento de Linguagem Natural, em inglês *Natural Language Processing*, é necessário conhecer o conceito de linguagem natural. Segundo Sarkar (2016), uma linguagem natural é desenvolvida pelos humanos através do uso natural e da comunicação, em vez de ser construída e criada artificialmente, como uma linguagem de programação de computador. Idiomas humanos como inglês, japonês e português são línguas naturais. Essas linguagens possuem regras e ambiguidades que dependem do idioma e podem ser comunicadas em diferentes formas, incluindo fala, escrita ou mesmo sinais.

Conforme descrito por Garcia, Varejão e Ferraz (2003, p. 64),

> a linguagem natural é a maneira mais espontânea de descrever o conhecimento de seu domínio de especialização. Muitos dos nossos pensamentos são realizados usando a própria linguagem natural e grande parte da nossa comunicação ocorre por meio dela. Portanto, é bem mais fácil para as pessoas que não são da área da computação verbalizar seu conhecimento usando a linguagem natural. Para um Engenheiro do Conhecimento, a linguagem natural, apesar de poder levar interpretações errôneas, também é facilmente processável, pois possui o mesmo modelo cognitivo (humano).

Segundo esses autores, as linguagens naturais são muito utilizadas nas fases iniciais de aquisição do conhecimento, como, por exemplo, nas entrevistas. Porém,

> se por um lado a linguagem natural é confortável para o especialista e suficientemente expressiva para ele descrever quase todo o seu conhecimento, ela apresenta uma série de características que dificultam o seu processamento computacional. Sentenças em linguagem natural podem ser ambíguas, inconsistentes, dependentes de contextos, imprecisas e incompletas. Além disso, linguagens naturais são combinatoriamente explosivas, uma vez que o número de vocábulos é enorme, o número de possíveis combinações de vocábulos para a formação de sentenças é infinitamente grande e, por fim, em razão de um mesmo vocábulo ou sentença poder possuir diversos significados, dependendo do contexto aos quais são aplicados. (GARCIA, VAREJÃO e FERRAZ, 2003, p. 64).

Sendo assim, desenvolver sistemas digitais capazes de compreender a linguagem humana é um grande desafio tecnológico, pois os computadores foram preparados para entender somente linguagens formais. Com o intuito de resolver essa questão, existe um campo de estudos dentro da Inteligência Artificial, conhecido como Processamento de Linguagem Natural, que surgiu nas décadas de 1940 e 1950, quando foram iniciados os primeiros esforços para a elaboração de programas para tradução automática.

As pesquisas relacionadas ao Processamento de Linguagem Natural têm como objetivo fazer com que as máquinas possam processar e compreender textos escritos em linguagem natural. Ladeira (2010, p. 43) descreve PLN como sendo a área responsável por manipular automaticamente a linguagem não controlada contida normalmente nos documentos textuais. Tem como função tratar os aspectos da comunicação humana por meio de processamentos automatizados realizados pelo computador. É um campo de estudos que está preocupado em desenvolver técnicas computacionais para permitir que um computador entenda o significado do texto (ZHAI; MASSUNG, 2016). Para isso, faz uso de algoritmos para processar documentos de forma que os computadores possam entender sentenças escritas em linguagens humanas. É uma subárea da Computação, Inteligência Artificial (IA) e da Linguística que estuda os problemas da geração e compreensão automática de línguas naturais (DAS, 2017, p. 2).

A área de PLN é considerada um conjunto de teorias e técnicas computacionais para analisar e representar naturalmente textos em um ou mais níveis da análise linguística com o propósito de realizar o processamento da linguagem humana para um conjunto de tarefas e aplicações (SILVA, [s.d.], p. 6). É uma área cujo foco está na interação entre computadores e linguagem (natural) humana e que apresenta grandes possibilidades em relação a novas descobertas e resolução de problemas que antes eram considerados impossíveis de resolver.

O cérebro humano e as habilidades cognitivas dos seres humanos possibilitam que as pessoas tenham capacidades para trocar informações, comunicar, pensar e sentir emoções,

sem necessidade de muito trabalho para que isso aconteça. É surpreendente quando se pensa em tentar replicar essa habilidade em máquinas, porém é uma tarefa dispendiosa. Apesar dos avanços no que diz respeito à computação cognitiva e a IA, ainda há muito que evoluir nessas áreas, pois existem muitas dúvidas se o computador realmente pode replicar um humano em todos os aspectos (SARKAR, 2016). Mas graças aos estudos já realizados, os computadores já são capazes de efetuar muitas atividades que antes eram feitas somente pelos humanos e que necessitavam muito esforço e tempo.

Atualmente, a grande demanda está em relação aos aplicativos de PLN e análise de texto, pois a capacidade de extrair informações úteis e *insights* acionáveis de uma grande quantidade de dados textuais não estruturados e brutos é uma tarefa complexa. Talvez o maior problema em relação a análises de texto não seja a falta de informação e sim a sua quantidade, muitas vezes chamada de sobrecarga informacional (SARKAR, 2016). Logo,

> PLN busca compreender a linguagem utilizada naturalmente pelos humanos. Essa busca para tornar o computador uma máquina que compreende e se comunica de forma semelhante a humanos vem ganhando cada vez mais ênfase na academia e indústria. O uso de PLN propicia a criação de interfaces de comunicação mais intuitivas para os usuários. Nesse cenário, diversas ferramentas e tecnologias surgiram nos últimos anos proporcionando uma maior facilidade para o desenvolvimento de aplicações com o uso de PLN. No entanto, o uso de linguagem natural ainda não é popular em softwares, pois existe uma barreira devido à complexidade para o desenvolvimento de aplicações que fazem uso de tal abordagem. (ANDRADE; BARROS; SANTOS [s.d.], p. 1).

Segundo Aranha e Passos (2006), o Processamento da Linguagem Natural é uma técnica importante para *Text Mining* por utilizar os conhecimentos da área da linguística para aproveitar ao máximo o conteúdo do texto, extraindo entidades, seus relacionamentos, detectando sinônimos, corrigindo palavras escritas de forma errada e ainda as desambiguizando. Participa, normalmente, na parte do pré-processamento dos dados, transformando-os em um formato que seja mais compreensível pelas máquinas.

Na visão de Gonzalez e Lima (2014, p. 5, grifo nosso),

> O PLN trata computacionalmente os diversos aspectos da comunicação humana, como sons, palavras, sentenças e discursos, considerando formatos e referências, estruturas e significados, contextos e usos. Em sentido bem amplo, podemos dizer que o PLN visa fazer o computador se comunicar em linguagem humana, nem sempre necessariamente em todos os níveis de entendimento e/ou geração de sons, palavras, sentenças e discursos. Estes níveis são: - **fonético e fonológico**: do relacionamento das palavras com os sons que produzem; - **morfológico**: da construção das palavras a partir unidades de significado primitivas e de como classificá-las em categorias morfológicas; **- sintático**: do relacionamento das palavras entre si, cada uma assumindo seu papel estrutural nas frases, e de como as frases podem ser partes de outras, constituindo sentenças; - **semântico**: do relacionamento das palavras com seus significados e de como eles são combinados para formar os significados das sentenças; - **pragmático**: do uso de frases e sentenças em diferentes contextos, afetando o significado.

Por fim, para melhor compreensão do conceito de PLN, a próxima subseção trará uma melhor explicação desses níveis, além de apresentar algumas técnicas e métodos que podem ser usados para melhorar os resultados do processamento de textos.

2.1.1 Métodos e Técnicas de análise no Processamento de Linguagem Natural

As técnicas de PLN podem ser utilizadas para produzir melhores resultados no processamento de dados. Tais recursos têm como objetivo melhorar o desempenho dos algoritmos que executarão as tarefas de análise, pois enquanto um humano pode entender instantaneamente uma sentença em sua língua nativa, é bastante desafiador para um computador entender uma linguagem natural. Em geral, isso pode envolver as seguintes fases:

2.1.1.1 Análise Fonética

A análise linguística denomina de Fonologia "o estudo do efeito acústico das formas sonoras da língua. "O nível fonológico trata da acústica e da articulação da fala" (SILVA, 2008, p. 23). Já a Fonética ocupa-se da descrição dos sons da fala e das condições pelas quais esses sons são reconhecidos e produzidos pelos falantes de uma língua" (SILVA et al. 2007, p. 18). Para Guarnier (2018, p. 38), "análise fonética, ou fonologia, se define como o reconhecimento de sons presentes nas palavras ou o estudo dos sons que compõem as palavras em um determinado idioma". De acordo com Liddy (2001), o nível fonológico está relacionado com a interpretação dos sons da fala por meio da pronúncia.

Sarkar (2016) define fonética como o estudo das propriedades acústicas dos sons produzidos pelo trato vocal humano durante a fala. Inclui estudar as propriedades dos sons, bem como eles são criados pelos seres humanos. Para o autor, a menor unidade individual da fala humana em uma linguagem específica é chamada de fonema.

Segundo Silva *et al*. (2007), o sistema sonoro pode ser representado por meio dos fonemas de uma língua natural. Em PLN, a representação e operacionalização dos fonemas são particularmente importantes para o tratamento dos sons produzidos pelo falante de um idioma para determinar os paradigmas sonoros, tais como as alterações de timbre e intensidade das palavras.

> Quando a máquina opera com o registro escrito, o conhecimento fonético-fonológico ganha importância na determinação dos paradigmas sonoros das palavras da língua, bem como as alterações de timbre e intensidade das palavras motivadas por interferência entre os sons concorrentes do vocábulo. Alguns fatos linguísticos podem ilustrar a complexidade do tratamento sonoro das palavras pela máquina: 1. a variação de timbre segundo a caracterização regional das palavras. No Brasil as vogais /e/ e /o/ em posição pretônica são

> pronunciadas de forma “aberta” na região nordeste, ao passo que na região sul e sudeste as mesmas vogais são “fechadas”. Exs.: feriado; coração. 2. a realização sonora de determinadas formas segundo suas posições na palavra. Nos exemplos a seguir podemos depreender três sons distintos representados pela mesma forma ortográfica: xadrez ≠ êxodo ≠ inox. 3. as palavras homófonas (aquelas com mesma forma sonora com significado diferente). Exs.: para / pára; pelo / pêlo. Esses casos acima demonstram a necessidade do conhecimento das especificidades da cadeia sonora de uma língua a fim de que a ferramenta computacional opere adequadamente com os fonemas. O esforço primordial nesse nível de processamento, porém, está na melhor representação fonética das palavras da língua, assim como a estipulação das restrições fonológicas que cada tipo de som acarreta para o sistema sonoro. (*SILVA et al*., 2007, p. 18).

Portanto, a análise fonética é bastante desafiadora, pois as ondas sonoras precisam ser processadas para a interpretação da linguagem especifica utilizada. Este tipo de processamento é normalmente utilizado em sistemas de reconhecimento de voz, que não é o objeto de estudo desta pesquisa.

2.1.1.2 Análise Morfológica

Sarkar (2016) define morfema como a menor unidade de linguagem que possui um significado distinto. Isso inclui coisas como palavras, prefixos, sufixos e assim por diante, que têm seus significados diferentes. E conceitua morfologia como o estudo da estrutura e significado dessas unidades ou morfemas em uma linguagem. Regras e sintaxes específicas geralmente governam a forma como os morfemas podem se combinar (SARKAR, 2016).

Hack *et al*. (2013, p. 7) afirmam que “o objetivo da análise morfológica do texto é identificar a classe morfológica de uma palavra (substantivo, verbo, pronome, etc.), bem como de sua inflexão, seja ela nominal (ex. gênero) ou verbal (ex. pessoa) ”. “A análise morfológica é responsável por definir artigos, substantivos, verbos e adjetivos, armazenando-os em um tipo de dicionário.” (SANTOS *et al*. 2014, p. 117).

Silva *et al*. (2007) alegam que as palavras de uma língua também podem ser segmentadas em unidades mínimas em termos de seu significado (gramatical ou lexical) denominadas morfemas. “Na língua portuguesa, a morfologia distingue-se em gramatical e lexical” (REIS, 2017, p. 18). De acordo com Luft (2008), na morfologia lexical são tratados os problemas como origem, formação e estrutura das palavras. A morfologia gramatical ocupa-se com a classificação das palavras, categorias gramaticais (gênero, número, grau, pessoa, modo, tempo, aspecto), paradigmas flexionais etc.

Segundo Silva (2008), a análise morfológica trata da estrutura do formato da palavra, o que inclui a classificação do termo de acordo com sua *Part of Speech* (PoS), como por exemplo a categoria gramatical e a descrição da estrutura do vocábulo em termos de inflexão, derivação e composição. Em linguística computacional, a “morfologia aparece no contexto de

reconhecimento automático de formação de palavra por meio da lematização, que relaciona a forma do termo com sua forma básica, e a categorização, que caracteriza as propriedades morfossintáticas da palavra.” (SILVA, 2008, p. 23).

Silva *et al.* (2007, p. 19) apresentam alguns exemplos de análise morfológica das palavras:

1. *pedra* => -a: morfema gramatical que indica os nomes terminados em “-a”.

2. *macaca* => -a: morfema gramatical que indica gênero feminino.

3. *procuramos* => -a: morfema gramatical que indica a primeira conjugação verbal;

-mos: morfema gramatical que indica primeira pessoa do plural do presente do indicativo.

4. *operação* => -ção: morfema lexical que indica evento.

5. *incerto* => in-: morfema lexical que indica negação.

Segundo Hack *et al*. (2013), uma das técnicas que também pode ser utilizada para este tipo de análise é a utilização de tabelas de afixos, que associa sufixos e prefixos com radicais de palavras.

O Quadro 1 mostra os sufixos associados aos radicais das palavras. Esse tipo de estrutura permitiria identificar, por exemplo, que a palavra ‘cafezinho’ é uma derivação da palavra café e está no diminutivo, pelo fato de ter utilizado o sufixo ‘-zinho’.

QUADRO 1: Exemplos de sufixos

Sufixo	Substantivo (radical)	Palavra derivada
-zinho	Café	Cafezinho
	Lugar	Lugarzinho
-zinha	Flor	Florzinha
	Lâmpada	Lampadazinha

Fonte: Adaptada de Hack *et al.* (2013, p. 8)

Por conseguinte, a determinação exata dos segmentos morfológicos e as relações que eles implicam na definição da palavra são particularmente importantes para o processamento linguístico. “O fenômeno da concordância, por exemplo, é uma dessas relações que exige a presença de certo morfema e não outro no interior da palavra, já que elas não são isoladas no texto.” (SILVA *et al*., 2007, p. 19).

2.1.1.3 Análise Sintática

A sintaxe é o estudo dos princípios e processos pelos quais as orações são construídas em uma determinada linguagem. Segundo Sarkar (2016), a sintaxe é geralmente

o estudo de sentenças, frases, palavras e suas estruturas. Isso inclui estudar como os vocábulos são combinados gramaticalmente para formar as frases, visto que a ordem sintática das palavras usadas em uma sentença é importante porque dependendo da posição de um termo, o sentido da frase pode mudar completamente.

De acordo com Chomsky (2002), a pesquisa sintática de uma determinada linguagem tem como objetivo a construção de uma gramática que possa ser vista como uma ferramenta que permite gerar as sentenças da linguagem e estudar as propriedades gramaticais que fazem isso efetivamente. Segundo Ahmad (2007), a sintaxe envolve a aplicação das regras da gramática de cada idioma, sendo que sua tarefa é determinar o papel de cada palavra em uma frase e organizar esses dados em uma estrutura que seja mais facilmente manipulada para análise posterior. “O propósito da análise sintática é determinar como as palavras são relacionadas umas com as outras em uma frase, revelando assim a estrutura sintática de uma sentença.” (ZHAI; MASSUNG, 2016, p. 39).

Para Silva (2008, p. 23), “o nível sintático trata, basicamente, da composição dos formatos de palavras em sentenças gramaticalmente bem formadas em termos de regras gramaticais”.

> Quando as palavras são combinadas entre si para formar um enunciado dotado de um sentido completo, sua distribuição na sentença não ocorre de maneira aleatória, mas, ao contrário, essa disposição segue regras estruturais bastante definidas. Essas regras determinam, por exemplo, o emprego dos pronomes, a aplicação da crase, a realização da concordância. Na manipulação dessas regras, faz-se uso de um conjunto de categorias definido em termos da sua função sintática, das quais são exemplos as categorias sujeito, objeto direto, complemento nominal, adjunto adverbial e assim por diante. (SILVA *et al*. 2013, p. 19).

Para Salton (1968), o conhecimento de propriedades sintáticas das palavras é importante para reconhecer certas relações que existem entre termos dentro das frases, por exemplo, combinações de sintagmas nominais, preposicionais, adverbiais e agrupamentos simples de sujeito-verbo-objeto. A análise sintática faz uso do dicionário criado na análise morfológica e procura mostrar os relacionamentos entre as palavras e, num segundo momento, verifica sujeito, predicado, complementos nominais e verbais, adjuntos e apostos (SANTOS *et al.*, 2014).

A análise sintática (*parsing*) é o procedimento que avalia os vários modos de como combinar regras gramaticais, com a finalidade de gerar uma estrutura de árvore que represente a estrutura sintática da sentença analisada. Se a frase for ambígua, o analisador sintático (*parser*) irá obter todas as possíveis estruturas sintáticas que a representam (GONZALEZ; LIMA, 2014, p. 6). A seção 2.1.1.6 apresentará com mais detalhes o Rotulador de função gramatical (PoS), que tem como finalidade identificar a função ou categoria sintática

de cada termo existente nas sentenças do documento em análise. Isso é útil para muitas tarefas de Processamento de Linguagem Natural.

2.1.1.4 Análise Semântica

Na análise semântica ocorre o encontro de termos ambíguos, de sufixos e afixos, ou seja, questões de significados associados aos morfemas componentes de um vocábulo, o sentido real da frase ou palavra (SANTOS, *et al*., 2014). O nível semântico está relacionado ao significado das palavras em busca de alcançarem certo sentido no escopo da sentença, não apenas nas expressões como uma unidade completa, mas nas suas unidades constitutivas (REIS, 2017).

Para Zhai e Massung (2016), o objetivo da análise semântica é determinar o que uma sentença quer dizer. Isso normalmente envolve a avaliação do conteúdo de uma frase inteira ou de uma unidade maior com base nos significados das palavras e na sua estrutura sintática. Na visão de Vieira e Lima (2001), a área da semântica é mais nebulosa do que o campo da sintaxe, por apresentar questões que são difíceis de tratar de maneira exata e completa, pois, segundo os autores, o significado está associado ao conhecimento de mundo e, além disso, ligado a pontos mais obscuros como os estados mentais e a consciência.

> Pode-se perceber que o estudo da semântica interpretará o significado das palavras utilizadas na comunicação tanto em palavras individuais quanto em expressões ou frases da linguagem natural. Portanto, no processamento semântico serão considerados os problemas enfrentados com os múltiplos sentidos (ambiguidade) de algumas palavras, por exemplo, a palavra “banco” que pode referir-se tanto a instituição financeira, quanto ao assento. (REIS, 2017, p. 19).

Para simplificar o estudo da semântica, costuma-se fazer determinados recortes teóricos que, consequentemente, limitam o poder de alcance das teorias propostas. Deste modo, os estudos do significado que procuram integrar outros fatores, como contexto e falantes, constituem outra área do conhecimento denominada pragmática (VIEIRA; LIMA, 2001).

2.1.1.5 Análise pragmática ou do discurso

Análise Pragmática é o estudo de como fatores linguísticos e não linguísticos podem afetar o sentido de uma expressão, de uma mensagem ou de um enunciado. Isso inclui tentar inferir se existem significados ocultos ou indiretos na comunicação (SARKAR, 2016). Segundo Zhai e Massung (2016), o objetivo da análise pragmática é determinar o significado no contexto para entender os atos da fala. Para esses autores, a linguagem natural é usada pelos homens para que eles possam comunicar-se uns com os outros. Assim, para entender melhor

o propósito da comunicação é necessária uma compreensão mais profunda da linguagem humana.

A análise pragmática tem como objeto estudar o significado de uma sentença que integra a diferença entre o significado literal da linguagem e o significado da linguagem em uso, ou seja, o contexto do falante na comunicação (REIS, 2017). Para Vieira e Lima (2001), na análise pragmática são estudadas questões ligadas ao uso da linguagem, abordando-se aquilo que é relativo a quem usa e ao contexto de uso. “Sistemas que trabalham nesse nível de representação costumam considerar o contexto linguístico (discurso) na interpretação das expressões da língua.” (VIEIRA; LIMA, 2001, p. 2).

A análise do discurso é necessária quando uma grande parte do texto com múltiplas sentenças deve ser examinada. Em tal caso, as conexões entre essas orações devem ser consideradas e a avaliação de uma frase individual deve ser colocada no contexto apropriado envolvendo outras expressões (ZHAI; MASSUNG, 2016). Na visão de Sarkar (2016), a análise do discurso avalia a linguagem na forma de frases e a troca de informações em conversas entre os seres humanos. Essas interlocuções podem ser faladas, escritas ou mesmo assinadas.

2.1.1.6 Rotulador de função gramatical (PoS)

O rotulador gramatical tem como objetivo identificar a função ou categoria sintática (substantivos, verbos, adjetivos e advérbios) de cada termo existente nas sentenças do documento em análise. Isso é útil para muitas tarefas de PLN.

> Como muitos termos têm mais de uma categoria sintática, o rótulo visa determinar qual destas categorias é a mais provável para um uso particular da palavra na sentença. Esta função também é conhecida como *Part-Of-Speech Tagging* ou simplesmente PoS e pode ser considerada como uma classificação ontológica primitiva, presente em quase todas as línguas. (BASTOS, 2006, p. 26)

O PoS é uma marcação especial atribuída a cada *token* (palavra) em um corpus de texto para indicar a parte do discurso e muitas outras categorias gramaticais, como tempo, número (plural / singular) etc. Essas etiquetas são usadas em algoritmos de buscas e em ferramentas de análise de documentos e têm convenções diferentes para marcar palavras em um corpus. O Quadro 2 apresenta alguns exemplos de *tags* e seus respectivos significados.

QUADRO 2: Exemplos de PoS

TAG	SIGNIFICADO	EXEMPLOS
ADJ	Adjetivo	novo, bom, alto, especial, grande
P	Preposição	a, ante, após até, com, contra, de, desde, em, entre
ADV	Advérbio	realmente, já, ainda, cedo, agora
CNJ	Conjunção	e, ou, mas, se, enquanto, embora
DET	Artigo	um, uma, alguns, mais, todos, não, quais
NOUN	Substantivo	ano, casa, custo, tempo, áfrica
NUM	Numeral	vinte, quarto, 2018, 13:30
PRO	Pronome	ele, ela, seus, meus, nós
PROPN	Nome próprio	João, Apple, Brasil
V	Verbo	é, digamos, somos, seria, cantando
VD	Verbo Pretérito	disse, levou, perguntou, fez
.	Sinais de pontuação	.,;!

Fonte: adaptação de Bird, Klein e Loper (2009, p. 183)

Nos experimentos desta pesquisa, será utilizada a linguagem de Programação Python, por se tratar de uma linguagem *open source,* que é muito usada na área de Processamento de Linguagem Natural. E para testar a marcação PoS, utilizou-se o framework spaCy[2] pelo fato dele conter diversos modelos e uma arquitetura profissional para trabalhar com várias tarefas de PLN (BANDEIRA, 2018). Segundo Bandeira (2018), a spaCy é uma biblioteca desenvolvida para Python que tem como objetivo auxiliar na criação de aplicações que conseguem processar e interpretar grandes volumes de texto, podendo também ser usada no pré-processamento e na extração de informações. Desse modo, ao testar a frase "João chutou a bola vermelha" com a spaCy, as palavras foram classificadas da seguinte forma:

FIGURA 2: Exemplo de marcação PoS

```
[(u'João', u'PROPN'),
 (u'chutou', u'VERB'),
 (u'a', u'DET'),
 (u'bola', u'NOUN'),
 (u'vermelha', u'ADJ')]
```

Fonte: Elaborada pela autora

2 https://spacy.io/

Apesar de parecer fácil construir um sistema capaz de codificar toda a informação presente em textos, durante muitas décadas, compilar esse conhecimento em um modelo de Aprendizado de Máquina foi um problema de PLN muito complexo. Mas, atualmente, os algoritmos de marcação PoS podem prever a função gramatical da palavra com bastante precisão. Todavia, ainda há muitas pesquisas nessa área que objetivam melhorar o processo de marcar uma palavra em um texto como correspondente a uma parte específica do discurso, com base em sua definição e em seu contexto.

Zhai e Massung (2016) afirmam que embora o conhecimento linguístico seja sempre útil, os mais avançados métodos de Processamento de Linguagem Natural tendem a depender mais do uso de técnicas estatísticas de Aprendizado de Máquina, com o conhecimento linguístico desempenhando apenas um papel secundário. Essas metodologias são bem-sucedidas para algumas tarefas de PLN, como por exemplo, parte da marcação da fala é uma tarefa relativamente fácil e os mais recentes marcadores podem ter uma precisão muito alta (acima de 97%). Já a análise é mais difícil, embora uma verificação parcial possa ser feita com precisão razoavelmente alta (por exemplo, acima de 90% para reconhecer frases nominais). No entanto, a análise de estrutura completa permanece muito difícil, principalmente devido às ambiguidades. A análise semântica é ainda mais complexa, sendo bem-sucedida apenas em alguns aspectos da análise, como, notadamente, na extração de informações (reconhecimento de entidades nomeadas como nomes de pessoas e organizações e relações entre entidades como quem trabalha em qual organização), desambiguação do sentido da palavra (distinguir sentidos diferentes de uma palavra em diferentes contextos de uso) e análise de sentimento (reconhecer opiniões positivas ou negativas sobre um produto) (ZHAI; MASSUNG, 2016).

Apesar das dificuldades que envolvem o campo de PLN, já existem muitas pesquisas desenvolvidas nesta área do saber, a próxima seção apresenta as principais aplicações dessa esfera de conhecimento.

2.1.2 Aplicações do Processamento de Linguagem Natural

Muitas aplicações já utilizam PLN como, por exemplo, Corretores Ortográficos (Microsoft Word), *Engines* de Reconhecimento de Voz (Siri, Google *Voice*), Classificadores de Spam, Mecanismos de Busca (Google), Tradução automática (Google Tradutor), Sistemas de Inteligência Artificial como Assistentes Pessoais (DAS, 2017), analisadores sintáticos e semânticos, processadores automáticos de dicionários e gramáticas (LADEIRA, 2010).

Segundo Ladeira (2010), dentre as inúmeras aplicabilidades de PLN, algumas estão voltadas para o desenvolvimento da própria área de Processamento de Linguagem Natural, enquanto outras são mais práticas e procuram atender um público mais amplo. Como

aplicações voltadas para o campo de PLN, Ladeira (2010) apresenta o Processamento Automático de *Thesauro* e Análise Sintática Automática. E como aplicações práticas, a autora cita os Tradutores Automatizados, Respondedores Automáticos, Análise de Estilo, Geração automática de Linguagem (Sumarização) e Recuperação da Informação.

Salton (1968) cita alguns trabalhos que fazem uso **automático de dicionários** e traz como exemplos o dicionário inglês-inglês usado para classificar palavras em grupos morfológicos, como parte de um programa automatizado de controle de vocabulário, os dicionários mecanizados voltados para análise morfológica e semântica completa da linguagem e também os dicionários automáticos multilíngues como componentes de um sistema de tradução semiautomático.

Em relação à **Análise Sintática Automática**, Salton (1968) afirma que existem vários trabalhos que abarcaram a análise sintática em modo interativo, em que as regras são aplicadas uma a uma e a sua aplicação na derivação de estruturas profundas ou de superfície para várias sentenças de entrada é demonstrada. O usuário tem a opção de aceitar ou rejeitar as regras e, assim, refinar a gramática.

De acordo com Sarkar (2016), a **tradução automática** foi uma das primeiras grandes áreas de pesquisa do campo de PLN, sendo uma das aplicações mais cobiçadas dessa área. É um processo em que a tradução de textos em linguagem natural é realizada por máquina. Tem como objetivo fornecer tradução sintática, gramatical e semanticamente correta entre dois idiomas. Inicialmente, o processo de tradução automática envolvia apenas a substituição simples de palavras de um idioma para outro e ignorava a estrutura gramatical da frase. Mas com o surgimento de novas pesquisas, as técnicas evoluíram e ficaram mais sofisticadas ao longo do tempo, incluindo a combinação de grandes recursos de corpus de texto, juntamente com técnicas estatísticas e linguísticas (SARKAR, 2016).

Aplicativos de reconhecimento de voz também utilizam PLN para responder perguntas, fazer recomendações e executar ações. O Siri, por exemplo, é um aplicativo no estilo de assistente pessoal exclusivo da Apple que está focado em aplicações de Inteligência Artificial com o objetivo de auxiliar os usuários em suas atividades básicas, como, por exemplo, fazer uma ligação telefônica.

Outro exemplo de aplicação que emprega PLN são os **corretores ortográficos**. Eles são úteis para tratar erros de digitação. Editores de textos, como o *Microsoft Word*, utilizam essa tecnologia.

Em relação aos **respondedores automáticos**, a ênfase maior está nos **sistemas de perguntas e respostas**. Nesse contexto, Gazzola (2011) desenvolveu uma aplicação web capaz de identificar e classificar opiniões em textos extraídos de um ambiente virtual de aprendizagem.

Outra aplicabilidade de PLN é a **classificação de textos**. Santos (2015) afirma que existem diversas técnicas de Processamento de Linguagem Natural que são usadas nas etapas de classificação de documentos. Elas são baseadas no fato de textos de diferentes categorias se distinguirem por propriedades da linguagem natural contidas em cada documento. As principais características para classificação de textos podem ser descobertas a partir da estrutura das palavras, da frequência das palavras e da estrutura da linguagem natural em cada documento. Esses sistemas existem em diversas aplicações como, por exemplo, na categorização de e-mails e remoção de spam, divulgação seletiva de informações aos consumidores de conhecimento, classificação automática de artigos científicos, identificação do tipo de documento, atribuição da autoria de documentos, entre outras aplicações (SANTOS, 2015).

Para finalizar as discussões referentes às aplicabilidades de PLN, destaca-se a **Recuperação de Informação (RI)**, uma área diferenciada que tem características diferentes dos tradutores automáticos ou dos sistemas de perguntas e respostas. Haas (1996) afirma que a RI inclui pelo menos quatro diferentes aplicações: recuperação de documentos, recuperação de parágrafos, classificação de documentos e extração de informação.

Para Haas (1996), as técnicas de PLN podem ser utilizadas em vários pontos e processos de RI, mas talvez sejam mais comumente usadas na criação e na união de representações do documento e da consulta. Segundo Gonzalez e Lima (2013), o PLN está presente em diferentes níveis e nas diversas abordagens que os pesquisadores têm procurado para solucionar o problema da RI. O conhecimento linguístico pode, principalmente através de processamentos morfossintático e semântico, trazer estratégias inteligentes para a área, tanto através de métodos estatísticos quanto pela aplicação linguística. Porém, o Processamento de Linguagem Natural tem ainda muitos desafios a vencer, mas, por certo, tem muitos benefícios a oferecer. Já é usado em muitas aplicações e embora muito se tenha avançado nesse campo de estudo, é fato que ainda há muito por ser feito e novas pesquisas nessa área ainda são necessárias.

2.2 Mineração de Textos

A Mineração de Textos, também conhecida como *Text Data Mining* ou *Knowledge Discovery in Texts*, é considerada uma evolução da área de Recuperação de Informações (SALTON; MCGILL, 1983). É um campo multidisciplinar que se baseia em Mineração de Dados (MD), Aprendizado de Máquina, RI, Linguística Computacional e Estatística. Pode ser vista como uma extensão da área de Mineração de Dados, focada na análise de textos. Ou seja, a Mineração de Textos busca desenvolver técnicas e processos para descoberta automática de conhecimentos valiosos a partir de uma coleção de documentos.

Na visão de Aranha e Passos (2006, p. 2), a Mineração de Textos “é um conjunto de métodos usados para navegar, organizar, achar e descobrir informação em bases textuais.” . Para Maia e Souza (2010), *Text Mining* constitui a extração de informações sobre padrões em grandes coleções de documentos.

De acordo com Fayyad *et al*. (1996), a Mineração de Dados é o processo não trivial de descoberta de padrões válidos, novos, úteis e compreensíveis nos dados. E a Mineração de Textos é uma especialização desse campo, uma área interdisciplinar que reúne Processamento de Linguagem Natural, Aprendizado de Máquina e Visualização da Informação. (NASSIF, 2011, p. 7).

> Mineração de Texto é um processo de descoberta de conhecimento que utiliza técnicas de análise e extração de dados a partir de textos, frases e palavras. Envolve a aplicação de algoritmos que processam textos e então identificam informações úteis e implícitas, que normalmente não poderiam ser recuperadas utilizando métodos tradicionais de consulta, pois a informação contida nestes textos não pode ser obtida de forma direta, uma vez que, em geral, estão armazenadas em formato não estruturado. (MORAIS; AMBRÓSIO, 2007, p. 1)

Ebecken, Lopes e Costa (2003) definem *Text Mining* como um conjunto de técnicas e processos que descobrem conhecimento inovador em textos. E, ao contrário da Mineração de Dados, a Mineração de Textos lida com dados intrinsecamente não estruturados, sendo necessário realizar o pré-processamento para representar o corpus em uma forma mais fácil de trabalhar computacionalmente (NASSIF, 2011).

No entanto, na Mineração de Dados, as informações são armazenadas em um formato estruturado. Assim, a função do pré-processamento é limpar e normalizar dados. Já na Mineração de Texto, as operações de pré-processamento se concentram na identificação e extração de recursos representativos de documentos em linguagem natural. Essas intervenções são responsáveis por transformar dados não estruturados, armazenados em coleções de documentos, em um formato intermediário mais estruturado explicitamente, o que não é uma preocupação para a maioria dos sistemas de MD.

Por conseguinte, a Mineração de Texto utiliza-se muito das técnicas de Mineração de Dados, mas também se baseia em avanços feitos em outras disciplinas de Ciência da Computação relacionadas com o tratamento da linguagem natural. Talvez mais notavelmente, as técnicas e metodologias de explorações de texto das áreas de recuperação de informações, extração de informações e linguística computacional baseada em corpus (FELDMAN; SANGER, 2006). Segundo Waegel (2006), *Text Mining* é uma área interdisciplinar que não interage apenas com *Data Mining*, Aprendizagem Máquina e Estatística, mas também com Linguagem Computacional ou Processamento da Linguagem Natural.

As tarefas de Mineração de Textos, com base nas pesquisas de Weiss *et al*. (2005); Shi e Kong (2009); Sundari e Sundar (2017), são apresentadas na Figura 3.

FIGURA 3: Tarefas de Mineração de texto

Fonte: Elaborada pela autora

Para realizar a Mineração de Textos, têm-se disponíveis diversas tarefas como Recuperação da Informação, Classificação, Extração de Informação, Sumarização, Clusterização, Modelagem de Tópicos dentre outras. As atividades de *Clustering*, por serem mais pertinentes ao contexto dessa pesquisa, serão descritas na seção 2.3.2.2.

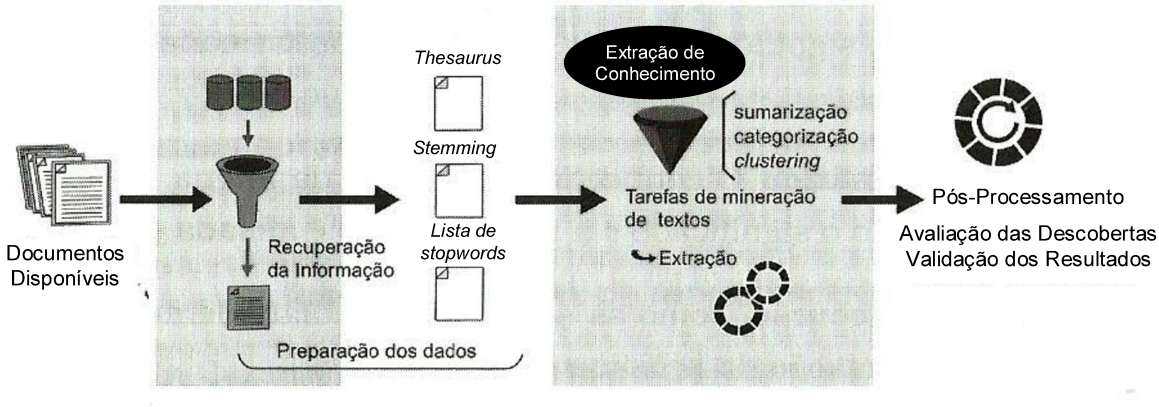
Contudo, o processo de Mineração de Textos pode conter várias etapas: pré-processamento, a redução da dimensionalidade, a extração de padrões e a validação e interpretação dos resultados. A seção 2.2.1, além de descrever esses passos, apresenta também as tarefas de tokenização, remoção das *stop words* e o *stemming,* tarefas que acontecem da fase do pré-processamento.

2.2.1 Processo de Mineração de Textos

Esta seção descreve as etapas necessárias para a análise de textos, bem como mostra as suas diferenças e apresenta uma visão clara da importância de cada uma delas.

Além de expor os conceitos que servirão como bases para diferentes metodologias utilizadas nesse processo. A Figura 4 ilustra os passos necessários para a análise de dados não estruturados. Após a coleta dos documentos, realiza-se a preparação do corpus, fase em que as *stop words* são removidas e a técnica de *stemming* é aplicada. Posteriormente, efetua-se o processamento para extração do conhecimento. Nessa fase, algumas atividades como sumarização, categorização e clusterização podem ser realizadas. Por fim, faz-se a análise e validação dos resultados obtidos (EBECKEN; LOPES; COSTA, 2003).

FIGURA 4: Etapas da Mineração de Textos



Fonte: Ebecken; Lopes; Costa (2003, p. 339)

Neste trabalho, a análise de texto foi realizada em um corpus de notícias coletado dos principais jornais *on-line*, publicado na Língua Portuguesa Brasileira, e aplicada no contexto de clusterização, com o objetivo de identificar automaticamente textos similares e assim, agrupá-los por semelhança. Para isso, a técnica de Mineração de Texto foi dividida em três etapas: Pré-processamento, Extração do conhecimento, Validação e Interpretação dos Resultados.

2.2.1.1 Pré-processamento

Para Silva (2010), o pré-processamento inclui todas as rotinas, processos e métodos necessários para a preparação dos dados. É nessa fase que os dados textuais são padronizados e representados em um formato adequado para extração do conhecimento (FELDMAN; SANGER, 2006). Tem a função de reduz o vocabulário e tornar os dados menos dispersos, característica essencial para o processamento computacional. Segundo Feldman e Sanger (2007), o objetivo do pré-processamento é extrair de textos escritos em língua natural, inerentemente não estruturados, uma representação concisa e organizada que seja

manipulável por algoritmos de *Text Mining*. Para esse fim, são executadas atividades de tratamento e padronização no corpus, seleção dos termos mais significativos e, por fim, a coleção textual é representada em um formato estruturado que preserve as principais características dos dados.

Muitos autores consideram o pré-processamento como a principal e mais cara etapa do processo de Mineração de Texto, fase em que o documento deve ser transformado em formato estruturado, como uma tabela de atributo-valor. Assim, o processo resultante, ou seja, a normalização linguística, é geralmente usado para encontrar os atributos dessa tabela.

Desse modo, quando se deseja trabalhar com coleções muito grandes, é interessante reduzir o corpus em um conjunto de palavras mais representativas. “Isso pode ser feito eliminando-se as *stop words* (como artigos e preposições), aplicando-se *stemming* (que reduz palavras distintas a sua raiz gramatical comum) e identificando-se os grupos de substantivos (que elimina adjetivos, advérbios e verbos).” (BAEZA-YATES; RIBEIRO-NETO, 2011, p. 28). Apesar dessas tarefas exigirem um trabalho exaustivo, elas têm influência diretamente na qualidade do resultado da análise.

À vista disso, antes de aplicar as técnicas de *clustering* nos textos de uma coleção, algumas tarefas de pré-processamento são normalmente executadas. Assim, nesta pesquisa, para a padronização e limpeza dos dados, primeiramente, os textos foram convertidos em letras minúsculas. Em seguida, aplicou-se a técnica de tokenização, removeram-se os caracteres especiais como sinais de pontuação, hifens e números, eliminaram-se as *stop words* e reduziram-se as palavras aos seus radicais através da aplicação da técnica de *stemming*. Devido à importância dessa etapa, as técnicas de pré-processamento de textos serão descritas em novos tópicos.

2.2.1.1.1 Tokenização

A tokenização é uma das primeiras atividades a ser executada durante a fase de pré-processamento do texto. Para Feldman e Sanger (2006), a tokenização envolve a quebra do texto em frases ou palavras. Consiste na identificação e separação dos caracteres que compõem cada símbolo ou palavra no texto, em que cada termo é separado por espaços, vírgulas, pontos, etc. Cada grupo de caracteres obtido é chamado de *token*, e a sequência de *tokens* forma uma sentença que corresponde ao texto original. (BASTOS, 2006, p. 21).

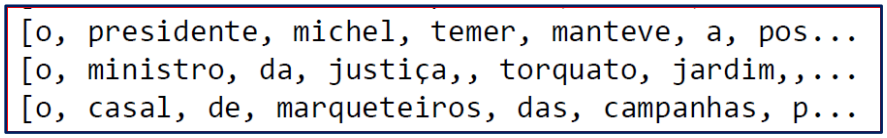
Sarkar (2016) define *tokens* como componentes textuais independentes e mínimos, que possuem alguma sintaxe e semântica definitivas. Um parágrafo ou um documento de texto tem vários componentes que podem ser separados em cláusulas, frases e palavras. Sendo assim, para dividir um corpus em sentenças e cada frase em palavras, é utilizada a

técnica de tokenização. Ou seja, nesta etapa, o texto é quebrado em componentes significantes menores chamados *tokens*.

Segundo Weiss *et al*. (2005), para analisar uma coleção de textos não estruturados, o primeiro passo é quebrar os textos em palavras, precisamente, *tokens*, pois isso é fundamental para todas as aplicações de PLN. Sendo assim, os algoritmos utilizados no processo de tokenização devem levar em consideração as características da linguagem a ser analisada bem como o objetivo da aplicação.

A Figura 5 mostra uma parte da coleção de notícias após a realização do processo de tokenização.

FIGURA 5: Recorte do corpus após a tokenização



Fonte: elaborada pela autora

O texto é uma sequência linear de caracteres ou palavras ou frases. Sendo assim, ele precisa ser segmentado em unidades linguísticas como palavras, pontuação, números, alfanuméricos, etc. Esse procedimento, conhecido como tokenização, pode ser considerado um pré-processamento, visto que faz a identificação das unidades básicas a serem processadas. E a identificação dessas unidades é extremamente importante para que os próximos passados do pré-processamento possam ser executados, pois os *tokens* necessitam ser normalizados para se obter dados textuais limpos e padronizados que são mais fáceis de entender, interpretar e usar em Processamento de Linguagem Natural e em Aprendizado de Máquina.

2.2.1.1.2 Remoção das stop words

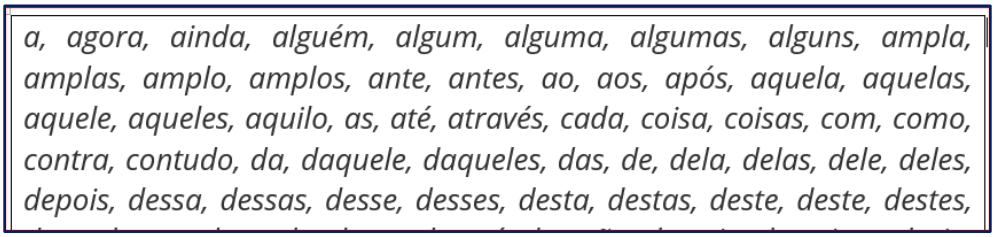
Stop words são palavras irrelevantes e insignificantes que aparecem em uma linguagem para ajudar a construir sentenças, mas que não representam nenhum conteúdo nos documentos (LIU, 2007, p.199). São palavras comuns, que repetem muitas vezes num texto e não acrescentam e nem retiram informações relevantes (SOLKA, 2007). “São termos considerados irrelevantes e somente desempenham um papel funcional no texto. São palavras que dependem do idioma e podem também depender do tópico de interesse.” (ALVES, 2010, p. 10). No ponto de vista dos autores Baeza-Yates e Ribeiro-Neto (2013), as palavras que são muito frequentes entre os documentos de uma coleção não são boas como discriminantes. Nesse sentido, a remoção de *stop word* ou remoção de *stop list* é a retirada

de palavras que, normalmente, são artigos, conjunções e preposições que aparecem no corpus.

Apesar de aparecerem com muita frequência em documentos, as *stop words* não são essenciais para dar sentido no texto, pois são usadas apenas para juntar palavras em uma frase. Devido à sua alta frequência de ocorrência, sua presença apresenta, muitas vezes, como um obstáculo na análise de textos, por isso, a necessidade de removê-las.

A Figura 6 ilustra uma parte do conjunto de *stop words* utilizado neste trabalho.

FIGURA 6: Recorte da lista de *stop words* utilizada neste trabalho



Fonte: Elaborada pela autora

A biblioteca NLTK (*Natural Language Toolkit*) do Python já possui uma lista pré-determinada de palavras, nos diferentes idiomas, que são consideradas *stop words*. No entanto, a partir da análise das palavras mais comuns no conjunto de notícias, é possível criar uma *stop list* mais adequada ao contexto. Assim, nesta pesquisa, além da lista de *stop words* para o idioma português, disponibilizada pelo NLTK, novas palavras foram adicionadas como o objetivo de melhorar o resultado da clusterização de textos.

2.2.1.1.3 Stemming

Para melhor compreensão da técnica de *stemming*, é necessário entender a estrutura das palavras. De acordo com Lucca e Nunes (2002), os elementos mórficos ou morfemas dividem-se em raiz, radical ou tema, vogal temática, afixos e desinências. A raiz é o elemento mórfico mais simples em que uma palavra pode ser reduzida. Obtém-se a raiz pela eliminação dos elementos secundários de formação. O Quadro 3 apresenta dois exemplos em que as raízes são extraídas das palavras:

QUADRO 3: Exemplo de raiz

Palavra	Raiz
Abdicar	Dic
Abnegar	Neg

Fonte: elaborado pela autora

Quanto ao radical, esse é considerado o elemento mórfico que fornece a significação da palavra. “Pode, não obstante, coincidir que o mesmo elemento venha a ser raiz e radical ao mesmo tempo.” (LUCCA; NUNES, 2009, p. 6). O Quadro 4 mostra os radicais das palavras “abdicar” e “abnegar”.

QUADRO 4: Exemplo de radical

Palavra	Radical
Abdicar	Abdic
Abnegar	Abneg

Fonte: Elaborado pela autora

Segundo Lucca e Nunes (2002), o Tema é constituído pelo radical mais uma vogal temática e, muitas vezes, coincide com o radical. Normalmente, as duas palavras, radical e tema, são consideradas sinônimas. Já a Vogal Temática “é o elemento mórfico que se agrega ao radical de uma palavra para que ela possa receber outros morfemas. Divide-se em nominais e verbais”. As nominais referem-se a um substantivo ou adjetivo e as verbais a um verbo. (LUCCA; NUNES, 2009, p. 7). Exemplos:

QUADRO 5: Exemplos de vogais temáticas

Palavra	Vogal temática
Rosa	A
Livro	O
Cantar	A
Beber	E
Cair	I

Fonte: Elaborado pela autora

Lucca e Nunes (2002) definem afixos como elementos mórficos que se agregam a uma raiz ou radical a fim de mudar o sentido de uma palavra. Os afixos subdividem-se em sufixos (pospostos ao radical) e prefixos (antepostos ao radical). E quanto à desinência, os autores a conceitua como o elemento mórfico que indica as flexões da palavra, podendo ser nominal (indica gênero e número) ou verbal (indica tempo, modo, número, pessoa e as formas nominais do verbo).

Diante do exposto, fica mais fácil compreender a técnica de *stemming*, pois esse processo é fortemente dependente da linguagem da informação textual. Em muitas línguas, uma palavra tem várias formas sintáticas, dependendo do contexto em que ela é usada. Na língua portuguesa, por exemplo, os verbos apresentam flexões de acordo a pessoa, a fim de

evidenciar quem fala e para quem se fala. Variam também no tempo para referenciar o momento em que se fala e no modo, que são as várias formas em que a ação do verbo pode expressar. Essas transformações são consideradas variações sintáticas, que têm como origem a mesma raiz. Porém, isso prejudica o resultado da busca dos sistemas de Recuperação da Informação, visto que um documento relevante pode conter uma variação da palavra usada na consulta, mas não o termo exato em si. No processo de *clustering,* as variações das palavras também prejudicam o resultado, entretanto, esse problema pode ser resolvido por meio da técnica de *stemming*.

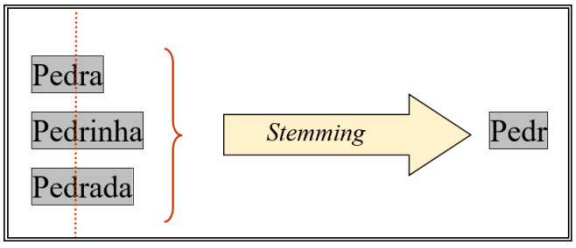
Segundo Morais e Ambrósio (2007), *stemming* é uma técnica de normalização linguística, na qual as variantes de um termo são reduzidas a uma forma comum denominada *stem* (radical). Isso resulta na eliminação de prefixos, sufixos e características de gênero, número e grau das palavras, reduzindo o número de atributos em até 50%. Na visão de Monteiro e Gomes *et al.* (2006), s*temming* consiste na remoção de variações de palavras, tais como plural, gerúndio, afixos, gênero e número, de modo que a palavra fique somente com o *stem, ou seja, com o radical*. Isto é, "concentra-se na redução de cada palavra do léxico até que seja obtida sua respectiva raiz e tem como principal benefício à eliminação de sufixos que indicam variação na forma da palavra, como plural e tempos verbais." (CARRILHO JUNIOR, 2007, p.40).

"O *stem* é a porção de uma palavra que resta após a remoção de afixos. Um exemplo típico é a palavra *connect*, a qual é o *stem* para as variantes *connected*, *connecting*, *connection* e *connections."* (BAEZA-YATES; RIBEIRO-NETO, 2013, p. 214). "O *stem* resultante não precisa ser uma palavra válida do idioma, porém deve conter o significado base original de suas palavras." (FLORES, 2009, p. 10).

Na visão de Baeza-Yates e Ribeiro-Neto (2013), a técnica de *stemming* é útil na melhoria do desempenho da análise de textos, porque reduz as variantes das mesmas palavras para um conceito comum. Além disso, tem o efeito secundário de reduzir o tamanho da estrutura do corpus, porque o número de termos distintos torna-se reduzido.

A Figura 7 ilustra o processo de *stemming.*

FIGURA 7: *Stemming*



Fonte: Elaborada pela autora

Para Madeira (2015), os algoritmos de *stemming* são extremamente dependentes da língua para o qual foram escritos. Deste modo, para textos em português, é imprescindível utilizar um *stemmer* que tenha sido projetado especialmente para termos nesse idioma. As principais versões são o RSLP (Removedor de Sufixo da Língua Portuguesa), proposto por Orengo e Huyck (2001), o Pegastemming (Gonzalez *et al*., 2003) e o algoritmo de Porter (Porter, 2005). A biblioteca NLTK inclui o RSLP Stemmer, que é um dos algoritmos de *stemming* concebidos para a língua Portuguesa, o Porter, que é bem popular, e o Snowball, que é uma versão do Porter que possui módulos em várias línguas. Desta forma, o RSLP, o Porter e o Snowball foram os algoritmos utilizados e testados nos experimentos desta tese.

Xavier, Silva e Gomes (2013, p. 86) definem o algoritmo Porter Stemmer em cinco etapas:

Etapa 1: Remoção de sufixos comuns (por exemplo: eza, ismos, ável, ível, oso, ações, mente, ânsia...);

Etapa 2: remoção de sufixos verbais, caso a palavra não tenha sido alterada na primeira etapa (por exemplo: ada, ida, aria, ará, ava, isse, iriam, aram, endo, indo, arão, íreis, êssemos...);

Etapa 3: remoção do sufixo “i”, se precedido de “c” e se a palavra foi alterada pelas etapas 1 ou 2;

Etapa 4: remoção de sufixos residuais (os, a, i, o, á, í, ó) se nenhuma das etapas anteriores alterou a palavra;

Etapa 5: remoção dos sufixos “e”, “é” e “ê”, tratamento do cedilha e das sílabas “gue”, “gué” e “guê”.

Quanto ao algoritmo *RSLP Stemmer*, Xavier, Silva e Gomes (2013, p. 86) afirmam que ele foi projetado especialmente para a língua portuguesa, com objetivo de ser simples e, ao mesmo tempo, eficaz. “O diferencial desse algoritmo é que além de analisar diversas regras específicas do idioma português, ainda conta com um dicionário de exceções. Esse radicalizador é composto por oito etapas, cada fase contendo um conjunto de regras, sendo que apenas uma regra em cada passo pode ser aplicada.” (XAVIER; SILVA; GOMES, 2013, p. 86).

Etapa 1: remoção do plural das palavras (normalmente “s”). Possui 11 regras definidas;

Etapa 2: transformação do gênero da palavra de feminino para masculino. São aplicadas 15 regras;

Etapa 3: redução adverbial, possui somente um sufixo (“mente”);

Etapa 4: reduz o grau das palavras de aumentativo, superlativo ou diminutivo para normal. Aplicação de 23 regras;

Etapa 5: redução nominal, originalmente eram definidos 61 sufixos para substantivos e adjetivos. Posteriormente, eles foram expandidos para 84. Caso a palavra seja reduzida nesta etapa, os passos 6 e 7 não são executados;

Etapa 6: redução verbal, são definidas 101 regras para mapear os verbos regulares em suas mais de 50 formas. Após a execução desse passo, o termo é reduzido a sua raiz, indo direto para a execução do passo 8;

Etapa 7: remoção da última vogal (“a”, “e” ou “o”) de palavras que não foram reduzidas pelas etapas 5 e 6;

Etapa 8: substitui todos os caracteres acentuados por seus equivalentes sem acentos.

Segundo Stein e Potthast (2007), os algoritmos de *stemming* podem apresentar dois tipos de erros: *overstemming* e *understemming*. O *overstemming* acontece quando são removidas mais letras do que o necessário, fazendo com que palavras com sentidos diferentes se reduzem ao mesmo *stem*. Por exemplo, o *stem* **comp** para as palavras **computador** e **comparar**. Já o *understemming* ocorre quando letras são deixadas a mais, surgindo *stems* diferentes para palavras com mesmo radical. Por exemplo, os *stems* **biolo** e **biolog**, respectivamente, para as palavras **biologia** e **biologista** (ALVARES, 2014, p. 1).

Diante do exposto, algoritmos de *stemming* serão testados nos experimentos desta pesquisa com a finalidade de verificar o desempenho deles com uma coleção de notícias no idioma português.

2.2.1.1.4 Lematização

O processo de lematização é muito parecido com o *stemming*, pois os afixos também são removidos, porém, obtém-se o lema da palavra e não o radical. A diferença é que o radical nem sempre é uma palavra lexicograficamente correta; isto é, pode não estar presente no dicionário. Já o lema, sempre estará presente no dicionário (SARKAR, 2016).

Para Lucca e Nunes (2002, p. 9), a lematização é o ato de representar as palavras através do infinitivo dos verbos e masculino singular dos substantivos e adjetivos.

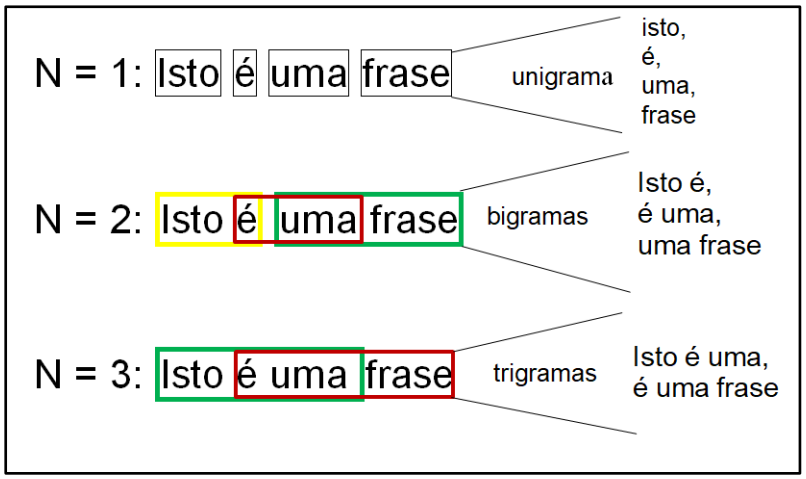
Na visão de Sarkar (2016), o processo de lematização é consideravelmente mais lento do que o *stemming* porque envolve um passo adicional, ou seja, haverá a remoção do afixo da palavra se, e somente se, o lema estiver presente no dicionário.

O pacote NLTK tem um módulo de lematização robusto que usa o *WordNet*, uma grande rede de conceitos interligados, para obter a raiz ou o lema. Segundo Santana (2017), o *WordNet* é um grande banco de dados léxico de substantivos, verbos, adjetivos e advérbios que são agrupados em conjuntos de sinônimos, cada um expressando um conceito distinto, interligados pelo seu significado semântico conceitual e suas relações léxicas.

2.2.1.1.5 N-grama

Após a execução da limpeza, o passo seguinte é a geração de n-grama. Essa etapa é responsável por gerar atributos com palavras simples (*n=1*) ou compostas, como "Aprendizado de Máquina" (*n=2*), para que um ou mais *tokens* que aparecem sequencialmente nos documentos sejam unidos. O valor de n, especificado pelo usuário, pode ser qualquer valor inteiro maior ou igual a um (SOARES; PRATI; MONARD, 2009). A Figura 8 apresenta exemplos de n-gramas para n=1, n=2 e n=3.

FIGURA 8: Exemplos de n-gramas



Fonte: Elaborada pela autora

Conforme apresentado acima, a técnica n-grama ajuda a identificar a ordem em que as palavras aparecem em uma frase. Assim, podem-se computar, por exemplo, os bigramas que ocorrem com mais frequência em um corpus. Segundo Tomović; Janičić *et al.* (2006 citado por CORREIA, 2012, p. 17), a técnica de n-gramas tem sido aplicada a um vasto número de casos de estudo e de domínios diferentes tais como, compressão de texto, correção ortográfica, reconhecimento ótico de caracteres (OCR), categorização de texto automatizado entre vários outros, tendo apresentado bons resultados com a sua utilização.

Posteriormente às etapas de limpeza e redução dos dados, os documentos são transformados em uma forma numérica, conforme descrito a seguir.

2.2.1.1.6 Cálculo de relevância de palavras

De acordo com Hack *et al*. (2013), nem todas as palavras presentes em um documento possuem o mesmo valor de relevância. Os termos que são utilizados mais frequentemente, com exceção das *stop words*, geralmente, têm significado mais importante. Deste modo, "a

ideia do cálculo da relevância de uma palavra dentro de um documento objetiva obter um peso referente ao uso do termo dentro do texto." (HACK *et al*. 2013, p. 3).

É nesta fase do pré-processamento que os textos são transformados e representados em um formato numérico para que os algoritmos de Aprendizado de Máquina possam compreender as informações contidas nos documentos. Normalmente, o texto é visualizado como uma sequência de palavras ou pode-se impor uma estrutura adicional como a sintaxe. Além disso, ele pode ser dissolvido em alguma estrutura, como por exemplo, dividido em *tokens*. Para isso, o conteúdo de cada documento é decomposto em termos e, em seguida, é verificada a frequência de cada palavra. Geralmente, o processo é aplicado em um conjunto de textos e o corpus é transformado em uma matriz atributo-valor, na qual cada linha representa um documento do conjunto e cada documento é descrito pelos valores dos atributos mais representativos da coleção (SOARES; PRATI; MONARD, 2009).

Segundo Hack *et al*. (2013), existem várias fórmulas para cálculo do peso. As mais comuns são baseadas em cálculos simples de frequência: frequência absoluta, frequência relativa, frequência inversa de documentos.

a) **Frequência Absoluta**

Conhecida por frequência do termo ou *Term Frequency* (TF), essa técnica consiste em contar a quantidade de aparições de um mesmo n-grama dentro de um documento. Assim, se denotarmos como tf_{ij} o número de vezes que o termo *j* aparece no documento *i*, o peso *w* da TF daquele termo em um documento seria simplesmente: $w_{ij}= tf_{ij}$

> TF representa a medida da quantidade de vezes que um termo aparece em um documento. Essa é a medida de peso mais simples que existe, mas não é aconselhada em alguns casos, porque, em análise de coleções de documentos, não é capaz de fazer distinção entre os termos que aparecem em poucos ou em muitos documentos. Este tipo de análise também não leva em conta a quantidade de palavras existentes em um documento. Com isso, uma palavra pouco frequente em um documento pequeno pode ter a mesma importância de uma palavra muito frequente de um documento grande. (HACK et al. 2013, p. 4).

A Tabela 1 ilustra as frequências das palavras (TF) de dois documentos, conforme exemplificado a seguir:

Doc 1 *– Gatos pretos dão mais sorte que gatos brancos.*

Doc 2 *– Meus gatos dão muito trabalho*

TABELA 1: Matriz TF dos documentos

	Gatos	Pretos	Dão	Mais	Sorte	que	brancos	Meus	muito	trabalho
Doc 1	2	1	1	1	1	1	1	0	0	0
Doc 2	1	0	1	0	0	0	0	1	1	1

Fonte: Adaptada de BARBOSA (2017)

O resultado é uma matriz de frequência, conhecida como *Bag-of-Word* (saco de palavras), que é a representação numérica do corpus. Nesse modelo, a ordem e a sequência das palavras não são levadas em conta e "os termos são considerados independentes, formando um conjunto desordenado em que a ordem de ocorrência das palavras não importa." (NOGUEIRA, 2009, p. 16).

O *Bag-of-Words* (BoW) é um nome mais elegante para as matrizes de frequência. É uma forma de representação de texto que computa a ocorrência das palavras em um documento. Segundo Baeza-Yates e Ribeiro-Neto (2013), a presença de um termo em um documento estabelece uma afinidade entre eles. Essas relações podem ser quantificadas, por exemplo, pela frequência do termo no documento. No modelo matricial, isso pode ser representado da seguinte forma:

$$\begin{array}{c} \\ k_1 \\ k_2 \\ k_3 \end{array} \begin{array}{c} \begin{array}{cc} d_1 & d_2 \end{array} \\ \begin{bmatrix} f_{1,1} & f_{1,2} \\ f_{2,1} & f_{2,2} \\ f_{3,1} & f_{3,2} \end{bmatrix} \end{array}$$

"Onde cada elemento $f_{i,j}$ representa a frequência do termo k_i no documento d_j. Usar a frequência de ocorrências para quantificar a relação entre termos e documentos fornece mais informação do que simplesmente registrar se o termo ocorre ou não no documento." (BAEZA-YATES; RIBEIRO-NETO, 2013, p. 28)

Nessa abordagem, cada texto é representado como um vetor de palavras que ocorrem no documento ou em representações mais sofisticadas, como frases ou sentenças (MATSUBARA, 2004). É uma abordagem muito simples e flexível, onde o modelo está preocupado apenas com o fato de palavras conhecidas ocorrerem ou não no documento. No Quadro 6, w_i representa uma palavra, d_j representa um documento e a_{ij} o peso de cada palavra no documento.

QUADRO 6: Modelo de BoW

	w_1	w_1	w_1	...	w_n
d_1	a_{11}	a_{12}	a_{13}	...	a_{1n}
d_1	a_{21}	a_{22}	a_{23}	...	a_{2n}
...	⋮	...	...	⋱	...
d_n	a_{n1}	a_{n2}	a_{n3}	...	a_{nn}

Fonte: (ALVES, 2019, p. 29)

Na forma matricial, cada elemento $\boldsymbol{a_{i,j}}$ representa a frequência do termo $\boldsymbol{d_i}$ no documento $\boldsymbol{w_j}$.

Para Alves (2010, p. 30), existem várias medidas para calcular os valores dos pesos de a_{ij}. Elas podem ser classificadas em dois tipos: binárias e baseadas em frequências. Os

pesos binários indicam a ocorrência ou não de um dado termo num determinado documento (1 – verdadeiro e 0 – falso), podendo ser utilizados, por exemplo, para extrair informações relativas à semelhança de documentos a partir do número de termos em comum. A Tabela 2 ilustra o *Bag-of-Word* gerada a partir de dois documentos.

TABELA 2: Matriz de incidência binária termo-documento

Documento 1

A esperta raposa marrom saltou sobre as costas do cachorro preguiçoso.

Documento 2

Agora é hora de todos os homens bons virem em auxílio de seu grupo.

Termo	Documento 1	Documento 2
agora	0	1
Auxilio	0	1
Bons	0	1
cachorro	1	0
Costas	1	0
esperta	1	0
Grupo	0	1
Homens	0	1
Hora	0	1
Marrom	1	0
preguiçoso	1	0
Raposa	1	0
Sobre	1	0
Soltou	1	0
Todos	0	1
Virem	0	1

Stop words
a
as
de
do
é
em
os
seu

Fonte: Elaborada pela autora

Quanto às medidas baseadas em frequências, segundo Nogueira (2009, p. 17), elas "visam contabilizar o número de ocorrências de um determinado termo em um dado documento, servindo como base para a extração de diversas medidas estatísticas na extração de padrões.". Ainda assim, "é uma abordagem simplista." (BAEZA-YATES; RIBEIRO-NETO, 2011, p. 28).

Apesar de ser uma representação simples, a abordagem *Bag-of-Words* normalmente obtém resultados experimentais melhores que representações mais sofisticadas (APTÉ; DAMERAU; WEISS, 1994, DUMAIS; PLATT; HECKERMAN; SAHAMI, 1998, LEWIS, 1992).

b) **Frequência Relativa**

Segundo Hack *et al*. (2013), o cálculo da Frequência Relativa leva em conta o tamanho do documento (quantidade de palavras que ele possui) e normaliza os pesos de acordo com essa informação.

O Valor da Frel é dado por:

$$F_{rel}(x) = \frac{F_{abs}(x)}{N} \quad (1)$$

De modo que frequência relativa (Frel) de uma palavra **x** em um documento qualquer é calculada dividindo-se sua frequência absoluta (Fabs) pelo número total de palavras no mesmo documento (N).

c) **Frequência Inversa de Documentos**

Para Hack *et al*. (2013), a Frequência Inversa de Documentos busca normalizar termos frequentes com base na sua ocorrência em todos os documentos analisados. O cálculo da Frequência Inversa de Documentos, em inglês *Inverse Document Frequency* (IDF), é realizado com base na informação da frequência absoluta do termo no documento e da frequência do termo em todos os documentos. Isso é capaz de aumentar a importância dos termos que aparecem em poucos documentos e diminuir a importância de termos que aparecem em muitos, justamente pelo fato dos termos de baixa frequência serem, em geral, mais discriminantes.

A fórmula mais comum utilizada para cálculo do peso de um termo utilizando a frequência inversa, segundo Moraes e Ambrósio (2003, p. 18) é:

$$Peso_{td}(x) = \frac{Freq_{td}}{DocFreq_{td}} \quad (2)$$

Onde,

$Peso_{td}$: é o grau de relação entre o termo *t* e o documento *d*;

$Freq_{td}$: número de vezes que o termo *t* aparece no documento *d*;

$DocFreq_{td}$: número de documentos que o termo *t* aparece.

d) **Frequência do Termo-Inverso da Frequência nos Documentos**

O TF-IDF (*Term-Frequency-Inverse Document Frequency)* é uma medida estatística usada para avaliar o quão importante uma palavra é para um documento em relação a uma coleção de documentos. Esse modelo é composto pela multiplicação dos termos TF (*Term Frequency*) e IDF. O TF mede com que frequência um termo ocorre em um documento e o IDF mensura quanto um termo é importante. Segundo Liu *et al*. (2005), o TF-IDF é o esquema de ponderação mais vastamente empregado.

Segundo Baeza-Yates e Ribeiro-Neto (2011), a primeira forma de ponderação da frequência dos termos foi proposta Luhn em 1957 e baseia-se na seguinte suposição: “O valor de peso de um termo k_i que ocorre em um documento d_j é simplesmente proporcional à frequência do termo $f_{i,j}$. Isto é, quanto mais frequentemente um termo k_i ocorrer no texto do

documento d_j maior será a sua frequência de termo $TF_{i,j}$". De acordo com esses autores, essa hipótese baseia-se na observação que termos com alta frequência são importantes para descrever os tópicos-chave de um documento. Já a *Term Frequency–Inverse Document Frequency* (TF-IDF)

> é um cálculo que mostra o valor que cada palavra tem em cada texto. A fórmula para calcular o TF-IDF é a frequência da palavra num texto vezes o inverso da frequência dessa palavra em todos os textos. Desta forma, uma palavra que aparece muitas vezes num texto não terá tanto valor se aparecer igualmente muitas vezes nos outros textos. Este método ajuda a distinguir palavras relevantes para o texto de palavras que são comuns. (RODRIGUES, 2016, p. 21).

Assim, para o Cálculo TF-IDF, utiliza-se a seguinte fórmula:

$$tfidf(w) = tf.\log(\frac{N}{df(w)}) \quad (3)$$

Onde:

- tf (w) – Frequência do termo (número de ocorrências de palavras em um documento)
- df (w) – frequência do documento (número de documentos contendo a palavra)
- N – número de todos os documentos
- tfidf (w) – importância relativa da palavra no documento

e) A Tabela 3 ilustra o cálculo da Frequência do Termo–Inverso da Frequência nos Documentos (TF-IDF) de dois textos, conforme exemplificado a seguir:

Doc 1 *– Gatos pretos dão mais sorte que gatos brancos.*

Doc 2 *– Meus gatos dão muito trabalho*

TABELA 3: Matriz TF-IDF

	Gatos	pretos	dão	mais	Sorte	que	brancos	meus	muito	trabalho
Doc 1	0	0.3	0.3	0.3	0.3	0.3	0.3	0	0	0
Doc 2	0	0	0.3	0	0	0	0	0.3	0.3	0.3

Fonte: Adaptada de BARBOSA (2017)

Na seleção dos atributos, os termos mais relevantes do conjunto são escolhidos. Para isso, utiliza-se a *Document Frequency* (DF). A DF procura eliminar palavras muito frequentes por possuírem pouco poder de discriminação e também as palavras pouco frequentes, por não contribuírem para o cálculo de similaridade entre os documentos. O resultado será uma matriz com os termos selecionados e seus respectivos pesos. Essa representação simplificada também é conhecida como Modelo de Espaço Vetorial.

2.2.1.1.7 Identificação de características

Um dos maiores desafios do processo de Mineração de Textos é a alta dimensionalidade dos dados. Uma pequena coleção de textos pode facilmente conter milhares de termos, muitos deles redundantes e desnecessários, que tornam lento o processo de extração de conhecimento, além de prejudicar a qualidade dos resultados (REZENDE; MARCACINI; MOURA, 2011). Segundo Mitchell (1997), quando se utiliza um corpus de textos ou de notícias extraídas da internet, é natural que se tenha dados de alta dimensionalidade. Sendo assim, para aumentar a precisão do algoritmo na etapa de mineração, faz-se necessário selecionar os termos mais representativos do conjunto e, com isso, ter um ganho informacional.

Para Gonçalves (2002), a identificação de características (*features)* tem como objetivo formar um dicionário que contenha os atributos que serão utilizados para representar os documentos individuais dentro de uma coleção. Independentemente do algoritmo e do tipo de dados que se pretende categorizar, essa etapa é importante no processo de categorização e "tem como objetivo primordial a obtenção dos termos mais relevantes para a distinção, e ainda para construir uma representação dos mesmos de forma estruturada." (CORREIA, 2012, p. 14). Segundo esse autor, a extração dos atributos mais representativos é de extrema importância, já que as características irrelevantes minimizam o desempenho dos algoritmos e o alto volume de dados aumenta a complexidade dos mesmos. Deste modo, o ideal é obter o menor número de atributos que seja o mais discriminante possível das informações que se pretendem categorizar (CORREIA, 2012).

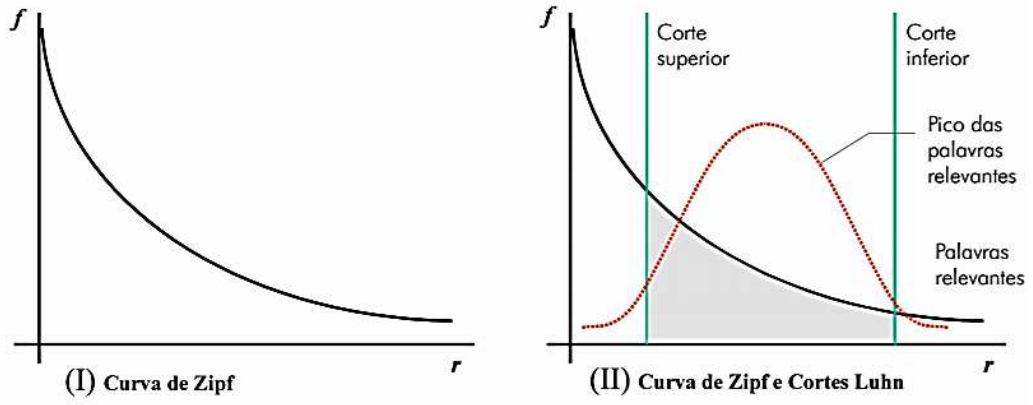
Nesse contexto, faz-se necessária a utilização de algumas técnicas de *Text Mining* para a seleção das características mais importantes. O primeiro passo é eliminar as *stop words*, pois isso reduz significativamente a quantidade de termos e diminui o custo computacional das próximas etapas (MANNING; RAGHAVAN; SCHÜTZE, 2008). Posteriormente, procuram-se identificar as variações morfológicas das palavras para reduzir os termos à sua raiz através da técnica de *stemming*. Em seguida, podem-se usar dicionários ou *thesaurus* para descobrir termos sinônimos. Além disso, é possível buscar na coleção a formação de termos compostos, ou n-gramas, que são termos formados por mais de um elemento, porém com um único significado semântico (MANNING; RAGHAVAN; SCHÜTZE, 2008). Neste trabalho, utilizou-se o conceito de bigramas, ou seja, n=2, para que os vetores de características incluam palavras individuais e combinações de duas palavras subsequentes.

Outra forma de realizar a seleção das características é avaliá-las por medidas estatísticas simples, como Frequência de Termo e Frequência de Documentos (REZENDE; MARCACINI; MOURA, 2011). Luhn (1958) propôs um método baseado na Lei de Zipf (1932),

conhecido também como Princípio do Menor Esforço, para a seleção de palavras utilizando a Frequência de Termos. Segundo Matsubara, Martins e Monard (2003), em textos, ao contabilizar a TF e ordenar o histograma resultante em ordem decrescente, forma-se a chamada Curva de Zipf, na qual o k-ésimo termo mais comum ocorre com frequência inversamente proporcional a k. Luhn (1958) usou a lei de Zipf como uma hipótese nula para especificar dois pontos de corte, denominados de superior e inferior, para excluir termos não relevantes. Os termos que excedem o corte superior são os mais frequentes e são considerados comuns por aparecer em qualquer tipo de documento, como por exemplo, as *stop words*. Já os termos abaixo do corte inferior são considerados raros e, portanto, não contribuem significativamente na discriminação dos documentos (MATSUBARA; MARTIN; MONARD, 2003).

A Figura 9 apresenta a curva da Lei de Zipf (I) e os cortes de Luhn aplicados a Lei de Zipf (II). O eixo cartesiano *f* representa a Frequência dos Termos e o eixo cartesiano *r* os termos correspondentes ordenados por frequência, do maior para a menor.

FIGURA 9: A curva de Zipf e os cortes de Luhn



Fonte: MATSUBARA; MARTINS; MONARD (2003, p. 14)

Assim, “são traçados pontos de corte superior e inferior da Curva de Zipf, de maneira que termos com alta e baixa frequência são descartados, considerando os termos mais significativos os de frequência intermediária.” (REZENDE; MARCACINI; MOURA, 2011, p. 9). Nesse sentido, para a seleção das *features*, pode-se utilizar o cálculo da Frequência do Termo no texto para definir um limite mínimo e máximo que um atributo deve aparecer para ser considerado válido, uma vez que tanto uma característica que aparece poucas vezes como uma que ocorre muitas vezes poderão não ter grande capacidade discriminativa.

Posto que os termos mais representativos do corpus já foram selecionados, deve-se buscar a estruturação dos documentos de maneira a torná-los processáveis por algoritmos.

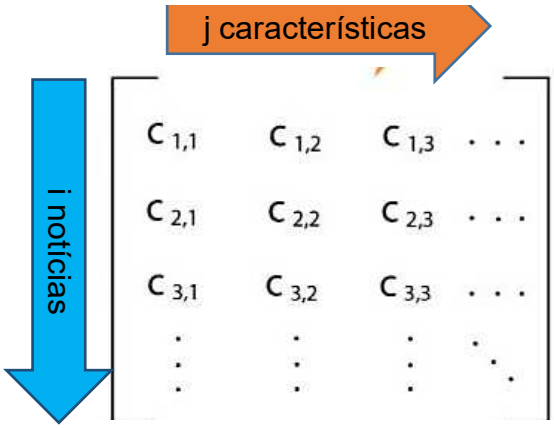
Rezende, Marcacini e Moura (2011) afirmam que o modelo mais utilizado para representação de dados textuais é o Modelo Espaço-Vetorial que será descrito na próxima seção.

2.2.1.1.8 Representação das features ou Vetorização de Textos

Segundo Correia (2012), independentemente das características extraídas, é sempre necessário representá-las numa estrutura organizada. Benffort, Bilbro e Ojeda (2018) relatam que os algoritmos de Aprendizagem de Máquina atuam em um espaço de atributos numéricos e esperam como entrada uma matriz bidimensional, em que as linhas são as instâncias e as colunas os atributos. Na análise de textos, as instâncias são documentos inteiros que podem variar de tamanho, podendo ser citações, artigos, *tweets* ou livros. Para Benffort, Bilbro e Ojeda (2018), todos os textos de uma coleção podem ser representados como vetor cujo comprimento é igual ao vocabulário do corpus.

Portanto, para aplicar as técnicas de *Machine Learning* em texto, é necessário transformar os documentos em uma representação vetorial, processo conhecido como *feature extraction* ou *vetorization*. A vetorização consiste na representação de um corpus na forma de um vetor de termos. A forma mais simplificada de representar um vetor de características é a *Bag-of-Words*, ilustrada na Figura 10, onde cada elemento $C_{i,j}$ representa a frequência do termo (característica) em cada documento (notícias).

FIGURA 10:Estrutura de um vetor de caraterísticas



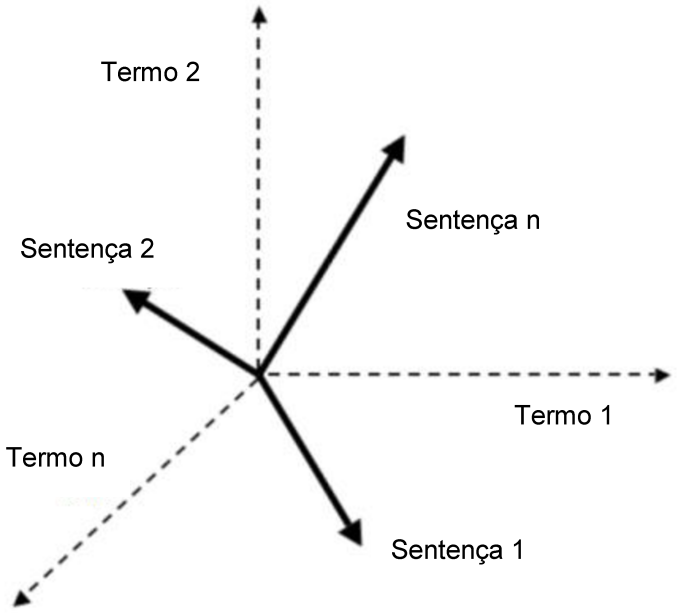
Fonte: CORREIA (2012, p. 18)

Para Hack *et.al* (2013), a maneira mais comum de vetorização de textos é associar cada termo com uma frequência, onde cada documento é representado por um vetor de termos e cada termo possui um valor associado que indica o seu grau de importância (denominado peso) no documento. Assim sendo, cada documento possui um vetor associado que é constituído por pares de elementos na forma (palavra 1, peso 1), (palavra 2, peso 2) ...

(palavra n, peso n). Nesse vetor, são representadas todas as palavras da coleção de textos e não somente aquelas presentes no documento. Os termos que o documento não contém recebem grau de importância zero e os outros são calculados através de uma fórmula de identificação de importância. Isso faz com que os pesos próximos de um (1) indicam termos extremamente importantes e os próximos de zero (0) caracterizam termos completamente irrelevantes (em alguns casos a faixa pode variar entre -1 e 1) (HACK *et.al*, 2013).

Essa representação, conhecida como Modelo de Espaço Vetorial (Salton *et al.*, 1975), em inglês *Vector Space Model* (VSM), é a versão aprimorada do BoW, na qual cada documento de texto é representado como um vetor e cada dimensão corresponde a um termo separado (palavra). Se um termo ocorrer no documento, seu valor se tornará diferente de zero no vetor. Para Ebecken, Lopes e Costa (2003), o Modelo de Espaço Vetorial é uma representação numérica direta do BoW. A Figura 11 ilustra esse Modelo.

FIGURA 11: Modelo Espaço Vetorial



Fonte: Elaborada pela autora

Salton (1989 citado por Ebecken; Lopes e Costa, 2003, p. 345) afirma que o Modelo de Espaço Vetorial tem a função de resolver problemas de representação de documentos utilizando representação geométrica. Assim, “os documentos são representados como pontos (ou vetores) em um espaço Euclidiano t-dimensional em que cada dimensão corresponde a uma palavra (termo) do vocabulário”. Nesse contexto,

> o *i-ésimo* componente do vetor documento expressa o número de vezes que a palavra com o índice *i* ocorre em um documento e cada palavra pode ter um peso associado para descrever sua significância. A similaridade entre dois

> documentos é definida como a distância entre os pontos ou como ângulo entre os vetores (desconsiderando o comprimento do documento). Apesar de sua simplicidade, o Modelo do Espaço Vetorial e suas variantes são frequentemente a forma mais comum de representar documentos textuais na mineração de coleções de documentos. Uma explicação para isso é que operações de vetores podem ser executadas muito rapidamente e existem algoritmos padronizados eficientes para realizar a seleção do modelo, a redução da dimensão e a visualização de espaços de vetores. (EBECKEN; LOPES E COSTA, 2003, p. 345).

Por conseguinte, para aplicar a técnica de agrupamento nos experimentos desta pesquisa, optou-se pelo Modelo Espaço Vetorial que, embora tenha sido proposto para indexação de palavras para o uso em Recuperação da Informação (ALMEIDA, 2007), é a representação de uso mais comum quando se pretende agrupar textos (LIU *et al*. 2005).

Por fim, todas as tarefas realizadas durante o pré-processamento são necessárias para reduzir a dimensionalidade dos textos. Para Pontes e Anchieta (*on-line*), a redução da dimensionalidade tem como objetivo diminuir a complexidade do problema em relação à análise de textos, visto que cada palavra presente na coleção de documentos textuais representa uma propriedade. Sendo assim, uma coleção de texto tem uma quantidade imensa de atributos. O efeito de um número grande de atributo é descrito pelo problema da maldição da dimensionalidade (FACELI *et al*., 2017). Segundo Nassif (2011), essa expressão foi assinalada por Bellman (1961) para descrever o fato que mais dimensões resultam em mais combinações e inviabilizam uma abordagem de completa enumeração das possibilidades, pois a tabularização e a visualização dos dados tornam-se crescentemente difíceis ou mesmo inviáveis, além de gerar um alto custo computacional, tornando a execução dos algoritmos muito lenta e até inexequível em vários casos. Dessa forma, para que dados com um elevado número de atributos sejam utilizados, a quantidade de características precisa, necessariamente, ser reduzida. Assim, “uma forma de minimizar o impacto do problema da dimensionalidade é combinar ou eliminar parte dos atributos irrelevantes.” (FACELI *et al*., 2017, p. 46).

Nesse contexto, para resolver o problema em questão, algumas técnicas, tais como *stemming*, remoção de *stop words* e extração de atributos podem ser utilizadas para reduzir a dimensão de uma coleção textual. Já a técnica de seleção de atributos busca selecionar os termos mais relevantes do documento. Assim, a redução geralmente é realizada através da seleção das melhores características, de acordo com algum critério. Para isso, utiliza-se a Frequência no Documento para eliminar as palavras muito frequentes e também as palavras pouco frequentes, visto que elas não colaboram no cálculo de similaridade entre os textos. E no processo de extração de *features,* as redundâncias entre os atributos são eliminadas, resultando em um conjunto de novas características menor que o original. (NASSIF, 2011, p. 11).

2.2.1.2 Extração de conhecimento

Este passo tem como objetivo buscar padrões que permitam representar o conhecimento que possa existir, de modo implícito, no conjunto de dados analisado (WITTEN; FRANK, 2005). O processo de análise de textos é dividido em várias etapas, uma delas é a extração de conhecimento. As tarefas realizadas nesta etapa dependem do objetivo final do processo e são divididas em duas categorias: preditivas e descritivas.

Facelli *et al.* (2017) afirmam que nos modelos preditivos, dado um conjunto de exemplos rotulados, o algoritmo de Aprendizado de Máquina preditivo, ou supervisionado, constrói um estimador.

> O rótulo ou etiqueta toma valores num domínio conhecido. Se esse domínio for um conjunto de valores nominais, tem-se um problema de classificação, também conhecido como aprendizado de conceitos, e o estimador gerado é um classificador. Se o domínio for um conjunto infinito e ordenado de valores, tem-se um problema de regressão, que induz um regresso (FACELLI *et al*., 2017, p. 54).

Portanto, nos modelos preditivos têm-se as tarefas de classificação e regressão. Assuntos que serão discutidos com mais detalhes no item 2.3.1.

Em relação aos modelos descritivos, “as tarefas do algoritmo de aprendizado descritivo, ou não supervisionado, se referem à identificação de informações relevantes nos dados sem a presença de um elemento externo para guiar o aprendizado.” (FACELLI *et al*., 2017, p. 178). Segundo esses autores, as tarefas descritivas podem ser divididas em associação, agrupamento e sumarização, temas que serão discutidos no tópico 2.3.2.

2.2.1.3 Visualização, Validação e interpretação

Esta etapa tem como objetivo interpretar e avaliar os padrões extraídos por um especialista do domínio, de modo a verificar se os resultados produzidos são válidos, úteis e se atendem aos objetivos propostos. Nessa perspectiva, “técnicas de visualização dos dados são muito úteis para a etapa de validação e interpretação dos resultados, podendo fornecer uma representação visual dos dados mais intuitiva e facilitando a compreensão dos padrões produzidos.” (NASSIF, 2011, p. 15). Assim, esta pesquisa utilizou a biblioteca *Matplotlib* do Python na criação dos gráficos para, assim, melhor visualizar os *clusters.*

Portanto, as ferramentas automatizadas que apresentam os resultados de uma forma visual, como por exemplo, nuvens de palavras, mapas, diagramas de dispersão e dendogramas, são de grande valia para auxiliar os analistas de domínio nas suas inferências.

Quanto à avaliação, Faceli *et al.* (2017, p. 203) afirmam que é nesta etapa que se avalia o resultado do agrupamento e deve, “de forma objetiva, determinar se os *clusters* são

significativos, ou seja, se a solução é representativa para o conjunto de dados analisado. Além de verificar a validade da solução, pode ajudar, por exemplo, na determinação do número apropriado de *clusters*", visto que, em geral, esse valor não é conhecido previamente.

Para avaliar o resultado do processo de *clustering*, Padilha (2017) apresenta três medidas: As externas, as internas e as relativas.

a) Medidas externas

Podem-se considerar que as medidas externas são supervisionadas e empregam critérios não inerentes aos próprios conjuntos de dados. Isso significa que já se tem algum conhecimento prévio ou especializado sobre o assunto em questão. Em seguida, comparam-se os resultados dos agrupamentos com o conhecimento especificado anteriormente ou pelo conhecimento de um especialista usando determinada medida de qualidade de *clustering*. São exemplos desse tipo de avaliação, o Índice Rand ajustado (*Adjusted Rand*) e o índice de Jaccard.

- **Índice Rand ajustado**

O *Adjusted Rand Index* pode ser visto como um critério absoluto (externo) ou como um padrão referencial que permite o uso de um conjunto de dados de classificação para realizar avaliação não somente de classificadores, mas também de resultados dos agrupamentos. Este índice avalia duas partições rígidas (*hard* ou *crisp*) R e G do mesmo conjunto de dados. A partição de referência R codifica o rótulo das classes, ou seja, ela particiona o conjunto de dados em k classes conhecidas. A partição G, por sua vez, seleciona o conjunto de dados em k categorias (grupos) e é aquela a ser avaliada. As categorias codificadas por G são chamadas de grupos, pelo contexto de algoritmos de agrupamento de dados (VARGAS, 2012).

- **Índice de Jaccard**

O índice de Jaccard foi utilizado em 1908, pelo francês P. Jaccard, em estudos sobre a distribuição de plantas ao longo de gradientes ambientais (LUDWIG; REYNOLDS, 1988 apud Zanzini, 2005). Segundo Zanzini (2005), o Índice de Jaccard constitui um dos indicadores de similaridade mais amplamente empregados em ecologia de comunidades. Compara qualitativamente a semelhança de espécies que existe entre amostras sucessivas retiradas em intervalos espaciais e temporais ou ao longo de um gradiente ambiental. É um coeficiente binário baseado, unicamente, na relação presença-ausência das espécies nas amostras comparadas. Quantitativamente, o índice de Jaccard varia entre 0 (comunidades

totalmente diferentes quanto à composição de espécies) e 1 (comunidades totalmente semelhantes quanto à composição de espécies) (ZANZINI, 2005).

b) Medida interna

Uma outra métrica usada para calcular a qualidade do *clustering* é chamada de medida interna, que não é supervisionada. Isso significa que os critérios são derivados dos dados em si. Nesse caso, avalia-se a qualidade do agrupamento considerando o quão bem os *clusters* estão separados e quão compactos eles são. De acordo com Faceli *et al.* (2017, p. 237), os critérios internos "medem a qualidade de um agrupamento com base apenas nos dados originais (matriz de objetos ou matriz de similaridade) ".

Como exemplos dessa medida, têm-se o Coeficiente de Silhueta**,** Índice Davies-Bouldin e o Índice Dunn. Esta pesquisa usou o Coeficiente de Silhueta para avaliação dos *clusters*, por isso, será apresentada uma descrição mais detalhada sobre ela.

- **Coeficiente da Silhueta**

Através do Coeficiente da Silhueta, gera-se o gráfico da silhueta e mede-se a qualidade do agrupamento. "A medida da silhueta baseia-se na proximidade entre objetos de um *cluster* e na distância dos objetos de um grupo ao *cluster* mais próximo (ROUSSEEUW, 1987 citado por FACELI, 2017, p. 243).

De acordo com Souza (2007, p. 17), essa técnica foi proposta por Reusseeu (1987) "para uso em métodos de obtenção de agrupamentos em partição. A ideia é auxiliar o pesquisador a escolher o número ótimo de grupos e, ao mesmo tempo, permitir que se construa uma representação gráfica do agrupamento encontrado".

> A silhueta é composta por um gráfico ilustrativo de ***C*** contendo um Índice S(i), i=1, ..., n, que reflete a qualidade da alocação da i-ésima unidade amostral ao seu grupo, permitindo uma visualização global da estrutura encontrada. Para obter as silhuetas é necessário conhecer a distribuição *A* nos grupos de ***C*** e a matriz D, com as parecenças necessariamente numa razão de escala. Inicialmente deve-se calcular o índice s(i), i-1, ..., n, para todos os elementos da amostra. (SOUZA, 2007, p. 17).

No caso da parecença ser uma dissimilaridade, e assumindo-se K ≥ 2, o procedimento é o seguinte (SOUZA, 2007):

- Para cada elemento *i* de *A*, calcula-se *a(i)*, a dissimilaridade média de *i* em relação aos indivíduos do mesmo grupo *C* ao qual ele pertence:

$$a(i) = \sum_{j \in C_l} \frac{d(i,j)}{n_l} \tag{4}$$

- Para cada grupo C_k ao qual *i* não pertença, calcula-se *b(i),* sendo *d(i, C_k)* a dissimilaridade média entre i e os elementos de C_k:

$$b(i) = \underset{i \notin C_k}{min}[d(i, C_k)] \tag{5}$$

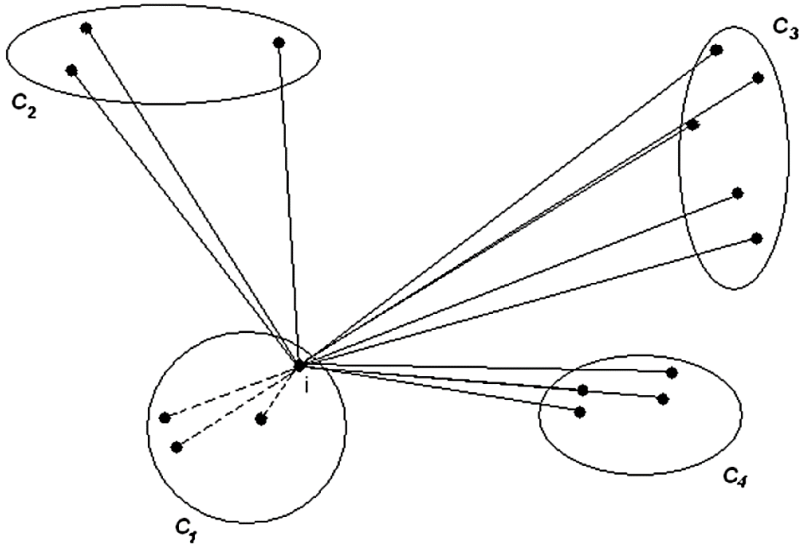
De acordo com Souza (2007, p. 17), "pode-se interpretar *b(i)* como a distância entre *i* e o grupo vizinho mais próximo a ele em termos do critério de dissimilaridade utilizado".

- Os índices *s(i)* são obtidos da seguinte forma:

$$S(i) = \begin{cases} 1 - \frac{a(i)}{b(i)} & se\ a(i) < b(i), \\ 0 & se\ a(i) = b(i), \Leftrightarrow s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \\ \frac{b(i)}{a(i)} - 1 & se\ a(i) > b(i), \end{cases} \tag{6}$$

A Figura 12 mostra um exemplo das distâncias consideradas no cálculo de *s(i)* num agrupamento hipotético com quatro grupos C_1, C_2, C_3, e C_4. Com referência ao *i*-ésimo ponto, as linhas pontilhadas representam as distâncias consideradas para encontrar *a(i)* e as contínuas para *b(i)*. Segundo Souza (2007, p. 17), para os casos onde *i* constitui por si só num grupo de elemento único, torna-se impossível calcular *a(i)*. Neste caso, Rousseeu (1987 *apud* Souza (2007, p.17) recomenda assumir *s(i) = 0*. A Figura 12 é um exemplo de agrupamento em que as linhas representam as distâncias do i-ésimo ponto aos demais elementos da amostra.

FIGURA 12: Linhas representando as distâncias do *i*-ésimo ponto



Fonte: (SOUZA, 2007, p. 18)

O valor da silhueta de um objeto está no intervalo [- 1, 1]. Se uma instância está bem situada dentro de seu grupo, sua silhueta apresentará um valor próximo de 1. Caso contrário, um valor próximo de -1 indica que a instância deveria ser associada a outro grupo, pois o agrupamento não foi adequado. Um modo de escolher o melhor valor de k (número de grupos) é selecionar aquele que resulta no maior valor da silhueta.

A Silhueta Média (*SM*) do *j*-ésimo grupo é dada por (SOUZA, 2007, p. 18):

$$SM_j = \frac{\sum_{i=1}^{n_j} s(i)}{n_j} \tag{7}$$

O Coeficiente de Silhueta Médio (CSM) é um índice de qualidade para todo o agrupamento C, dado pela média de s(i) (SANTOS, 2007, p. 18):

$$CSM = \frac{\sum_{i=1}^{n} s(i)}{n} \tag{8}$$

A Quadro 7 apresenta a interpretação desse coeficiente.

QUADRO 7: Valores da Silhueta

S(i)	Descrição
0,71 – 1,00	Uma estrutura forte foi encontrada
0,51 – 0,70	Uma estrutura razoável foi encontrada
0,26 – 0,50	A estrutura é fraca e pode ser superficial. É aconselhável usar outros métodos para esses dados.
≤ 0,25	Nenhuma estrutura substancial foi encontrada

Fonte: (VALE, 2008, p. 30)

Para calcular os valores da silhueta, o processo de inicialização dos k centroides iniciais pode ser aleatório ou pode usar o k-*Means*++ que, apesar de ainda escolher randomicamente os k centroides iniciais, ele faz uma ponderação de acordo com o quadrado de suas distâncias àquele centroide que seja mais próximo, dentre os já escolhidos. “O processo de inicialização do *k-Means*++ não escolhe como o próximo centroide uma instância que esteja mais distante dos centroides já escolhidos mas sim, escolhe uma instância com uma probabilidade proporcional à sua distância aos centroides já escolhidos. ” (OLIVEIRA, 2018, p. 27). Assim, para verificar a eficiência das formas de inicialização dos k centroides, serão testados nos experimentos desta pesquisa a forma randômica e o *k-Means*++.

- **Índice *Davies-Bouldin***

O índice *Davies-Bouldin* não depende do número de agrupamentos e do método de partição dos dados, o que o torna adequado para avaliação de algoritmos de partição. O índice é dado em função da razão entre a soma da dispersão interna dos agrupamentos e a distância (separação) entre os *clusters* (GONÇALVES, 2005).

- **Índice *Dunn***

"O Coeficiente *Dunn* mede o grau de generalização dos agrupamentos, ou seja, o quanto um agrupamento é '*fuzzy.*'" (EVERITT, 2001 apud VALE, 2005, p. 50). "É apropriado para a identificação de *clusters* compactos e bem separados." (HALKIDI *et al*., 2002, apud FACELI *et al*., 2017, p. 243).

c) Medida relativa

No caso da medida relativa, comparam-se diretamente os diferentes grupos usando aqueles obtidos através de diferentes configurações de parâmetros para o mesmo algoritmo. Por exemplo, para o mesmo programa, usa-se um número diferente de k. Com isso, geram-se diferentes *clusters*. Para comparar os resultados, pode-se usar qualquer um dos índices acima citados para ver quão bem definindos estão os grupos.

Segundo Faceli *et al.* (2017), os índices baseados em critérios relativos comparam diversos agrupamentos para decidir qual deles é o melhor em algum aspecto. "Eles podem ser utilizados para comparar algoritmos de *clustering* ou para determinar o valor mais apropriado para o parâmetro de um algoritmo. Índices empregados em tal critério se baseiam apenas nos dados originais".

Por fim, a interpretação, segundo Faceli *et al.* (2017), refere-se ao processo de examinar cada *cluster* com relação a seus objetivos para rotulá-los, descrevendo a natureza do grupo. A interpretação dos *clusters* é mais que apenas uma descrição. Além de ser uma forma de validação dos aglomerados encontrados e da hipótese inicial, de um modo confirmatório, os *clusters* podem permitir avaliações subjetivas que tenham um significado prático. Ou seja, o especialista pode ter interesse em encontrar diferenças semânticas de acordo com os atributos de cada grupo. "Nessa etapa, é fundamental o apoio do especialista de domínio, pois é com o conhecimento a respeito dos dados que é possível identificar significados para os *clusters* e as possíveis relações entre eles." (FACELI *et al.*, 2017, p. 207). Além disso, os autores afirmam que as formas de visualizar os aglomerados obtidos são de grande ajuda por fornecer ao especialista do domínio uma maneira fácil e intuitiva de observar os resultados do processo de *clustering*.

2.3 Aprendizado de Máquina

Aprendizado de Máquina (AM), ou *Machine Learning*, pode ser definido como uma área que pesquisa métodos computacionais relacionados à aquisição automática de novos conhecimentos, novas habilidades e novas formas de organizar a informação já existente (Mitchell, 1997). Na visão de Muller e Guido (2017), AM é um campo de pesquisa que tem uma interseção com a Estatística, Inteligência Artificial e Ciência da Computação e que também é conhecido como Análise Preditiva ou Aprendizagem Estatística.

> Aprendizado de Máquina é uma área de Inteligência Artificial (AI) cujo objetivo é o desenvolvimento de técnicas computacionais sobre o aprendizado, bem como a construção de sistemas capazes de adquirir conhecimento de forma automática. Um sistema de aprendizado é um programa de computador que toma decisões com base em experiências acumuladas por meio da solução bem-sucedida de problemas anteriores. (MONARD; BARANAUSKAS, 2003, p. 89)

Segundo Mitchell (1997), AM pode ser definido como a capacidade de melhorar o desempenho na realização de alguma tarefa por meio da experiência. Desta forma, no Aprendizado de Máquina, os computadores são programados para aprender com a experiência anterior. Monard e Baranauskas (2003, p. 90) afirmam que o AM é uma ferramenta poderosa para a aquisição automática de conhecimento. Porém, deve-se observar que não existe uma única técnica que apresente o melhor desempenho para todos os problemas. Sendo assim, “é importante compreender o poder e a limitação dos diversos algoritmos de AM utilizando alguma metodologia que permita avaliar os conceitos induzidos por esses algoritmos em determinados problemas”.

Baeza-Yates e Ribeiro-Neto (2013) afirmam que o AM é uma área ampla da IA que está preocupada com o projeto e o desenvolvimento de algoritmos que aprendem a partir dos dados fornecidos como entrada. Nesse contexto, “os padrões aprendidos, que podem ser bem complexos, são então usados para fazer predições relativas a dados ainda não vistos e novos.” (BAEZA-YATES; RIBEIRO-NETO, 2013, p. 278).

Nesse contexto, o Aprendizado de Máquina é semelhante ao aprendizado humano em pelo menos um aspecto: é um aprendizado baseado em experiências. A máquina aprende através do reconhecimento de padrões. Dessa forma, quanto maior for a quantidade de dados em que a máquina for “alimentada”, mais preciso será o reconhecimento desses padrões.

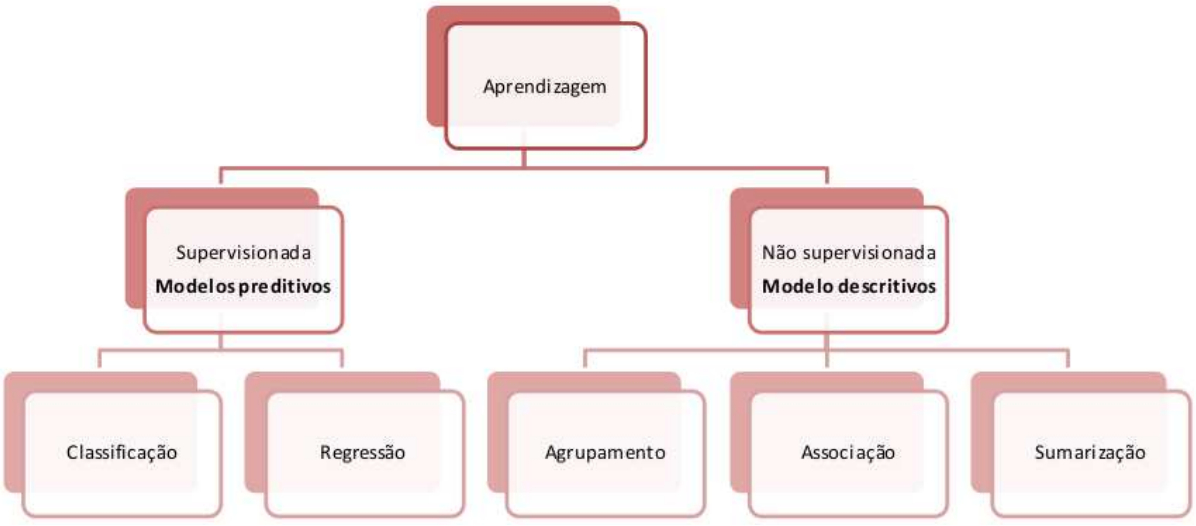
Para os autores Muller e Guido (2017), as aplicações de métodos de *Machine Learning*, nos últimos anos, tornaram presentes na vida cotidiana. Desde as recomendações automáticas de quais filmes assistir, que comida pedir ou quais produtos comprar, rádio *on-line* personalizado, reconhecimento de amigos por meio de fotos etc. Isso se tornou possível devido a muitos sites e dispositivos modernos possuírem algoritmos de Aprendizado de

Máquinas em seus núcleos. Sites complexos como o Facebook, Amazon e Netflix e aplicações comerciais contêm modelos de AM implementados. Além disso, o Aprendizado de Máquina teve uma tremenda influência nos sistemas de Recuperação da Informação e na forma como a pesquisa é feita atualmente.

Segundo Faceli *et al.* (2017), AM é uma das áreas de pesquisa que mais tem crescido nos últimos anos. Assim, por sempre surgirem novas variações nas características dos problemas reais a serem tratados, a todo momento são propostas formas diferentes de utilizar os algoritmos existentes, além de serem realizadas continuamente adaptações nesses programas. Dessa forma, se torna importante novos estudos nessa área, pois são muitas as aplicações que já beneficiam e podem se beneficiar ainda mais do Aprendizado de Máquina, visto que é possível desenvolver algoritmos cada vez mais eficazes, além da alta capacidade de recursos computacionais atualmente disponíveis, o que facilita o desenvolvimento de tecnologias cada vez mais potentes.

Por conseguinte, os algoritmos de AM são amplamente empregados em diversas tarefas, que podem ser organizadas de acordo com o método de aprendizado a ser adotado. Com base nos critérios escolhidos para a realização de cada atividade, os métodos de aprendizado podem ser divididos em aprendizado supervisionado e aprendizado não supervisionado, conforme ilustrado na Figura 13.

FIGURA 13: Hierarquia do Aprendizado



Fonte: Adaptação de Monard e Baranauskas (2003)

Segundo Monard e Baranauskas (2003, p. 90), “a indução é a forma de inferência lógica que permite obter conclusões genéricas sobre um conjunto particular de exemplos. Ela é caracterizada pelo raciocínio originado em um conceito específico e generalizado, ou seja,

da parte para o todo". Conforme ilustrado na Figura 13, o aprendizado indutivo subdivide em supervisionado e não supervisionado, que serão descritos nas seções 2.3.1 e 2.3.2

2.3.1 Aprendizado supervisionado

Segundo Matos (*on-line*), o termo aprendizado supervisionado é usado sempre que o programa é "treinado" sobre um conjunto de elementos pré-estabelecido. "Baseado no treinamento com os dados pré-definidos, o programa pode tomar decisões precisas quando recebe novos dados". Aprendizado supervisionado "refere-se à capacidade que determinados algoritmos têm de aprender a partir de exemplos." (GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 73). Ou seja, nesse tipo de aprendizado, a máquina é alimentada com uma amostragem, que é rotulada de acordo com suas características dominantes. Assim, o algoritmo vai aprender a partir dessas amostras. Na prática, é como se a máquina fosse instruída com a orientação de um professor, por isso o termo aprendizado supervisionado.

Compreende-se, assim, que nesse método é fornecido ao algoritmo de aprendizado, ou indutor, um conjunto de exemplos de treinamento para os quais o rótulo da classe associada é conhecido (CHEESEMAN; STUTZ, 1996).

> O **aprendizado supervisionado** compreende a abstração de um modelo de conhecimento a partir dos dados apresentados na forma de pares ordenados (entrada, saída desejada). Por **entrada** entende-se o conjunto de valores das variáveis (atributos) de entrada do algoritmo para um determinado caso. Tais variáveis são denominadas **atributos previsores**. A saída desejada corresponde ao valor de uma variável (denominada **atributo-alvo**) que se espera que o algoritmo possa produzir sempre que receber os valores especificados em **entrada**. (GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 73).

Muller e Guido (2017) afirmam que aprendizagem supervisionada é usada sempre que se deseja prever uma determinada saída a partir de uma determinada entrada, e existem exemplos de pares de entrada / saída, ou seja, tem-se um conjunto de dados cujas saídas já são conhecidas.

> Um algoritmo é dito supervisionado quando usa informação fornecida por seres humanos ou obtida por meio de assistência humana como dado de entrada. No caso padrão, um conjunto de classes e exemplos de documentos para cada classe são fornecidos. Os exemplos são determinados por especialistas humanos e constituem o conjunto de treinamento, que é então utilizado para aprender uma função de classificação. Uma vez que essa função for aprendida, é então usada para classificar novos documentos não vistos. (GONÇALVES, 2013, p. 281).

Assim, um modelo de Aprendizado de Máquina é construído a partir de pares de entrada / saída que compõem o conjunto de treinamento. Desse modo, o objetivo é fazer previsões precisas para novos dados cujas saídas são desconhecidas. Apesar do

aprendizado supervisionado geralmente requerer esforço humano para construir o conjunto de treinamento, depois a tarefa é automatizada e, na maioria das vezes, acelera uma tarefa árdua ou inviável manualmente.

Para Markov e Larose (2007), a partir de uma coleção de objetos identificados com uma classe, o sistema de aprendizagem cria um mapeamento entre os elementos e as classes que pode ser usado para classificar novos objetos que ainda não foram rotulados. Como a rotulagem (classificação) do conjunto inicial (treinamento) é feita por um agente externo ao sistema, essa configuração é chamada de aprendizagem supervisionada. Assim, os algoritmos desse modelo são treinados usando exemplos rotulados, tendo um conjunto de dados de entrada e as possíveis saídas desejadas. Ou seja, o algoritmo de aprendizagem recebe um conjunto de dados de entrada junto com as saídas corretas correspondentes. Com base em uma coleção de exemplos, o programa faz previsões, buscando padrões nos rótulos. Ao encontrar o melhor padrão possível, ele utilizará essa referência para fazer predições para dados de testes sem rótulo.

> O aprendizado supervisionado requer uma função de aprendizado dos dados de treinamento fornecidos como entrada. No caso da classificação de textos, os dados de treinamento são compostos de pares do tipo documento-classe indicando as classes corretas para os documentos dados, de acordo com especialistas humanos. Esses dados de treinamento são então usados para aprender uma função de classificação, a qual pode ser usada para fazer predições de classes para dados ainda não vistos ou novos. A abordagem só funciona se a função aprendida for tal que ela possa pegar dados que não foram vistos antes e prever classes para eles com alta precisão. (BAEZA-YATES; RIBEIRO-NETO, 2013, p. 278).

Goldschmidt, Passos e Bezerra (2015, p. 73), com o objetivo de formalizar a ideia subjacente ao aprendizado supervisionado, consideraram um conjunto de pares ordenados em que cada par é da forma ***(x, f(x))***, onde:

- ***x*** é um vetor com os valores dos atributos previsores;

- ***f(x)*** é o valor do atributo alvo, que corresponde à saída de uma função ***f***, desconhecida, aplicada a ***x***.

Assim, cada um desses pares é denominado um exemplo de ***f***.

Com o propósito de explicar melhor a função, os autores Goldschmidt, Passos e Bezerra (2015) descrevem que:

> O aprendizado supervisionado consiste em, dada uma coleção de exemplos de ***f***, obter uma função ***h*** que seja uma aproximação de ***f***. A função ***h*** é chamada de **hipótese** ou **modelo** de ***f***. [...]. A identificação da função ***h*** consiste de um processo de busca, em um espaço de **hipótese candidatas *H***, pela hipótese que mais se aproxime da função original ***f***. Esse processo de busca é o aprendizado supervisionado e a hipótese selecionada é o modelo de conhecimento abstraído a partir dos dados. Todo algoritmo que possa ser utilizado na execução do aprendizado é chamado algoritmo de

> aprendizado. O conjunto de todas as hipóteses que podem ser obtidas a partir de um algoritmo de aprendizado ***L*** é representado por H_L. Cada hipótese pertencente a H_L é representada por h_L.
>
> A acurácia de uma hipótese ***h*** retrata a capacidade de ***h*** em mapear corretamente cada vetor de entradas ***x*** em ***f(x).*** O conjunto de pares ***(x, f(x))*** utilizados na identificação da função ***h*** é denominado conjunto de treinamento. Por outro lado, o conjunto de pares ***(x, f(x))*** utilizados para avaliar a acurácia de ***h*** é denominado conjunto de teste. Assim, o algoritmo de aprendizado ***L*** pode ser interpretado como uma função $L: T \rightarrow H_L$, onde ***T*** é o espaço composto por todos os conjuntos de treinamento possíveis para ***L***. (GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 73).

Portanto, o aprendizado supervisionado utiliza padrões para prever os dados do rótulo, também chamado de *labels*, em dados adicionais não rotulados. Esse modelo de aprendizagem se divide em duas subcategorias: Classificação e Regressão.

2.3.1.1 Classificação

Classificação é o processo de adotar algum tipo de entrada e atribuir um rótulo a ela. De acordo com Gonçalves (2013, p. 277):

> Desde os tempos antigos, dos primeiros dias da Grande Biblioteca de Alexandria por volta de 300 a. C., bibliotecários tinham que lidar com armazenamento de documentos para recuperação e leitura futura. Com o passar do tempo, o tamanho das coleções cresceu e o problema tornou-se cada vez mais difícil. Procurar por um livro em particular dentre centenas de livros tornou-se uma tarefa tediosa, demorada e impraticável. Para aliviar o problema, bibliotecários começaram a rotular os documentos. Isso forneceu metainformação ao seu conteúdo, permitindo assim que os livros fossem organizados com uma visão que permitia busca rápida e recuperação. Uma das primeiras abordagens para rotular documentos foi atribuir um identificador único para cada documento. Isso resolvia o problema sempre que o usuário soubesse dos identificadores dos livros que eles queriam, mas não resolvia o problema mais genérico de encontrar documentos sobre um assunto ou *tópico* específico. Nesse caso, a solução natural é agrupar os documentos por tópicos comuns e nomear cada grupo com um ou mais rótulos significativos. Cada grupo rotulado é o que chamamos de uma *classe*, isto é, um conjunto no qual podemos inserir documentos cujo conteúdo pode ser descrito pelo seu rótulo.

Esse processo de inserção dos documentos em classes é comumente conhecido como classificação de textos.

Desta forma, com a finalidade de aprimorar os recursos para organização da informação, estudos foram realizados e algoritmos foram desenvolvidos para a indução automática de sistemas capazes de lidar com problemas de classificação.

Segundo Jacob (1991), os sistemas de classificação são usados para organizar o conhecimento. A classificação como processo envolve a atribuição ordenada e sistemática de cada entidade a uma e apenas uma classe dentro de um sistema de classes mutuamente exclusivas e não sobrepostas. Este processo é legal e sistemático: lícito porque é realizado

de acordo com um conjunto estabelecido de princípios que governam a estrutura de classes e as relações entre elas; e sistemática porque exige a aplicação consistente desses fundamentos dentro da estrutura de uma ordenação prescrita da realidade. O esquema em si é artificial e arbitrário: artificial porque é uma ferramenta criada com o propósito expresso de estabelecer uma organização significativa; e arbitrário, porque os critérios usados para definir classes no esquema refletem uma perspectiva única do domínio, excluindo todas as outras perspectivas (JACOB, 2004).

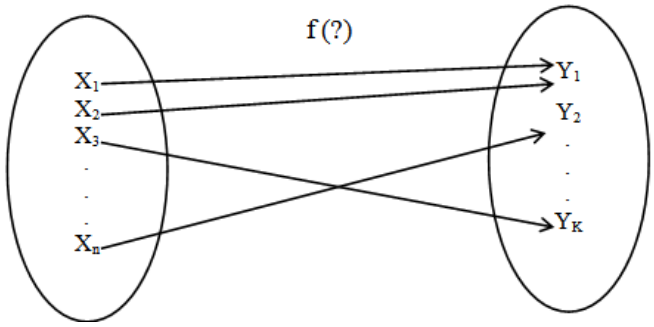
Segundo Goldschmidt, Passos e Bezerra (2015), na tarefa de classificação os atributos do conjunto de dados são divididos em dois tipos: Atributo previsor e atributo-alvo. Assim,

> para cada valor distinto do atributo-alvo tem-se uma classe que normalmente corresponde a um rótulo categórico pertencente a um conjunto predefinido. A tarefa de classificação consiste em descobrir uma função que mapeie um conjunto de registros em um conjunto de classes. Uma vez descoberta, tal função pode ser aplicada a novos registros de forma a prever a classe em que tais registros se enquadram. Como exemplo, considere uma financeira que possui o histórico de seus clientes e o comportamento destes em relação ao pagamento de empréstimos contraídos previamente. Considere também dois tipos de clientes: adimplentes e inadimplentes. Essas que são as classes dos problemas (valores do atributo-alvo). Uma aplicação da tarefa de classificação, neste caso, consiste em descobrir uma função que mapeie corretamente os clientes, a partir de seus dados (valores dos atributos previsores). (GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 25).

Em síntese, na tarefa de classificação, um dos grupos possui somente um atributo, que corresponde ao atributo-alvo, ou seja, a propriedade pela qual se deve fazer a predição de um valor. Nesse caso, o atributo é categórico (domínio composto por categorias/classes) e o outro conjunto contém os atributos a serem utilizados na predição do valor, denominados previsores ou de predição.

De acordo com Goldschmidt, Passos e Bezerra (2015, p. 89), a tarefa de classificação pode ser compreendida como “a busca por uma função que permita associar corretamente cada registro X_i de um conjunto de dados a um único rótulo categórico $\mathbf{Y_i}$, denominado classe. Uma vez identificada, essa função pode ser aplicada a novos registros de forma a prever as classes em que tais registros se enquadram”. A Figura 14 ilustra as associações entre registros de dados e suas respectivas classes.

FIGURA 14: Associações entre registros de dados e classes



Fonte: GOLDSCHMIDT, PASSOS E BEZERRA (2015, p. 89)

Nesse contexto, Goldschmidt, Passos e Bezerra (2015, p. 89) formalizaram a classificação da seguinte forma:

> Considere um conjunto de pares ordenados em que cada par é da forma ***(x, f(x))***, onde ***x*** é um vetor de entrada n-dimensional e ***f(x)*** a saída de uma função ***f***, desconhecida, aplicada a ***x***. A tarefa de classificação consiste em, dada uma coleção de exemplos de ***f***, obter uma função (hipótese) ***h*** que seja uma aproximação de ***f***. A imagem de ***f*** é formada por rótulos de classes retirados de um conjunto finito e toda hipótese ***h*** chamada de **Classificador**. O aprendizado consiste na busca pela hipótese ***h*** que mais se aproxime da função original ***f***. (GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 89).

Ademais, os sistemas de classificação são usados geralmente quando as previsões são de natureza distinta, ou seja, um simples “sim ou não”. Exemplo: Mapeamento da imagem de uma pessoa para classificar como masculino ou feminino. Vale destacar que o conceito de classificação é o mesmo, independentemente do tipo ou natureza do objeto (dado estruturado, texto, imagem, som) a ser classificado.

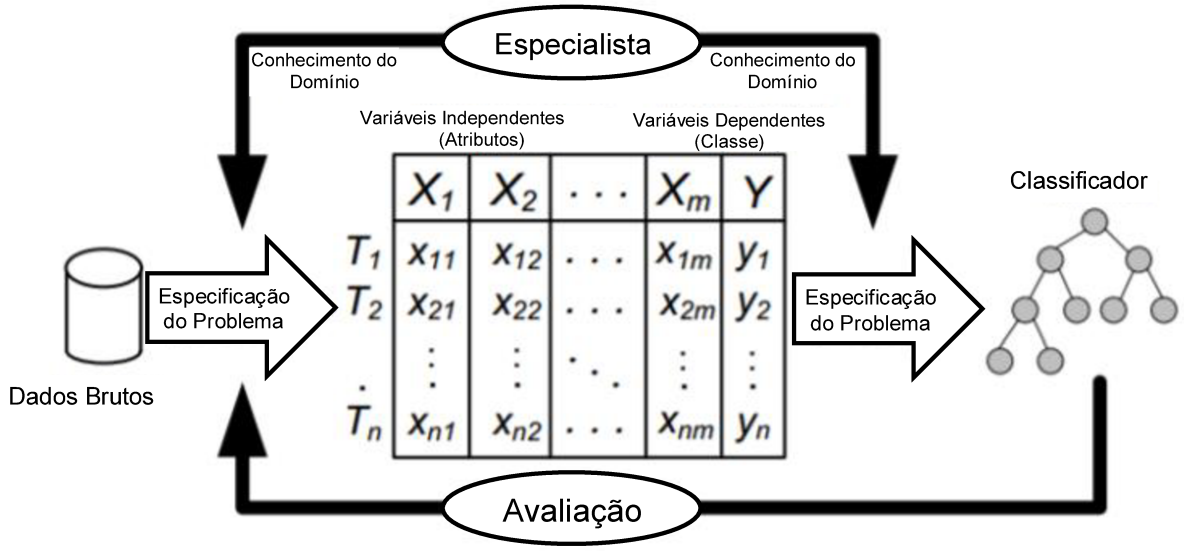
Para Denecke (2008), a classificação de documentos pode ser desenvolvida, basicamente, por meio de dois métodos, ou seja, por aprendizagem estatística de máquina e por seleção de recursos lexicais e processamento de linguagem natural. Segundo Girinardi (1995), nas técnicas baseadas no modelo estatístico, os termos de indexação são extraídos a partir de uma análise de frequência das palavras ou frases em cada documento e em toda a fonte de informação. Nas técnicas linguísticas, os termos de indexação são extraídos utilizando técnicas de processamento da linguagem natural, como, por exemplo, análise morfológica, lexical, sintática e semântica.

De acordo com Gonçalves (2013, p. 278), a classificação de textos provê um meio de organizar a informação que permite melhor compreensão e interpretação de dados. Para ilustrar isso, o autor apresenta o exemplo de uma grande companhia de engenharia que executa dezenas de projetos anualmente, sendo que cada projeto dessa empresa gera

centenas de escritos, resultando em milhares de documentos produzidos pela companhia. Nesse contexto, separar esses textos em um conjunto de classes gera uma visão estruturada de toda a informação do negócio, que se constitui em um ativo valioso para auxiliar o processo de tomada de decisão.

Em suma, na classificação, o algoritmo de aprendizado constrói um classificador capaz de determinar corretamente a classe de novos exemplos ainda não etiquetados, dado um conjunto de classes e um conjunto de exemplos de treinamento. Na maioria dos casos, esse processo pode usar o conhecimento de um domínio para fornecer alguma informação previamente conhecida como entrada ao indutor. E, após induzido, o classificador é normalmente avaliado e o processo de classificação pode ser repetido, se necessário (MONARD, BARANAUSKAS, 2003). A Figura 15 ilustra esse processo.

FIGURA 15: Processo de classificação



Fonte: MONARD; BARANAUSKAS (2003, p. 92)

> Primeiramente, os dados brutos são preparados em um conjunto de exemplos para que possam ser processados. Um conjunto de exemplos é composto por valores de atributos, que são características do exemplo, e pelo atributo classe. Nessa figura é mostrado o formato padrão de um conjunto de exemplos *T* com m exemplos e *n* atributos. A linha *i* refere-se ao *i*-ésimo exemplo onde $i = 1,2,...,$ n e a entrada x_{ij} refere-se ao valor do *j*-ésimo atributo X_j do exemplo $_i$, onde $j = 1,2,..., m$. Após o processamento dos dados, esse conjunto de exemplos será submetido à entrada do algoritmo de indução para que seja feito o treinamento do classificador. O objetivo do treinamento é encontrar uma função que mapeie cada exemplo T_i com a sua classe y_i correspondente. (BORGES, 2012, p. 7).

Assim, “após a etapa de treinamento, tem-se um classificador que deve ser capaz de predizer corretamente o rótulo de novos exemplos, que ainda não foram rotulados.”

(REZENDE, 2005 apud BORGES, 2012, p. 8). De acordo com Monard e Baranauskas (2003), normalmente, um conjunto de exemplos é dividido em dois subconjuntos: A coleção de treinamento é utilizada para o aprendizado do conceito e a de testes usada para medir o grau de efetividade do conceito aprendido. "Esses conjuntos são normalmente disjuntos para assegurar que as medidas obtidas, utilizando o conjunto de testes, sejam de um conjunto diferente do usado para realizar o aprendizado, tornando a medida estatisticamente válida. " (MONARD, BARANAUSKAS, 2003, p. 97).

Existem na literatura diversos algoritmos de aprendizado supervisionado para classificação de textos. Segundo Gonçalves (2013), os algoritmos supervisionados de classificação de textos mais representativos são: Árvore de Decisão, k-Vizinhos mais Próximos (*k-Nearest Neighbor* (kNN)), *Rocchio*, *Bayes* Ingênuo, Máquinas de Vetores de Suporte (*Support Vector Machines* (SVM)) e Ensembles. Porém, como esses algoritmos não serão usados nesta pesquisa, realizar-se-á uma descrição mais detalhada apenas com as técnicas de aprendizado não supervisionado.

2.3.1.2 Regressão

A regressão é similar à classificação, pois busca fazer previsões. Esses prognósticos são derivados de experiências ou conhecimentos anteriores. É uma subcategoria de aprendizagem supervisionada usada quando o valor que está sendo previsto difere de um "sim ou não" e segue um espectro contínuo. Por exemplo, dada uma imagem de um homem ou de uma mulher, precisa-se prever sua idade com base nas informações da imagem.

De acordo com Michie *et al.* (1994), "a regressão compreende a busca por uma função que mapeie os registros de um banco de dados em um intervalo de valores reais. Esta tarefa é similar à tarefa de classificação, com a diferença de que o atributo-alvo assume valores numéricos." (citado por GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 25). A técnica de regressão é comumente usada para modelar relações complexas entre elementos de dados fazendo estimativa de uma variável a partir da outra.

2.3.2 Aprendizado não supervisionado

A aprendizagem não supervisionada é o termo usado quando um programa pode automaticamente encontrar padrões e relações em um conjunto de dados, como, por exemplo, fazer a análise e o agrupamento de um conjunto de e-mails, sem que o programa possua qualquer conhecimento prévio sobre os dados (DAS, 2017). Segundo Muller e Guido (2017), o aprendizado não supervisionado inclui todos os tipos de Aprendizado de Máquina que não tem saída conhecida, nenhum conhecimento é passado para o algoritmo de

aprendizado, ele apenas é alimentado com um conjunto de dados de entrada e solicitado a extrair conhecimento a partir dessas informações.

> No Aprendizado de Máquina não supervisionado, também conhecido como aprendizado por observação e descoberta, a tarefa do algoritmo é agrupar exemplos não rotulados, i.e., exemplos que não possuem o atributo classe especificado. Nesse caso, é possível utilizar algoritmos de aprendizado para descobrir padrões nos dados a partir de alguma caracterização de regularidade, sendo esses padrões denominados *clusters*. Exemplos contidos em um mesmo cluster são mais similares, segundo alguma medida de similaridade, do que aqueles contidos em *clusters* diferentes. (MATSUBARA, 2004, p. 41)

Portanto, no aprendizado não supervisionado não há classe associada aos exemplos e o indutor analisa os elementos fornecidos e tenta determinar se alguns deles podem ser agrupados de alguma maneira, formando grupos de objetos similares (CHEESEMAN; STUTZ, 1996). Nesse tipo de aprendizado, o conjunto de treinamento consiste apenas de exemplos sem nenhum valor associado. Tipicamente, o problema resume-se em particionar a amostra de treinamento em *clusters*, através de técnicas de clusterização.

Segundo Gonçalves (2013), o aprendizado não supervisionado distingue-se do aprendizado supervisionado por não ser fornecido nenhum dado de treinamento. “Os algoritmos de aprendizado não supervisionados incluem modelos de redes neurais, análise de componentes independentes e *clustering*. Para propósito de categorização de textos, o *clustering* é o tipo de algoritmo de aprendizado não supervisionado de maior interesse.” (GONÇALVES, 2013, p. 278). Assim sendo, esse tema será abordado em uma nova seção.

2.3.2.1 Categorização

Deepthi e Prasad (2013) consideram que *clustering* de documentos e categorização de texto são sinônimos. De acordo com os autores, essa técnica está relacionada com o raciocínio por trás do agrupamento de dados. “O agrupamento de documentos é basicamente uma técnica mais específica para organização não supervisionada de documentos, extração automática de tópicos e recuperação ou filtragem rápida de informações.” (DEEPTHI; PRASAD, 2013, p. 76).

Jacob (2004), no artigo “*Classification and Categorization: A Difference that Makes a Difference*”, afirma que embora existam semelhanças óbvias entre classificação e categorização, as diferenças entre esses termos têm implicações significativas para a constituição de um ambiente de informação. Para a autora, a falta de distinção entre esses dois sistemas de organização parece resultar em uma concepção errônea de que eles são, de fato, sinônimos. Um equívoco que pode ser reforçado pelo fato de que ambos são mecanismos para organizar a informação.

Segundo Jacob (2004, p. 518), a categorização é o processo de dividir o mundo em grupos de entidades cujos membros são, de alguma forma, semelhantes entre si. Ou seja, a categorização divide o mundo em grupos ou categorias cujos membros compartilham alguma similaridade perceptível dentro de um dado contexto. “Categorizar é agrupar entidades (objetos, ideias, ações, etc.) por semelhança. ” (LIMA, 2010, p. 109).

A teoria clássica das categorias se baseia em três proposições básicas (SMITH; MEDIN,1981 *apud* JACOB, 2004, p. 520):

1. A intenção de uma categoria é uma representação resumida de uma categoria inteira de entidades.

2. As características essenciais que compõem a intenção de uma categoria são individualmente necessárias e conjuntamente suficientes para determinar a associação dentro do grupo.

3. Se uma categoria (A) estiver aninhada dentro da categoria superordenada (B), os recursos que definem a categoria (B) estão contidos no conjunto de recursos que definem a categoria (A).

Proposição I: Afirma que a definição (**intenção**) de uma categoria é a união das características essenciais que identificam a associação (**extensão**) dessa categoria. Além disso, como todos os membros de uma única categoria devem compartilhar esse conjunto de características essenciais, cada membro é igualmente representativo da categoria como um todo. Por essa razão, a estrutura interna de uma categoria é considerada não classificada, ou sem classificação, porque nenhum membro pode ser mais típico ou mais representativo de uma categoria do que qualquer outro membro.

Proposição II: A proposição II estabelece que, como cada membro da categoria deve exibir todas as características essenciais que compõem a intenção da categoria, a posse do conjunto de recursos que define a categoria é suficiente para determinar a associação na categoria. E existe um relacionamento binário entre uma entidade e uma categoria, de tal forma que uma entidade é ou não é um membro de uma categoria particular. Assim, os limites das categorias são ditos fixos e rígidos.

Proposição III: Identifica o relacionamento de herança existente entre categorias em uma estrutura hierárquica: qualquer membro de uma categoria que seja um subconjunto de uma categoria superior deve exibir não apenas o conjunto de recursos essenciais que determinam a associação no subconjunto, mas também o conjunto de recursos essenciais que determinam a associação em qualquer categoria superior na qual o subconjunto é aninhado.

Segundo Jacob (2004), a teoria clássica sustenta que pertencer a uma categoria específica (extensão) implica a posse do caráter essencial e definidor (intenção) da categoria.

Por exemplo, se a intenção da categoria “pássaro” consiste nas características “põe ovos”, “tem asas”, “voa” e “constrói ninhos em lugares altos”, cada membro da categoria deve exemplificar o conjunto completo dos recursos definidos. Deste modo, se uma entidade não voar, ela não poderá ser considerada membro da categoria “pássaro”, mesmo que ponha ovos, tenha asas e construa ninhos em lugares altos. E, como todos os membros da categoria são definidos pelo mesmo conjunto de características, nenhum pássaro pode ser mais típico ou mais representativo da categoria do que qualquer outro pássaro. Assim, de acordo com a teoria clássica, um papagaio e um pombo seriam igualmente representativos da categoria “pássaro”.

> Embora sistemas de classificação e categorização sejam tantos mecanismos para estabelecer ordem através do agrupamento de fenômenos relacionados, diferenças fundamentais entre eles influenciam em como esta ordem é efetuada - diferenças que fazem uma diferença no contexto de informação estabelecido por cada um desses sistemas. Enquanto a classificação tradicional é rigorosa na medida em que determina que uma entidade é ou não é um membro de uma classe particular, o processo de categorização é flexível e criativo e desenha associações não-vinculantes entre entidades - associações que são baseadas não em um conjunto de princípios pré-determinados, mas no simples reconhecimento de similaridades que existem através de um conjunto de entidades. Classificação divide um universo de entidades em um sistema arbitrário de classes mutuamente exclusivas e não sobrepostas que são arranjadas dentro do contexto conceitual estabelecido por um conjunto de princípios estabelecidos. O fato de que nem o contexto nem a composição dessas classes variam é a base para a estabilidade de referência fornecida por um sistema de classificação. Ao contrário, categorização divide o mundo da experiência em grupos de categorias nos quais os membros portam alguma similaridade imediata dentro de um dado contexto. Que este contexto pode variar - e com ele a composição da categoria - é a base tanto para a flexibilidade e o poder de categorização cognitiva. (JACOB, 1992 traduzido por GARRIDO, 2011, p. 13).

Existem poucos trabalham que tratam as diferenças entre os termos classificação e categorização. A literatura apresenta, muitas vezes, as duas palavras como sinônimas. Essa imprecisão terminológica obscurece o entendimento dos pesquisadores e dificulta o esclarecimento das diferenças desses vocábulos no contexto de sistemas de organização da informação (JACOB, 1992). Sendo assim, o processo de agrupamento é, na verdade, uma forma de categorizar porque na classificação, as classes já existem previamente.

A técnica de *clustering* é um exemplo de categorização e será descrita no próximo tópico.

2.3.2.2 Clusterização

Segundo Goldschmidt, Passos e Bezerra (2015, p. 25), a clusterização ou agrupamento é uma técnica “utilizada para segmentar os registros de uma base de dados em

subconjuntos ou *clusters*, de tal forma que os elementos de um *cluster* compartilhem propriedades comuns que os distingam de elementos nos demais *clusters*", e cujo objetivo é maximizar a similaridade intracluster e minimizar a similaridade intercluster. "Diferente da tarefa de classificação, em que cada registro está associado a um ou mais rótulos predefinidos, a clusterização precisa identificar os grupos de dados." (FAYYADD et al., 1996 citado por GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 26).

Segundo Hair *et al*. (2005), a análise de agrupamentos é uma técnica analítica para desenvolver subgrupos significativos de indivíduos ou objetos. Especificamente, o objetivo é categorizar uma amostra de entidades (indivíduos ou objetos) em um número menor de grupos mutuamente excludentes, com base nas similaridades entre as entidades.

Clusterização tem sido estudado nas áreas de Processamento de Linguagem Natural, Mineração de Dados, Recuperação de Informações e Reconhecimento de Padrões. É um modelo de representação espacial, em que um conjunto de objetos é distribuído em subconjuntos menores, chamados de *clusters* (ZURINI; SBORA, 2011), que usa Aprendizado de Máquina e PLN para entender e categorizar dados textuais não estruturados.

Segundo Faceli *et al*. (2017) não existe uma definição formal e única para *cluster*, muito pelo contrário, existe uma grande variedade de definições na literatura para esse termo. Bárbara (2000, apud Faceli *et al*., 2017, p.192) apresenta algumas definições comuns para *clusters*. Segundo o autor, eles podem ser separados, baseados em centro, contínuos ou encadeados, baseados em densidade e em similaridade.

- *Cluster* bem separado: é um conjunto de pontos tal que qualquer ponto em um determinado grupo está mais próximo ou é mais similar a cada outro ponto nesse *cluster* do que a qualquer outro ponto não pertencente a ele.
- *Cluster* baseado em centro: é um conjunto de pontos tal que qualquer ponto em um dado *cluste*r está mais próximo ou é mais similar ao centro desse *cluste*r do que ao centro de qualquer outro grupo. Neste caso, o centro do *cluster* pode ser um centroide, como a média aritmética dos pontos do aglomerado, ou um metoide, que é o ponto mais representativo do *cluster*.
- *Cluster* contínuo ou encadeado: É um conjunto de pontos tal que qualquer ponto em um dado *cluster* está mais próximo ou é mais similar a um ou mais pontos nesse *cluster* do que a qualquer ponto que não pertença a ele. Esse aglomerado também é conhecido como vizinho mais próximo ou agrupamento transitivo.
- *Cluster* baseado em densidade: um *cluster* é uma região densa de pontos, separada de outras regiões de alta densidade por áreas de baixa densidade.
- *Cluster* baseado em similaridades: Um *cluster* é um conjunto de pontos que são similares, enquanto pontos em *clusters* diferentes não são similares.

Assim, “cada definição de *cluster* resulta em critério de agrupamento que essencialmente é uma forma de selecionar uma estrutura ou modelo para representar os grupos que melhor se ajustam a um determinado conjunto de dados.” (ESTIVILL-CASTRO, 2002 apud FACELI *et al*., 2017, p. 192).

A propósito, o processo de criar grupos de objetos semelhantes é conhecido como *clustering*. Agrupamento ou *clustering* é uma técnica de organizar dados em grupos cujos membros apresentam alguma semelhança. É, portanto, uma coleção em que as instâncias de dados em um mesmo *cluster* são semelhantes entre si e diferentes das instâncias de dados dos outros *clusters*. Portanto, a tarefa de *clustering* é encontrar grupos ocultos em um conjunto de dados (LIU, 2007).

Segundo Jain e Dubes (1988), o objetivo da técnica de agrupamento é encontrar uma estrutura nos dados em que os objetos pertencentes a cada *cluster* compartilham alguma característica ou propriedade relevante para o domínio do problema em estudo, ou seja, são similares de alguma maneira. Sendo assim, a finalidade do *clustering* é formar grupos diferentes, mas não forçosamente disjuntos, contendo membros muito semelhantes entre eles. Ao contrário do processo de classificação, que segmenta a informação associando-a a grupos já determinados, o *clustering* é uma forma de segmentar informação em grupos não previamente definidos (CAMPOS, 2005).

Para Markov e Larose (2007), c*lustering* é um modelo de aprendizagem que não usa objetos rotulados e, portanto, não é supervisionado. É uma técnica em que nenhuma suposição é feita a respeito dos grupos. Diferente do conceito de classificação, o processo de clusterização não conta com classes predefinidas e exemplos de treinamento rotulados. Deste modo, realiza uma forma de aprendizado sem nenhuma forma de supervisão.

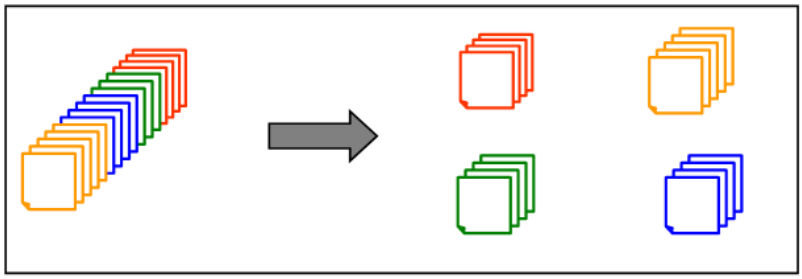
> O Modelo de Aglomerados (*Clusters*), também conhecido como *Clustering Model*, utiliza técnicas de Agrupamento (ou *Clustering*) de documentos. Seu funcionamento consiste em identificar documentos de conteúdo similar (que tratem de assuntos parecidos) e armazená-los ou indexá-los em um mesmo grupo ou aglomerado (cluster). A identificação de documentos similares em conteúdo dá-se pela quantidade de palavras similares e frequentes que eles contêm. (MORAES e AMBROSIO, 2007, p. 11).

Em síntese, a técnica de agrupamento de texto busca agrupar uma coleção de textos não estruturados em grupos de categorias diferentes para que os documentos no mesmo *cluster* descrevam o mesmo assunto (KRISHNA; BHAVANI, 2010).

Portanto, clusterização é um método usado para criar grupos com base em propriedades comuns aos itens. O objetivo é encontrar padrões, agrupando objetos semelhantes ou organizando-os de acordo com suas características. Os dados são reunidos por similaridades, onde os itens dentro de cada *cluster* possuem muitas características em

comum e muitas diferenças comparadas aos itens de outros grupos (DAS, 2017). A Figura 16 esboça esse processo.

FIGURA 16: Separação de documentos semelhantes em *clusters*



Fonte: Elaborada pela autora

Ferlin (2008, citado por Goldschmidt; Passos; Bezerra, 2015, p. 96) afirma que alguns algoritmos de agrupamento requerem que o usuário forneça o número de k (número de *clusters* a formar). Assim, de acordo com esse valor, os objetos são então separados de forma que elementos mais similares sejam alocados aos mesmos grupos e os menos similares a grupos diferentes.

Por conseguinte, um sistema de agrupamento pode ser útil no processo de recuperação da informação, pois agrupa os resultados da pesquisa em conjuntos de documentos estreitamente relacionados. *Clustering* pode melhorar o resultado da busca, tornando-a mais relevante, ao se concentrar em agrupar conjuntos de documentos similares (MARKOV e LAROSE, 2007). Em uma coleção de notícias, o agrupamento também é benéfico, pois facilita o acesso às informações que são interessantes para o usuário, uma vez que o leitor irá acessar somente as notícias dos *clusters* mais relevantes, ou seja, aquelas que estão de acordo com o interesse do leitor.

Segundo Markov e Larose (2007), existem vários tipos de clusterização, dependendo da forma como os grupos são representados, as propriedades do *cluster* e os tipos de algoritmos utilizados para o agrupamento. Para esses autores, há quatro dimensões ao longo das quais as técnicas de agrupamento podem ser categorizadas (MARKOV; LAROSE, 2007, p. 62):

1. **Baseado em modelo (conceitual) versus particionamento**. O agrupamento conceitual cria modelos (representações explícitas) de *clusters*, enquanto o particionamento simplesmente enumera os membros de cada grupo.

2**. Determinista versus probabilista**. A associação do grupo pode ser definida como um valor booleano (*cluster* determinístico) ou como uma probabilidade (agrupamento probabilístico).

3. **Hierárquico versus plano**. O agrupamento plano divide o conjunto de objetos em subconjuntos, enquanto o agrupamento hierárquico cria estruturas no formato de árvore hierárquica, também conhecidas como dendograma.

4. **Incremental versus lote**. Os algoritmos de lote usam todo o conjunto de objetos para criar o agrupamento, enquanto os incrementais levam um elemento por vez e atualizam o *cluster* para acomodá-lo.

Segundo Jiang *et al*. (2004), cada técnica é baseada em um critério de agrupamento e usa uma medida de proximidade e outras tecnologias para encontrar uma estrutura ótima que descreva os dados. Existe um grande número de algoritmos de agrupamento, cada um buscando formar *clusters* de acordo com um critério diferente (LAW; TOPCHY; JAIN, 2004).

Assim, com o grande aumento de informações textuais disponibilizadas na internet, os algoritmos de *clustering* tornaram-se ferramentas importantes para análise de documentos em pesquisas modernas. O agrupamento automático de textos pode ser efetuado de diferentes formas e com algoritmos distintos, que utilizam estratégias matemáticas e algorítmicas diferenciadas, com finalidades específicas na organização da informação.

> Mais especificamente, os algoritmos de agrupamento de dados são usualmente utilizados em análises exploratórias de dados, quando há pouco ou nenhum conhecimento prévio sobre os dados que serão analisados, pois eles permitem a descoberta de padrões válidos, úteis e desconhecidos nos dados. (NASSIF, 2011, p. 2).

Segundo Liu (2007), a técnica de agrupamento precisa de uma função de distância para medir a semelhança de dois pontos de dados ou uma medida de distância para verificar o espaço entre dois objetos. Para, assim, descobrir o agrupamento intrínseco dos elementos de entrada através do uso de um algoritmo de *clustering* e uma função de distância. À vista disso, espera-se que um algoritmo de agrupamento, ao ser alimentado com um conjunto de instâncias de textos, divida a coleção em subconjuntos de forma a maximizar a similaridade dos documentos do subconjunto e a dissimilaridade entre subconjuntos, onde a medida de similaridade é definida de antemão.

Conforme descrito por Agrawal *et al*. (1998), Ester *et al*. (1996), Ng e Han (1994), Han e Kamber (2001) e citado por Carlantonio (2011, p. 21), há alguns requisitos que devem ser atendidos para se obter um melhor agrupamento, dentre eles:

- **Descobrir grupos de forma arbitrária**: o formato dos *clusters*, considerando o espaço euclidiano, pode ser esférico, linear, alongado, elíptico, cilíndrico, espiralado etc. Os métodos de agrupamento baseados nas medidas de Distância Euclidiana ou Manhattan tendem a encontrar grupos esféricos de tamanho e densidade similares.

- **Identificar grupos de tamanhos variados**: Além da forma, alguns métodos tendem a fazer os *clusters* com tamanho homogêneo.

- Aceitar os diversos tipos de variáveis possíveis: Os métodos precisam ser capazes de lidar com variáveis contínuas, discretas e nominais.

- **Ser insensível à ordem de apresentação dos objetos:** Um mesmo conjunto de dados quando apresentado em diferentes ordens devem conduzir aos mesmos resultados.

- **Trabalhar com objetos com qualquer número de atributos ou dimensões:** Os métodos devem manejar, com eficiência, objetos com altas dimensões e fornecer resultados compreensíveis.

- **Ser escalável para lidar com qualquer quantidade de objetos:** Os métodos devem ser rápidos e escalonáveis em função do número de dimensões e da quantidade de objetos a serem agrupados.

- **Fornecer resultados interpretáveis e utilizáveis**: As descrições dos grupos devem ser facilmente assimiladas. Desse modo, é importante que os algoritmos utilizem representações simples.

- **Ser robusto na presença de *outliers* (ruídos):** A maioria das bases de dados do mundo real contém ruídos, dados desconhecidos ou errôneos, mas isso não deve afetar a qualidade dos grupos obtidos.

- **Exigir o mínimo de conhecimento para determinar os parâmetros de entrada:** Os valores apropriados são frequentemente desconhecidos e difíceis de determinar, especialmente, para conjunto de objetos de alta dimensionalidade. Em alguns métodos, os resultados do processo de agrupamento são bastante sensíveis aos parâmetros de entrada.

- **Aceitar restrições**: aplicações do mundo real podem necessitar agrupar objetos de acordo com vários tipos de restrições. Assim, os métodos devem encontrar grupos de dados com comportamento que satisfaça as restrições especificadas.

- **Encontrar o número adequado de *clusters***: encontrar o número ideal de *clusters* de um conjunto de objetos é uma tarefa difícil. Muitos métodos precisam de um valor de referência especificado pelo usuário.

Goldschmidt, Passos e Bezerra (2015) dividem os métodos de agrupamento em três famílias básicas: baseados em distâncias, baseados em distribuição de probabilidades e os algoritmos de agrupamento baseados em densidade. Os autores apresentam também outro modelo de categorizar que é baseado na forma do agrupamento gerado. Nesse contexto, “existem os algoritmos de agrupamento que geram partição (Algoritmos Partitivos) e os que geram hierarquia (Algoritmos Hierárquicos).” (GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 97).

Halkidi, Batistakis e Vazirgiannis (2001) descrevem vários paradigmas para algoritmos de agrupamento, entre eles, os Algoritmos de Agrupamento por Particionamento, os Algoritmos de Agrupamento Probabilístico ou Nebuloso (*Fuzzy*) e os Algoritmos de

Agrupamento Hierárquico. Nas próximas subseções será apresentada uma breve descrição sobre essas técnicas de *clustering*.

2.3.2.2.1 Clustering particional ou baseados em distância

Os algoritmos de agrupamento particional dividem o conjunto de dados em ***k*** grupos. Para isso, eles primeiramente escolhem ***k*** objetos como sendo os centros dos ***k*** grupos e os elementos são então divididos entre os ***k*** *clusters,* de acordo com a medida de similaridade adotada, "de modo que cada objeto fique no grupo que forneça menor valor de distância entre o objeto e o centro do referido grupo. Os algoritmos utilizam então uma estratégica iterativa, que determina se os objetos devem mudar de grupo, fazendo com que cada grupo contenha somente os elementos mais similares entre si." (GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 98).

Nesse sentido, as técnicas de agrupamento particional criam partições baseadas em pontos de dados. Fundamentam-se na ideia de que um ponto central pode representar um *cluster*. Normalmente, os métodos por particionamento diferem entre si pela forma que é constituída a melhor partição. Segundo Goldschmidt, Passos e Bezerra (2015, p. 96), os algoritmos baseados em distâncias supõem-se a existência de ***n*** pontos de dados $\mathbf{x}_1$, $\mathbf{x}_2$, ..., $\mathbf{x}_{n,}$, de modo que cada ponto pertença a um espaço *d* dimensiona *l* $\boldsymbol{R}^d$. A tarefa de agrupamento desses pontos de dados, preparando-os em ***k*** grupos, consiste em encontrar ***k*** pontos $\boldsymbol{m}_j$ em $\boldsymbol{R}^d$ de tal forma que o valor da expressão a seguir seja minimizado.

$$\frac{\sum_i \min_j \delta(x_i, m_j)}{N} \tag{9}$$

Na fórmula acima, $\boldsymbol{\delta}$ ($\boldsymbol{x}_i$, $\boldsymbol{m}_j$) denota a distância entre $\boldsymbol{x}_1$ e $\boldsymbol{m}_j$. Os pontos $\boldsymbol{m}_j$ são denominados centroides ou médias dos grupos. (GOLDSCHMIDT; PASSOS; BEZERRA, 2015).

Alguns exemplos de algoritmos de agrupamento baseados em distância são: *K-Means, X-Means, K-Modes, K-Medoid* e *Bisicting k-means*.

- K-means

O *k-means* é o mais conhecido algoritmo de agrupamento particional. É um procedimento que usa a noção de centroide, que é o ponto médio de um grupo de pontos.

Liu (2007) afirma que o *k-means* talvez seja o algoritmo mais utilizado dentre todos os métodos de agrupamento devido a sua simplicidade e eficácia, além de ser eficiente na clusterização de grandes conjuntos de dados. É um dos algoritmos de aprendizado não supervisionado mais simples e conhecido.

Por sua vez, o *k-means* divide um conjunto de objetos, com base em seus atributos, em k *clusters*, onde k é uma constante predefinida ou definida pelo usuário. Logo, dado um número fixo de k, esse algoritmo cria um conjunto de k grupos e distribui a coleção de documentos em aglomerados usando a similaridade entre os vetores do documento e os centroides dos *clusters*. Um centroide é o vetor médio de todos os vetores do documento no respectivo grupo (LOPES, 2004).

Na visão dos autores Goldschmidt, Passos e Bezerra (2015, p. 125),

> o *k-means* seleciona k pontos do conjunto de dados. Esses pontos são denominados sementes. Essas sementes são os representantes iniciais, ou centroides, dos k grupos a ser formados. Em seguida, para cada ponto (ou registro do conjunto de dados), calcula-se a distância deste ponto a cada um dos centroides. Atribui-se este ponto ao grupo representado pelo centroide cuja distância é a menor entre todas as calculadas. O resultado desse passo inicial é que cada ponto do conjunto de dados fica associado a um e apenas um dos k grupos. Após a alocação inicial, o método segue iterativamente, por meio da atualização dos centroides de cada grupo e da realocação dos pontos ao centroide mais próximo. O novo centroide de cada grupo G é calculado pela média dos pontos alocados a G. O processo iterativo termina quando os centroides dos grupos param de se modificar, ou após um número preestabelecido de iteração ter sido realizado.

Apesar do *k-means* apresentar um ótimo desempenho, ele apresenta uma desvantagem em relação ao número de grupos, visto que a constante k precisa ser especificada com antecedência, como é o caso de todos os outros modelos de agrupamento baseados em centroide. Mas, mesmo assim, ele é o algoritmo de *clustering* mais popular, pois é frequentemente usado devido à sua facilidade de uso, bem como ao fato dele ser escalável com grandes quantidades de dados.

O algoritmo *k-means* pode ser descrito da seguinte maneira (LINDEN, 2009, p. 24):

1. Escolher k distintos valores para centros dos grupos (de forma aleatória ou usando K-means++)

2. Associar cada ponto ao centro mais próximo

3. Recalcular o centro de cada grupo

4. Repetir os passos 2-3 até nenhum elemento mudar de grupo.

Para encontrar os centros iniciais dos *clusters* e determinar o modo como o algoritmo será inicializado, o *k-means* pode usar a abordagem aleatória ou a abordagem *k-means++*. Na aleatória, os centros iniciais dos grupos são escolhidos aleatoriamente, ou seja, os centroides iniciais serão gerados de forma totalmente aleatória sem um critério para seleção

e na *k-means*++, de acordo com o Guia do desenvolvedor do *Amazon SageMaker*[3], a escolha é realizada da seguinte maneira:

1. Comece com um *cluster* e determine seu centro. Selecione aleatoriamente uma observação do seu conjunto de dados de treinamento e use o ponto correspondente à observação como centro do grupo.
2. Determine o centro do *cluster 2*. Dentre as demais observações do conjunto de dados de treinamento, escolha uma aleatoriamente. Escolha uma que seja diferente da selecionada anteriormente. Essa observação corresponde a um ponto que está distante do centro do grupo 1. Utilizando a coleção de notícias como exemplo, faça o seguinte:
    - Para cada uma das notícias restantes, encontre a distância do ponto correspondente a partir do centro do *cluster 1*. Eleve a distância ao quadrado e atribua uma probabilidade que seja proporcional a esse resultado. Dessa forma, uma notícia diferente da escolhida anteriormente terá mais probabilidade de ser selecionada como centro do *cluster 2*.
    - Escolha uma das notícias aleatoriamente, com base nas probabilidades atribuídas na etapa anterior. O ponto que corresponde à notícia é o centro do *cluster* 2.
3. Repita a etapa 2 para localizar o centro do *cluster 3*. Dessa vez, encontre as distâncias das notícias restantes a partir do centro do *cluster 2*.
4. Repita o processo até que você tenha os centros de k *clusters*.

Em suma, o *k-means* “é extremamente veloz, geralmente convergindo em poucas iterações para uma configuração estável, na qual nenhum elemento está designado para um *cluster* cujo centro não lhe seja o mais próximo.” (LINDEN, 2009, p. 24). Por esse motivo, os testes para esta pesquisa foram, primeiramente, baseados no algoritmo *k-means*, pelo fato dele já ter sido avaliado, implementado e testado em diversas pesquisas usando documentos em outras línguas.

- X-means

O algoritmo x-*means* é uma expansão do algoritmo *k-means*, com a inovação de determinar o número ótimo de *clusters*, sem necessidade de informar o número de k, conforme acontece no *k-means*. “Esse algoritmo deriva do conhecido *k-means*, a diferença é que o *x-means* procura descobrir o número de grupos dentro de um intervalo entre [2, k], onde k é o número máximo de grupos dado pelo usuário como entrada na execução.” (AFONSO, 2013, p. 59).

[3]https://docs.aws.amazon.com/pt_br/sagemaker/latest/dg/algo-kmeans-tech-notes.html

Como a desvantagem do *k-means* está relacionada à necessidade de informar o número de k com antecedência, e esse problema foi resolvido no *x-means*, é importante testar esse algoritmo com um corpus em português para verificar o seu desempenho.

- K-medoid

O algoritmo *k-medoid* usa o conceito de medoide, que é o ponto mais representativo (central) de um conjunto de pontos. Por sua definição, é necessário que um medoide seja um ponto de dados real. O objetivo do *k-medoid* é encontrar um conjunto de *clusters* que não se sobrepõem, de modo que cada grupo tenha um ponto mais representativo, ou seja, um ponto que esteja mais centralmente em relação a alguma medida, como por exemplo, a distância. Esses pontos representativos são chamados de medoides (KAUFMAN; ROUSSEEUW, 1990).

Para Goldschmidt, Passos e Bezerra (2015, p. 128), o algoritmo *k-medoid* concentra-se, primeiramente, em encontrar o *medoid* (o objeto mais centralmente localizado em um grupo). Os elementos restantes são então agrupados com o *medoid* e por um não *medoid*, visando à melhoria do agrupamento. A qualidade é estimada usando uma função que mede a similaridade média entre os objetos e *o medoid* de seu *cluster*.

Segundo Doni (2004), o método *k-medoid* utiliza o valor médio dos elementos em um grupo com um ponto de referência chamado medoide. Esse é o elemento mais centralmente localizado em um grupo. Para esse autor, a estratégia básica é encontrar K grupos em N elementos e, arbitrariamente, descobrir um elemento representativo (medoides) para cada *cluster*. Cada objeto restante é agrupado com o medoide ao qual ele é mais similar. A estratégia, então, iterativamente, troca um dos medoides por um dos não medoides enquanto a qualidade do agrupamento resultante é melhorada.

Para França (2012, p.13), “a diferença básica desse algoritmo em relação ao *k-means* está na utilização de uma das observações do conjunto original como elemento representativo, chamado medoide, localizado no centro do *cluster*, ao invés da tradicional escolha do centro do grupo”. França afirma que, assim como o *k-means*, esse método é bem adequado na hipótese dos grupos serem esféricos, ocupando cada medoide, uma observação mais central do grupo.

- Bisecting k-means

Bisecting k-means, segundo JinHuaXu e HongLiu (2010), é uma combinação do *k-means* e o agrupamento hierárquico. Porém, em vez de particionar os dados em *k clusters* em cada iteração, *esse algoritmo* divide um *cluste*r em dois subgrupos em cada etapa até que os agrupamentos *k* sejam obtidos. Como o *bisecting k-means* é baseado no *k-means*, ele

mantém os méritos deste algoritmo e possui a vantagem de ser mais eficiente quando *k* é grande. Além disso, no algoritmo *k-means*, o cálculo envolve todos os pontos de dados do conjunto e todos os k centroides, já o *bisecting k-means*, em cada etapa, apenas os pontos de dados de um *cluster* e dois centroides estão envolvidos na computação. Assim, o tempo de processamento é reduzido.

2.3.2.2.2 Agrupamento probabilístico ou baseado em distribuição

Os algoritmos de agrupamento probabilístico utilizam-se a abordagem probabilística.

> Os algoritmos baseados em distribuições consideram que o conjunto de dados de entrada foi gerado por uma ou mais distribuições de probabilidades (i.e., por uma mistura de distribuições), consideradas desconhecidas, mas fixas. A partir disso, esses algoritmos tentam identificar os parâmetros dessas distribuições de probabilidades subjacentes. (GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 97).

De acordo com McLachlan e Basford (1988), na abordagem de agrupamento probabilístico, os dados são considerados uma amostra independente extraída de um modelo de misturas de várias distribuições de probabilidade. Um exemplo de algoritmo probabilístico é o *Expectation Maximization* (EM).

McLachlan e Krishnan (2007) definem o *Expectation Maximization* como um método de agrupamento não supervisionado que é baseado em cálculos estatísticos. Pode ser considerado como uma expansão do *k-means*, pois atribui os objetos ao grupo que é mais similar, com base na média do grupo. Porém, em vez de atribuir cada objeto a um grupo exclusivo, o *EM* distribui os objetos para os aglomerados de acordo com um peso que representa a probabilidade do elemento pertencer a tais grupos.

Na visão dos autores Goldschmidt, Passos e Bezerra (2015, p. 97),

> O EM considera que os pontos são provenientes de uma mistura de distribuições gaussianas cujos valores de média e de variância são desconhecidos. Além disso, também são desconhecidas as probabilidades de cada objeto ser proveniente de alguma das distribuições. Inicialmente esses valores são definidos de forma aleatória. A seguir o algoritmo passa a executar iterativamente dois passos, **E** e **M**. No passo **E** (*expectation*), a verossimilhança do conjunto de dados como um todo é calculada. No passo **M** (*maximization*), os valores desconhecidos são recalculados por meio da maximização da função do passo anterior. A execução desses dois passos continua até que algum critério de convergência seja satisfeito.

Por conseguinte, o algoritmo *ME* é usado, principalmente, quando os dados estão incompletos e sua principal aplicabilidade está relacionada com a busca por solução de problemas referentes a *clustering* e reconhecimento de padrões e em outras aplicações em que técnicas estatísticas são utilizadas.

2.3.2.2.3 Propagação por Afinidade (Affinity Propagation)

O algoritmo *Affinity Propagation* (AP) foi publicado por Frey e Dueck em 2007 e tem se tornado cada vez mais popular devido à sua simplicidade, aplicabilidade e desempenho (ILIASSICH, *on-line*). É um algoritmo de *clustering* relativamente novo, baseado no conceito de "passagem de mensagens" entre pontos de dados. O AP não requer que o número de *clusters* seja determinado ou estimado antes da sua execução. Ele toma a semelhança entre pares de pontos de dados como medida de entrada e simultaneamente considera todos os itens como potenciais exemplares. Assim, as mensagens são trocadas entre os pontos até que um conjunto de alta qualidade de *clusters* correspondentes, gradualmente, emerge (DAS, 2017).

De acordo com Lima (2017, p. 18),

> A técnica é responsável por selecionar um determinado número de *clusters* de acordo com a base de dados existente. O *Affinity Propagation* usa como conjunto de dados as principais semelhanças entre os mesmos, onde as semelhanças s (i, k) indicam quão adequado são os dados de k para cada ponto de i. Quando o objetivo é minimizar os erros quadrados, cada similaridade é estabelecida como sendo o inverso do erro quadrado (distância euclidiana). Sendo s (k, k) um número real, se pode inferir que para cada ponto k, seus pontos serão escolhidos como pontos principais. Estes pontos são denominados pontos exemplares. O número de pontos exemplares será o número de *clusters*, influenciados pelos valores de entrada exemplares. Em princípio, se sugere que todos os dados possam ser eleitos como tais, mas este ponto pode ser transformado para produzir o número de *clusters*. O valor compartilhado pode ser a mediana (caso se trate de um número moderado de *clusters*), ou seus mínimos (se for o resultado de um número pequeno de *clusters*). O algoritmo funciona como se segue: dado o conjunto de dados bidimensionais, em que a Distância Euclidiana é utilizada como medida de similaridade. Cada cor é estipulada ao ponto dependendo da evidência de ser o centro do *cluster*, e as distâncias entre um ponto i e um ponto k são medidas mediante a força que pode ser transmitida entre si. A responsabilidade r (i, k) é enviada entre pontos, indicando qual é o ponto forte em relação a um outro ponto exemplar. A disponibilidade a (i, k) é enviada a partir dos candidatos aos pontos para indicar o grau em que estes podem ser o centro do *cluster*. Em seguida, é mostrado o efeito do valor da preferência de entrada (comum para todos os pontos de dados) no número de exemplares identificados (número de grupos).

De acordo com DAS (2017), o AP define a afinidade dos pontos de dados alternando duas etapas de passagem de mensagem para atualizar duas matrizes: a matriz de responsabilidade (R) e a matriz de disponibilidade (A). Desse modo, o algoritmo cria os *clusters* enviando mensagens entre pares de amostras e um conjunto de dados é então descrito utilizando um pequeno número de exemplares, que são identificados como os mais representativos. As mensagens enviadas entre pares representam a adequação para uma amostra ser o exemplar da outra, que é atualizada em resposta aos valores de outros pares. Esta atualização acontece iterativamente até a convergência, que é o número de iterações

sem alteração no número de *clusters* estimados. Para completar o processo, os exemplares finais são escolhidos e, consequentemente, o agrupamento é criado.

O *Affinity Propagation* pode ser interessante, pois ele escolhe o número de *clusters* com base nos dados fornecidos. Para este efeito, os dois parâmetros importantes são a preferência, que controla quantos exemplares são utilizados, e o fator de amortecimento (*Damping factor*). De acordo com o site *Scikit Learn*[4]*,* o fator de amortecimento (varia entre 0,5 e 1) é a medida que o valor atual é mantido em relação aos valores de entrada (ponderado 1 - amortecimento). Isso para evitar oscilações numéricas ao atualizar esses valores (mensagens).

2.3.2.2.4 Clustering Hierárquico

Os algoritmos de agrupamento hierárquico receberam esse nome porque eles criam grupos que podem ser representados na forma hierárquica. A hierarquia pode ser construída de forma descendente ou de baixo para cima (aglomeração). A abordagem de cima para baixo começa com um *cluster* que inclui todos os documentos e recursivamente os grupos são divididos em subgrupos. Na abordagem aglomerativa, cada documento é inicialmente considerado como um *cluster* individual. Então, sucessivamente, os *clusters* mais parecidos são reunidos até que todos os documentos sejam agrupados.

De acordo com Afonso (2013), os algoritmos de *clustering* hierárquico particionam os documentos em aglomerados e o objetivo é que “os grupos cada vez mais tenham características diferentes entre si, ou seja, gerando o agrupamento por dissimilaridade entre os grupos. Neste caso, não há formação de hierarquias, há apenas um nível hierárquico.” (AFONSO, 2013, p. 54).

Ademais, os algoritmos de agrupamento hierárquicos são baseados na distância, ou seja, usam uma função de similaridade para medir a proximidade entre os textos. Eles também são conhecidos como métodos de agrupamento baseados em conectividade e fundamentam-se no conceito de que objetos similares estarão mais próximos dos objetos relacionados no espaço vetorial do que os não relacionados. Nesse modelo, os *clusters* são formados conectando os elementos com base em suas distâncias e podem ser visualizados usando um dendrograma (SARKAR, 2016).

Portanto, esses algoritmos são úteis para organizar documentos hierarquicamente, por exemplo, cada tópico pode conter subtópicos e assim sucessivamente. Segundo Liu (2007) a hierarquia de tópicos é particularmente útil na organização de textos.

4 https://scikit-learn.org

Segundo Allahyari *et al*. (2017), existem três métodos de agrupamento hierárquico: *Single Linkage*, *Average Linkage* e *Complete Linkage*. Porém, também será utilizado nos experimentos desta pesquisa o método *Ward* (*Ward's method*). Portanto, uma descrição mais detalhada desses algoritmos será apresentada:

- *Single Linkage* (ligação mínima)

No *Single Linkage* a distância entre dois *clusters* é calculada entre os pontos mais próximos, também chamado de "agrupamento de vizinhos". Nesse modelo, a divisão hierárquica começa a partir da matriz de similaridades entre objetos e em seguida os pares de elementos são ordenados conforme sua similitude de modo descendente, formando, desta forma, os *clusters* hierárquicos, que começam com os pares mais parecidos. A cada etapa, aglomera-se uma observação adicional para formar novos *clusters*. Desta forma, o primeiro grupo é aquele com duas observações que apresentam a menor distância entre si. (LOPES, 2004).

Em suma, no agrupamento por ligação mínima, segundo Faceli *et al*. (2017), a distância entre *clusters* é dada pela distância entre os objetos dos dois aglomerados que estão mais próximos, ou seja, é a distância mínima entre quaisquer dois objetos, um de cada grupo.

- *Average Linkage* (ligação média)

O método *Average Linkage*, inicialmente, comporta conforme o método de ligação simples, ou seja, começa agrupando os dois objetos mais semelhantes e, em seguida, utiliza a média aritmética das distâncias dos indivíduos de cada grupo. Desta forma, a distância entre os *clusters* é calculada entre os centroides. Ou seja, a semelhança entre dois grupos é a similaridade média entre pares de documentos nesses aglomerados.

- *Complete Linkage* (ligação máxima)

No *Complete Linkage*, a distância entre *clusters* é calculada entre os pontos mais distantes. Ou seja, a similaridade entre dois grupos é a semelhança do pior caso entre qualquer par de documentos nesses aglomerados.

Desse modo, o algoritmo de ligação máxima, após agrupar os dois objetos mais similares, de menor distância, verifica a distância máxima do primeiro grupo com os elementos restantes. Dessa forma, procura-se garantir que os indivíduos de um conjunto guardem a máxima distância dos outros grupos.

- Ward's method (método Ward)

No método *Ward*, também denominado método da mínima variância, a medida da distância entre dois *clusters* é a soma das distâncias ao quadrado entre os dois grupos. Essa técnica busca maximizar a verossimilhança em cada nível de hierarquia sob as hipóteses de mistura de normas multivariadas, matrizes esféricas de covariância iguais e probabilidades amostrais idênticas. Tende a unir *clusters* com número pequeno de observações e é fortemente enviesado na direção de produzir grupos com mesmo formato e número de observações (CARVALHO *et al.* 2009).

Segundo Hair *et al*. (2005), o método *Ward* consiste em um procedimento de agrupamento hierárquico no qual a medida de similaridade usada para fazer a junção dos grupos é calculada como a soma de quadrados entre os dois *clusters* feita sobre todas as variáveis. Esse método tende a resultar em aglomerados de tamanhos aproximadamente iguais devido a sua minimização de variação interna. Em cada estágio, combinam-se os dois agrupamentos que apresentarem menor aumento na soma global de quadrados dentro dos *clusters*.

Por conseguinte, para que os *clusters* sejam formados, é necessário medir a similaridade entre os elementos da amostra para que os grupos de objetos semelhantes sejam formados. Assim, devido à importância do tema, as medidas usadas para o cálculo da distância serão descritas na próxima seção.

2.3.2.3 Medida da Similaridade e identificação de *clusters*

Segundo Ticom (2007), nos procedimentos de Mineração de Textos sempre são utilizadas medidas matemáticas. "Elas podem servir para avaliar a distância entre dois vetores, ou ainda quando se deseja atribuir pesos às palavras mais relevantes de um texto, e principalmente na mensuração do desempenho das técnicas de MT." (TICOM, 2007, p. 17). Por isso, serão descritas algumas das principais medidas de avaliação existentes nesta seção.

Um conceito importante para os algoritmos de Mineração de Dados ou Textos é a noção de similaridade.

> Uma vez que o conjunto de dados pode ser interpretado como um conjunto de pontos em um espaço k-dimensional, o conceito de similaridade entre dois pontos pode ser traduzido como a distância entre estes pontos. Quanto menor a distância entre dois pontos, maior a similaridade entre os registros por eles representados. (GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 71).

Assim, o conceito de similaridade é fundamental para a construção de um *cluster*, pois se dois padrões são similares de acordo com algum critério usado pela técnica de agrupamento, então eles serão agrupados em um mesmo aglomerado, caso contrário, serão alocados em grupos diferentes. Por isso a definição de medidas que permitam comparar padrões pertencentes a um mesmo espaço de características é essencial para a maioria dos processos de *clustering* (LACHI; ROCHA, 2005).

Nesse contexto, o principal objetivo da similaridade de texto é analisar e medir como duas entidades de documentos estão próximas ou distantes umas das outras. Roses (2002) afirma que a distância medida entre pontos de cada objeto equivale ao nível ou grau de similaridade entre eles. Esta medida é utilizada para identificar em qual *cluster* um elemento deve ser alocado.

Desse modo, a similaridade pode ser calculada entre termos e entre os documentos. Na similaridade de termos é medida a semelhança entre *tokens* individuais ou palavras e na similaridade de documentos é medida a semelhança entre documentos de textos inteiros. O objetivo desta etapa é implementar e usar várias métricas de distância para medir e analisar a semelhança entre os textos.

São exemplos de métodos para calcular a semelhança entre os textos: Distância Euclidiana, Distância Manhattan, Distância de Minkowski, Similaridade de Cosseno e Coeficiente de Correlação de Person.

2.3.2.3.1 Distância Euclidiana

A Distância Euclidiana é, provavelmente, a medida de distância mais conhecida e mais usada. Ela simplesmente é a distância geométrica no espaço multidimensional. É a distância comum entre dois pontos (HUANG, 2008), ou seja, a distância entre dois pontos em um plano bidimensional ou dois pontos em um espaço tridimensional. No caso do agrupamento de texto, geralmente são mais de três dimensões, uma vez que o número de dimensões é igual ao número de palavras diferentes no corpus.

Segundo Goldschmidt, Passos e Bezerra (2015), o conceito da distância é formalizado da seguinte maneira: Sejam ***E*** o conjunto de pontos que representa o conjunto de dados e ***x***, ***y*** e ***z*** elementos quaisquer de ***E***. A distância entre um par de pontos de ***E*** é uma função ***d: E*** x ***E*** → (i.e., a cada par de pontos, a função ***d*** associa um valor real) que atende às seguintes restrições:

(i) $d(\mathbf{x}, \mathbf{x}) = 0$

(ii) $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$

(iii) $d(\mathbf{x}, \mathbf{y}) \leq d(\boldsymbol{x}, \mathbf{z}) + d(\mathbf{z}, \mathbf{y})$

Assim, “considere que $\mathbf{x_i}$ e $\mathbf{y_i}$ são os componentes (coordenadas) dos elementos (vetores, pontos) **x** e **y**, respectivamente. Considere também que $\boldsymbol{n}$ é a quantidade de componentes (coordenadas) de cada ponto”. Sendo assim, a equação 10 apresenta a função da Distância Euclidiana, “que é a distância entre dois pontos medida pela linha reta que liga esses dois pontos.” (GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 72).

$$\boldsymbol{d(x,y) = \sqrt{\sum_{i=1}^{n} (xi - yi)^2}} \tag{10}$$

Conforme mostrado na expressão 10, os parâmetros x_i e y_i são subtraídos diretamente um do outro. Segundo Lopes (2004) a distância Euclidiana leva em consideração a magnitude das diferenças dos valores dos dados. Logo, ela preserva mais informação sobre os dados e pode ser preferível. Todavia, embora seja uma medida comum para dados convencionais, esta métrica e suas variações não são consideradas propícias para dados textuais, apresentando resultados pobres quando comparadas a outras métricas.

2.3.2.3.2 Distância Manhattan

A Distância Manhattan, também conhecida como distância City-Block, é a distância entre dois pontos medida ao longo dos eixos coordenados em ângulos retos (GOLDSCHMIDT; PASSOS; BEZERRA, 2015). “É uma métrica que, como a distância Euclidiana, possui parâmetros que são subtraídos uns dos outros diretamente e deve-se ter o cuidado de que os parâmetros sejam normalizados.” (LOPES, 2004, p. 94).

Na expressão (11), a Distância Manhattan é o operador de módulo (ou valor absoluto). (GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 72):

$$\boldsymbol{d(x,y) = \sqrt{\sum_{i=1}^{n} |x_i - y_i|}} \tag{11}$$

Assim como a Distância Euclidiana corresponde ao comprimento do caminho mais curto entre dois pontos, a Distância Manhattan é a soma das distâncias ao longo de cada dimensão. Segundo Kugler, Tortato Júnior e Lopes (2003), a Distância de Manhattan é uma simplificação da Distância Euclidiana e, por isso, é uma medida simples e fácil de implementar. Segundo esses autores, essa medida é mais eficiente para aplicações em tempo real devida a sua simplicidade.

2.3.2.3.3 Distância de Minkowski

A Distância de Minkowski é uma generalização da distância Euclidiana. “É a distância generalizada entre dois pontos” (GOLDSCHMIDT; PASSOS; BEZERRA, 2015, p. 72). Se valor é calculado pela seguinte equação:

$$dist = (\sum_{k=1}^{n} | p_k - q_k |^r)^{\frac{1}{r}} \quad (12)$$

Em que *r* é um parâmetro, *n* é o número de dimensões (atributos) e p_k e q_k são, respectivamente, os k-ésimos atributos (componentes) dos objetos de dados *p* e *q*.

2.3.2.3.4 Similaridade de Cosseno

Segundo Klinczak (2016), a medida de Similaridade do Cosseno é a mais empregada para aplicações que envolvem o tratamento de textos e sua função é mostrar a similaridade entre dois documentos através do ângulo formado. Ao usar essa medida, assume-se que a direção dos vetores de documentos é mais importante do que o comprimento e a distância entre eles (ZHONG, 2005).

A Similaridade de Cosseno é uma das medidas mais comumente utilizadas para agrupamento de texto e demonstrou ser a melhor medida para agrupar dados de alta dimensão (HUANG, 2008). Esse método envolve, como o nome revela, o cosseno do ângulo entre os dois vetores do termo e tem grande utilização em medidas de documentos.

> Se existirem dois vetores, a medida do cosseno entre estes dois vetores será um menos o cosseno do ângulo formado entre eles. A medida do cosseno será grande (perto de um) se os vetores forem quase ortogonais (este caso significa que existem poucas palavras comuns entre os documentos), e pequena (perto de zero) se os vetores forem similares (grande quantidade de palavras comuns a ambos). (TICOM, 2007, p. 18).

A expressão do cosseno para avaliar a similaridade entre dois documentos pode ser escrita pela seguinte equação (FULLAM, 2002 apud TICOM, 2007):

$$M_{cos} = \frac{\sum_{k=1}^{j}(d1_k * d2_k}{\sqrt{v_{d1} * v_{d2}}} \quad (13)$$

Onde: *d1* e *d2* são documentos representados por vetores; *j* é igual ao total de termos; e o produto escalar é representado por:

$$v_{d1} = \sum_{k=1}^{i} d1_k^2 \quad (14)$$

Segundo Klinczak (2016), a diferença entre a Similaridade do Cosseno e a Distância Euclidiana é que a Distância Euclidiana é uma função de distância métrica em que os valores são menores e mais próximos aos vetores. Por outro lado, a Similaridade do Cosseno não é uma função de distância, sendo definida como um intervalo fixo entre 0 e 1. Pode-se ter o mesmo resultado da Similaridade do Cosseno usando a Distância Euclidiana, desde que esta esteja com os vetores normalizados e contendo apenas valores positivos.

2.3.2.3.5 Coeficiente de correlação de Pearson

Para Ticom (2007), dada duas amostras de observações medidas em uma escala de intervalos ou razões, pode-se mensurar o grau de associação linear entre elas por intermédio do Coeficiente de Correlação de Pearson ou apenas Coeficiente de Correlação Amostral. “Assumindo que ambas variáveis (X e Y) são intervalos entre variáveis, as mesmas são bem aproximadas por uma distribuição normal como também sua distribuição conjunta é normal bivariada.” (TICOM, 2007, p. 18).

O coeficiente de Pearson é dado pela expressão (BOLBOACĂ, 2006 apud TICOM, 2007):

$$C_{Pea} = \frac{\sum_{k=1}^{j}(d1_k - \overline{d1})(d2_k - \overline{d2})}{\sqrt{(\sum_{k=1}^{j}(d1_k - \overline{d1})^2(\sum_{k=1}^{j}(d2_k - \overline{d2})^2)}} \quad (15)$$

Onde $\overline{d1}$ e $\overline{d2}$ são iguais à média da amostra de d1 e d2.

> Este coeficiente de correlação pode variar entre -1 e 1. Ele assume o valor 1 quando os pontos estão exatamente sobre uma reta em declive positivo. Neste caso, um aumento em uma das variáveis corresponde necessariamente a um aumento na outra. R assume o valor -1 quando os pontos estão exatamente sobre uma reta de declive negativo. Nesta situação, um aumento em uma das variáveis corresponde a uma diminuição na outra. Estes dois casos correspondem ao máximo de associação linear, que é possível observar entre duas amostras. Quando as amostras são independentes, o valor do coeficiente será próximo de zero ou mesmo zero. Uma interpretação usual do coeficiente de correlação amostral passa por considerar o seu valor elevado ao quadrado, R^2, a que se chama coeficiente de determinação. Uma vez que $-1 \leq R \leq 1$, o coeficiente de determinação está sempre entre 0 e 1. Resumindo, o coeficiente de correlação de Pearson mede o grau de associação linear entre duas variáveis medidas em uma escala de intervalos ou razões. Se as variáveis tiverem distribuição Normal podemos efetuar um teste de hipóteses para averiguar se o coeficiente de correlação da população é significativamente diferente de zero, o que significará, nesse contexto, que as variáveis são independentes. Convém sempre construir um diagrama de dispersão para ter uma ideia sobre a linearidade da relação entre as variáveis. (TICOM, 2007, p. 19).

Em síntese, O Coeficiente de Correlação de Pearson é uma medida do grau de relação linear entre duas variáveis quantitativas. Esse coeficiente varia entre os valores -1 e 1. O valor

0 (zero) significa que não há relação linear, o valor 1 indica uma relação linear perfeita e o valor -1 também indica uma relação linear perfeita, mas inversa, ou seja, quando uma das variáveis aumenta a outra diminui. Quanto mais próximo estiver de 1 ou -1, mais forte é a associação linear entre as duas variáveis.

2.3.2.4 *Topic Modeling*

A área de estudo relacionada com a Mineração de Texto é subdividida em diversos tipos de análises. Uma delas é a Modelagem de Tópicos cujo principal interesse é descobrir quais tópicos são os mais recorrentes em uma coleção de diferentes documentos.

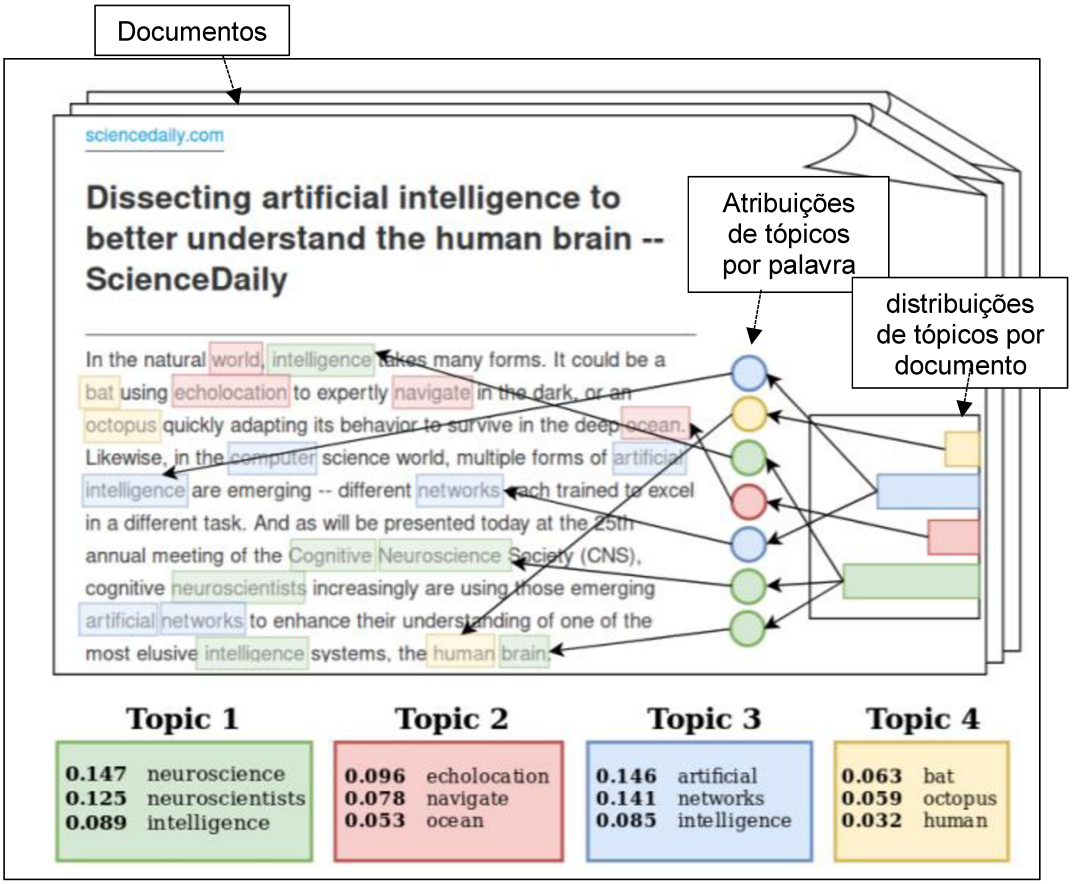
A Modelagem de Tópicos, em inglês *Topic Modeling*, é um campo de estudo que surgiu em 2003 e que também é chamada de Modelos Probabilísticos de Tópicos (*Probabilistic Topic Models*) (BLEI, 2011). Pode ser definida como uma técnica não supervisionada de *Machine Learning* que busca identificar grupos de palavras que ocorrem juntas e, assim, descobrir tópicos abstratos que incidem em um corpus. O objetivo é descobrir subestruturas semânticas dentro de uma coleção de textos (Lance, 2017).

Na opinião de Chang (2016), os algoritmos de Modelagem de tópicos são um conjunto de métodos de Aprendizado de Máquina que facilita a revelação de estruturas temáticas ocultas em grandes coleções de textos.

> A exploração de grandes volumes de dados é simplificada pelos modelos probabilísticos na descoberta dos tópicos. Os tópicos são estruturas com valor semântico e que, no contexto de mineração de texto, formam grupos de palavras que frequentemente ocorrem juntas. Esses grupos de palavras quando analisados, dão indícios a um tema ou assunto que ocorre em um subconjunto de documentos. A expressão tópico é usada levando-se em conta que o assunto tratado em uma coleção de documentos é extraído automaticamente, ou seja, tópico é definido como um conjunto de palavras que frequentemente ocorrem em documentos semanticamente relacionados. (FALEIROS; LOPES, 2016, p. 9)

A Figura 17 ilustra o processo de extração de tópicos derivados de uma coleção de texto.

FIGURA 17: Modelagem de Tópicos



Fonte: Silveira (2018, p. 9)

Observa-se à direita da Figura 17 uma representação da distribuição de tópicos em cada documento que é responsável por determinar o grau de ocorrência do tópico em um determinado texto. À esquerda está a representação da distribuição de palavras geradas a partir da distribuição de tópicos. Na parte inferior da imagem está o ranqueamento de tópicos a partir da distribuição de palavras, contendo os termos com maior probabilidade de ocorrer na coleção dado um determinado tópico (SILVEIRA, 2018).

A Modelagem de Tópicos geralmente envolve o uso de técnicas matemáticas e estatísticas para extrair tópicos, temas ou conceitos de uma coleção de documentos. Usa técnicas estatísticas específicas incluindo *Latente Semantic Indexing* (LSI) e *Latent Dirichlet Allocation* (LDA) para descobrir estruturas semânticas latentes (desconhecidas) conectadas em dados textuais para, assim, gerar os tópicos. (Sarkar, 2019).

A Indexação Semântica Latente foi desenvolvida na década de 1970 como um método estatístico usado para correlacionar termos de corpora vinculados semanticamente. É uma tecnologia usada não somente na sumarização de texto, mas também na recuperação e pesquisa de informações (SARKAR, 2019).

Segundo Sarkar (2019), o principal princípio por trás da LSI é que termos similares tendem a ser usados nos mesmos contextos e, portanto, tendem a coocorrerem mais. O termo LSI deriva do fato de que essa técnica tem a capacidade de descobrir termos ocultos e desconhecidos que se correlacionam semanticamente para formar tópicos. Lance (2017) afirma que a Modelagem de Tópicos pode usar a LSI, mas essa técnica não é considerada um modelo de tópico autêntico, pois não é um modelo probabilístico. Uma inovação dessa técnica é a Análise Probabilística de Semântica Latente, em inglês *Probabilistic Latent Semantic Analysis* (PLSA). Uma extensão do PLSA é a Alocação de Dirichlet Latente (LDA) que é uma técnica muito usada em Modelagem de tópicos.

Lance (2017) define o LDA como um modelo probabilístico generativo popular que permite a análise de grandes conjuntos de dados semânticos pelos pesquisadores. “Um modelo generativo é aquele que aleatoriamente gera os dados a partir das variáveis latentes”. (FALEIROS; LOPES, 2016, p. 13). Para Rader e Wash (2015), LDA é uma técnica de *Bag-of-Words* na qual as palavras são consideradas independentes, ou seja, a ordem dos termos não é relevante. No entanto, o algoritmo analisa as frequências e a coocorrência dos termos em um documento e o quanto ele é comum nos outros documentos do corpus.

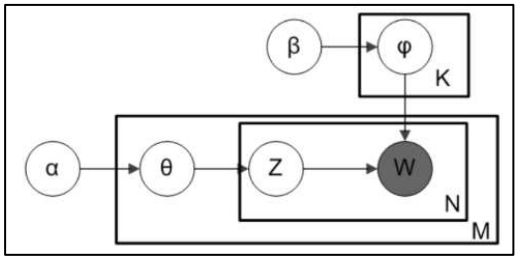
Faleiros e Lopes (2016) afirmam que o LDA não é um algoritmo com descrições sequenciais de instruções para encontrar tópicos a partir de uma coleção de textos. É um modelo probabilístico no qual é descrito como os documentos são gerados. Nessa técnica, as variáveis observáveis são os termos de cada documento e as variáveis não observáveis são as distribuições de tópicos cujos parâmetros são dados a priori no modelo.

> A distribuição utilizada para amostrar a repartição de tópicos é a distribuição de Dirichlet. No processo generativo, o resultado da amostragem da Dirichlet é usado para alocar as palavras de diferentes tópicos e que preencherão os documentos. Assim, pode-se perceber o significado do nome Latent Dirichlet Allocation, que expressa a intenção do modelo de alocar os tópicos latentes que são distribuídos obedecendo a distribuição de Dirichlet. (FALEIROS, LOPES, 2016, p. 13)

Segundo Silveira (2018), a intuição principal do LDA é que os documentos cobrem um determinado número de tópicos, ao passo que estes são representados por um determinado número de palavras.

Uma notação para o modelo LDA é representada na Figura 18.

FIGURA 18: Modelo LDA



Fonte: Sarkar (2019, p. 389)

De acordo com Sarkar (2019, p, 389)

- k é o número de tópicos;
- N é o número de palavras no documento;
- M é o número de documentos para analisar;
- α - Priore da distribuição de Dirichlet, relacionada a distribuição documento-termo.
- β - Priore da distribuição de Dirichlet, relacionada a distribuição tópico-palavra.
- θ - distribuição de tópicos por documentos.
- φ - distribuição dos tópicos sobre as palavras do vocabulário
- φ (k) é a distribuição de palavras para o tópico k
- θ(i) é a distribuição de tópicos para o documento i
- z(i, j) é a atribuição do tópico para w(i,j)
- w(i,j) é a j-ésimo palavra no i-ésimo documento
- φ e θ são distribuições Dirichlet, z e w são multinominais

O pseudocódigo a seguir apresenta o algoritmo interativo LDA (DOIG, 2015):

Inicializar parâmetros
Inicialize atribuições de tópicos aleatoriamente
Iterar
Para cada palavra em cada documento:
    Reamostrar tópico por palavra, considerando todas as outras palavras e suas atribuições de tópicos atuais
Obter resultados
Avaliar modelo

Nas visões de Steyvers e Griffiths (2007 citado por Silveira, 2018), o modelo de tópicos é um modelo gerador de documentos, onde cada documento pode ser gerado estatisticamente. Segundo os autores, esse processo ocorre da seguinte forma (STEYVERS; GRIFFITHS, 2007 citado por SILVEIRA, 2018)

- Escolhe-se uma distribuição de probabilidade relacionada aos tópicos;

- Para cada palavra presente em um documento qualquer, escolhe-se aleatoriamente um tópico *t* de acordo com a distribuição de probabilidade;
- Amostra-se uma nova palavra da distribuição de tópicos *t.*

Nos dizeres de Silveira (2018), os processos de representação de documentos e palavras como tópicos probabilísticos possuem vantagens quando comparados com representações puramente vetoriais, pois na representação vetorial os termos do documento são representados como vetores em um espaço sem levar em consideração o contexto nos quais estão empregados e sem analisar a relação de dependência com os outros termos.

Em suma, nesta seção foram expostas as definições e os principais aspectos relacionados a Processamento de Linguagem Natural, Mineração de Textos, Aprendizado de Máquina e Clusterização. Além disso, foram destacadas as principais técnicas empregadas no PLN bem como apresentados os processos utilizados para a análise de textos. Ademais, nesse detalhamento também foram expostos aspectos importantes para a realização do agrupamento de textos, tais como técnicas para preparação dos dados e algoritmos de agrupamento tradicionais. Além do mais, apresentou-se uma descrição a respeito das medidas de similaridade. E por fim, narrou sobre Modelagem de Tópicos, apresentando seus principais conceitos e técnicas.

Dentre os recursos de análise de textos apresentados neste capítulo, serão utilizados neste trabalho os métodos de pré-processamento, tais como tokenização, remoção das *stop words*, *stemming* e N-gramas. Além disso, realizará o cálculo de relevância das palavras para obter somente os termos mais significativos do corpus. Essas soluções são essenciais para reduzir a dimensão da coleção de notícias em estudo.

Como a proposta desta pesquisa é trabalhar com *clustering*, escolheram-se alguns algoritmos de aprendizado não supervisionado para agrupar as notícias por similaridade. O *K-Means* foi selecionado por ser o algoritmo de agrupamento particional mais simples e conhecido e por ser eficiente na clusterização de grandes conjuntos de dados. Porém, essa técnica necessita de um tipo especial de parâmetro que é o valor de k, um número que representa a quantidade de *clusters* que as notícias devem ser particionadas. Embora seja de fato um parâmetro dos métodos de agrupamento, nos experimentos da literatura de *clustering*, o número de grupos é geralmente considerado conhecido antecipadamente, pois é baseado nos dados coletados. Como as categorias da coleção de notícias usada neste estudo são desconhecidas, o método *Elbow* será utilizado para encontrar o número ideal de *clusters* em todos os experimentos desta pesquisa.

O método *Elbow* calcula o SSE (*Sum of Squared Error*). Segundo Kodinariya e Makwana (2013) *apud* Heinzen e Martins (2018), o método *Elbow* é uma das mais tradicionais formas usadas para encontrar o número ideal de k de uma determinada amostra. Ele analisa

a porcentagem de variância para encontrar o número de *clusters*. Essa técnica consiste em identificar, em um gráfico, a diferença brusca na variância entre os diferentes números de grupos, indicando que, em determinados casos, a adição de mais *clusters* não interferirá na variância dos dados presentes no conjunto de dados.

Quanto aos algoritmos probabilísticos ou baseados em distribuição, optou-se por utilizar nos experimentos o *Affinity Propagation*, pois, de acordo com a literatura estudada, ele é simples e tem apresentado bom desempenho no agrupamento de textos. Além disso, o AP não requer que o número de *cluster*s seja estimado pelo usuário antes da execução. Em relação aos algoritmos hierárquicos, os quatro métodos (*Single Linkage*, *Average Linkage*, *Complete Linkage*) serão testados. Como a diferença deles está na forma que é realizado o cálculo da distância, não será trabalhoso fazer o experimento com todos eles e, assim, verificar se a maneira de calcular a distância entre dois grupos influi nos resultados.

Para calcular a similaridade entre as notícias, optou-se por usar nos experimentos a Distância Euclidiana, por ser a distância mais conhecida e usada; a Similaridade de Cosseno pelo fato de ser a mais empregada em aplicações de análise de textos; e a Distância de Manhattan, por ser uma medida simples e de fácil implementação.

Por fim, para avaliar o resultado, preferiu usar o Coeficiente da Silhueta para medir a qualidade de toda a estrutura dos agrupamentos, pois essa medida possibilita a visualização gráfica dos agrupamentos e além de ser uma técnica comumente encontrada na literatura, é de fácil implementação no Python, linguagem de programação que será utilizada nos experimentos desta pesquisa.

3 REVISÃO DE ESTADO DA ARTE

O objetivo do estado da arte é realizar uma investigação nos trabalhos existentes no campo em que se deseja pesquisar, buscando identificar e avaliar toda pesquisa relevante relacionada ao tema para, assim, constatar o que esses trabalhos apresentam em comum, além de contribuir para uma compreensão mais abrangente do desenvolvimento da área.

Nesta pesquisa, realizou-se uma busca no Google Scholar, na biblioteca eletrônica Scielo, no Portal Capes e em Bancos de Teses e Dissertações, no período de outubro de 2017 a março de 2018. O Portal Capes facilita o acesso a várias bases de dados, porém, nesta pesquisa optou-se pelas bases LISA, EBSCO, *Web of Science* e *IEEE Xplore*. A Lisa foi escolhida por ser direcionada à profissionais da Biblioteconomia e Ciência da Informação. A Ebsco, por permitir selecionar a área de conhecimento que se deseja pesquisar, além de apresentar boas opções de filtros. A *Web of Science* por ser uma das bases mais recomendadas do mundo, abrangendo as áreas das ciências, ciências sociais, artes e humanidades. A *IEEE Xplore*, além das opções de filtro proporcionada, ela exibe a quantidade de vezes que os artigos recuperados foram citados, o que facilita a identificação das publicações de maior relevância.

Assim, como o objetivo de encontrar trabalhos relacionados com o tema clusterização de notícias da web ou que estivessem dentro do contexto de clusterização de textos, iniciou-se a pesquisa com a base Ebsco, utilizando como critério de inclusão o descritor "*clustering*" com o objetivo de recuperar os artigos publicados nos últimos dez anos. Nesse caso, foi recuperada uma quantidade grande de artigos, o que dificultou a análise. Para tanto, optou-se por realizar novamente a consulta usando no campo de descrição as palavras "*News clustering*". Como resultado dessa busca, a Ebsco retornou apenas seis documentos. Ao realizar uma leitura parcial nos textos recuperados, somente um artigo atendeu os objetivos desta investigação.

Dentro desse contexto, realizou-se uma nova pesquisa utilizando a palavra-chave "*Text Clustering*" e, nesse caso, 111 artigos foram recuperados. Para reduzir a quantidade de documentos, fez-se uma nova restrição na busca, selecionando apenas as publicações dos últimos 5 anos, o que resultou em um total de 65 textos recuperados. Para selecionar as publicações principais, realizou-se uma leitura parcial, dando atenção especial para os artigos em que as palavras-chave estivessem mais relacionadas como o tema desta pesquisa e que foram citados um número maior de vezes.

Realizou-se o mesmo procedimento nas demais fontes de pesquisa, pesquisando os artigos publicados nos últimos dez anos e reduzindo o período para cinco somente nos casos em que as bases retornaram muitos artigos. Com base nos resultados das buscas, observou-

se que existem poucos trabalhos relacionados a agrupamentos de notícias recuperadas das principais mídias de massa. Portanto, foram selecionados trabalhos referentes a agrupamento de texto, principalmente, os focados nas áreas de *clustering* de texto da web. Observa-se que as palavras-chave permaneceram no idioma inglês, visto que esses termos comumente não são traduzidos nos trabalhos publicados em português.

Por fim, com os trabalhos selecionados, uma revisão sistemática foi realizada e apresentada, a seguir, por ordem cronológica.

Huang (2008) comparou e analisou a eficácia das medidas de agrupamento particional e hierárquico para conjuntos de documentos textuais. O autor usou os *datasets 20 newsgroups* (postagem de grupos de notícias contendo uma variedade de tópicos, incluindo política, computação etc.), *WebKB* (coleção de páginas web sobre computação), *Classic* (coleção de artigos acadêmicos) além de outras coleções, com todos os textos no idioma inglês. Nos experimentos, testaram-se as medidas de distância e similaridade e os algoritmos *k-means* e hierárquicos em um conjunto de documentos textuais. Mais especificamente, foram avaliadas cinco medidas com experimentos empíricos: distância euclidiana, similaridade de cosseno, coeficiente de Jaccard, coeficiente de correlação de Pearson e média de divergência de Kullback-Leibler. Para o autor, há três componentes que afetam os resultados finais no processo de *clustering*, que são a representação dos objetos, a medidas de distância ou similaridade e o algoritmo de *clustering* utilizado.

Hadjidj *et al*. (2009) apresentaram um ambiente integrado de Mineração de Dados de e-mails utilizando-se algoritmos de classificação e de agrupamento para organizar as mensagens do corpus de e-mails *Enron* por assunto e por autor. Para o agrupamento, os autores utilizaram os *Expectation Maximization*, *K-Means* e *Bisecting K-Means*. Nessa pesquisa, uma vez que os *clusters* são obtidos, cada grupo é marcado com as palavras e frases mais frequentes. Isso facilita a localização, pois além de identificar o assunto de um grupo de e-mails, o *cluster* também pode ser empregado para acelerar a busca por palavras-chave. O método foi capaz de detectar padrões e descobrir autoria de e-mails e, através de análises de características estilométricas, o sistema conseguiu identificar os autores mais plausíveis de e-mails anônimos. A estilometria é o estudo estatístico de cinco características diferentes de estilo de escrita (lexical, sintático, estrutural, específico de domínio e idiossincrático) (HADJIDJ *et al*. (2009).

Iqbal *et al*. (2010) propuseram a extração de características de estilos de escrita de um conjunto de e-mails anônimos para posteriormente agrupar as mensagens por autor. Foram analisadas várias combinações de características, como léxicas, sintáticas, estruturais e específicas de domínio e três algoritmos de agrupamento: *K-means*, *Bisecting K-means* e *Expectation Maximization*. Nos experimentos, os autores avaliaram qual algoritmo de

clustering teve melhor desempenho para um determinado conjunto de dados de email, qual é a força relativa de cada um dos quatro tipos diferentes de recursos de escrita, qual o efeito da variação do número de autores nos resultados e quais são os efeitos da variação do número de mensagens de email por autor na qualidade dos *clusters*. Os pesquisadores concluíram que o algoritmo *k-means* foi mais eficaz no agrupamento das mensagens atingindo uma acurácia de 90% e a exatidão foi ainda melhor ao aumentar o tamanho do corpus, porém a precisão de todos os três algoritmos reduziu à medida que mais autores foram adicionados ao projeto experimental. O algoritmo EM apresentou resultados insignificantes e difíceis de melhorar com o ajuste de parâmetros.

Bouras e Tsogkas (2010) propuseram um aprimoramento do algoritmo de *k-means* usando o conhecimento externo dos hiperônimos do *WordNet* de forma a enriquecer o "saco de palavras" usado antes do processo de agrupamento, além de auxiliar o procedimento de geração de rótulos. Para isso, os autores investigaram a aplicação de um grande espectro de algoritmos de agrupamento, bem como medidas de similaridade, para artigos de notícias que se originam da Web. Segundo os autores, o aprimoramento do *k-means* resultou em uma melhoria significativa em relação aos *k-means* padrão para um corpus de artigos de notícias derivados de grandes portais de notícias. Além disso, o processo de rotulagem gera *tags* de *cluster* que são úteis e de alta qualidade.

Krishna e Bhavani (2010) propuseram o uso de um método conhecido, chamado algoritmo Apriori, para extrair os conjuntos de itens mais frequentes e elaborar uma abordagem eficiente para o agrupamento de texto com base nos conjuntos de itens mais frequentes. Uma abordagem planejada que consiste nas etapas de pré-processamento do texto, mineração de itens frequentes, particionamento dos documentos de texto com base em conjuntos de itens frequentes e por fim, agrupamento de documentos de texto em *clusters*. Os autores utilizaram o banco de dados Reuter 21578 para a experimentação e o desempenho de agrupamento da abordagem proposta foi efetivamente analisado. Por fim, os pesquisadores consideraram a abordagem eficaz para agrupamento de texto de acordo com os conjuntos de itens frequentes que fornecem redução significativa de dimensionalidade.

Nassif (2011) aplicou técnicas de agrupamentos de textos à Computação Forence, com o objetivo de agrupar documentos em análises periciais de computadores apreendidos durante investigações policiais. Para ilustrar tal abordagem, Nassif realizou um estudo comparativo de seis algoritmos de agrupamento de dados (*K-means*, *K-medoids*, *Single Link*, *Complete Link e Average Link*) que foram aplicados a cinco bases de dados textuais provenientes de investigações reais. Além disso, o autor estudou técnicas para estimar o número de grupos automaticamente, dado que este é um parâmetro crítico de vários algoritmos e é normalmente desconhecido a priori. Segundo Nassif, o *k-means*, apesar de

apresentar uma constante de tempo de execução elevada devido a muitos cálculos de distância entre objetos num espaço de alta dimensionalidade, o algoritmo é bastante escalável por possuir complexidade computacional linear em relação ao número de objetos, sendo indicado para grandes bases de dados.

Aggarwal, Zhao e Yu (2012) projetaram um novo algoritmo de *clustering*, que combina algoritmos clássicos de particionamento com modelos probabilísticos, a fim de criar uma abordagem de agrupamento mais eficaz. Os autores usaram informações auxiliares para fornecer insights adicionais, com o objetivo de melhorar a qualidade dos *clusters*. Essas informações são logs de usuários, links entre os textos e metadados associados aos documentos. No experimento foram utilizados o *Cora Data Set*, uma base que contem 19.396 publicações científicas no domínio da ciência da computação, o *DBLP-Four-Area Data Set*, que é um subconjunto extraído do DBLP que contém quatro áreas de pesquisa relacionadas à Banco de Dados, Mineração de Dados, Recuperação de Informações e Aprendizado de Máquina, o *IMDB Data Set*, uma coleção sobre filmes, e uma abordagem que verifica cuidadosamente a coerência das informações secundárias com as características do conteúdo de textos usados para formar os *clusters*. No processo, o algoritmo requer duas fases: A de inicialização, na qual uma abordagem de *cluster* de texto padrão é usada sem nenhuma informação secundária. Os centroides bem como o particionamento criado pelos *clusters* formados na primeira fase fornecem um ponto de partida inicial para a segunda etapa. E a fase principal, etapa do algoritmo que é executada começando com os grupos formados no estágio inicial e iterativamente os *clusters* são reconstruídos com o uso das informações auxiliares. Essa fase executa iterações alternadas que usam o conteúdo de texto e informações auxiliares para melhorar o processo de *clustering*.

Lama (2013) desenvolveu um sistema de *clustering* baseado em Mineração de Texto utilizando o algoritmo de *k-means*. Em sua tese, o autor usou uma coleção de notícias e os links dos diferentes sites dos quais as informações foram coletadas. Esses dados foram obtidos no formato XML e usados para alimentar o sistema de *cluster*. O arquivo XM, contendo as manchetes foi então pré-processado usando técnicas de pré-processamento de documentos e, finalmente, as notícias foram agrupadas com base em suas semelhanças. Os *clusters* foram formados baseados nos centroides e na distância mínima entre os documentos.

Afonso (2013), em sua pesquisa de doutorado, investigou o algoritmo Evolucionário Convencional de agrupamento, o conhecido algoritmo *Expectation Maximization* e o algoritmo de agrupamento *X-Means*, uma extensão do conhecido algoritmo *k-means*, aplicando-os a artigos científicos escritos na língua portuguesa. O autor aplicou os algoritmos *clustering* em diversos corpora de testes com textos de diferentes áreas do conhecimento e observou

formalmente se grupos de artigos científicos diferentes produzem resultados melhores ou piores para textos escritos no idioma português do Brasil.

Lönnberg e Yregård (2013) propuseram uma técnica para agrupamento de artigos de notícias em larga escala. Os autores examinaram diferentes abordagens sobre como agrupar as notícias para que dois artigos sobre a mesma informação pertençam ao mesmo *cluster*. Para isso, eles examinaram os algoritmos tradicionais de agrupamento e as etapas de pré-processamento e propuseram uma solução capaz de lidar com uma grande quantidade de artigos, além de produzir *clusters* de alta qualidade e fazer isso em um curto período de tempo. Porém, a metodologia de Lönnberg e Yregård tem a limitação de trabalhar somente com artigos escritos em inglês.

Godfrey *et al*. (2014) aplicaram técnicas de *clustering* a uma coleção de cerca de 30 mil *tweets* extraídos do Twitter. Para agrupar os pequenos comentários, os autores utilizaram o algoritmo *k-means* e o método de Fatoração de Matrizes Não-Negativas (NMF, do inglês *Non-Negative Matrix Factorization*). A técnica NMF foi estudada como um método para análise de dados capaz de extrair conhecimento sobre um objeto a partir do estudo de suas partes (LEE; SEUNG, 2000). Os autores compararam os métodos e os dois algoritmos apresentaram resultados semelhantes, mas NMF mostrou-se mais rápido e forneceu resultados mais facilmente interpretados.

Driscoll e Thorson (2015) exploraram no artigo "*Searching and Clustering Methodologies: Connecting Political Communication Content across platforms*", métodos para criar conjuntos de dados integrados de conteúdo de redes sociais relacionadas a eventos políticos. Os autores usaram agrupamento de texto semiautomatizado para identificar postagens similares em todos os serviços de redes sociais. Essas abordagens ajudam a revelar notícias falsas relacionadas à política, pois podem ocorrer distorções quando a informação é acessada em apenas uma plataforma isoladamente.

Para Xiong *et al*. (2016), o algoritmo de clusterização *k-means* é influente na Mineração de Dados. O tradicional *k-means* tem sensibilidade para os centros iniciais de *cluster*, levando o resultado do agrupamento depender excessivamente desses centros. A fim de superar essa falha, os autores propuseram um novo algoritmo de clusterização de texto, que na verdade é um *k-means* aprimorado, que é capaz de otimizar os centros iniciais dos *clusters*, pois a seleção de centros determina a qualidade do agrupamento. Para verificar a viabilidade e eficácia do algoritmo proposto, utilizou-se o corpus chinês fornecido pelo professor Li Ronglu, da Universidade de Fudan, como dados experimentais. Foram selecionados aleatoriamente 880 textos pertencentes a 5 categorias.

Nessa pesquisa, os autores apresentaram um modelo que combina o "método de otimização de distância" e o "método de densidade" para determinar os melhores centros

iniciais dos grupos. Nesse contexto, o algoritmo primeiramente calcula a densidade de cada objeto no conjunto de dados e, em seguida, julga qual elemento é um ponto isolado. Depois de remover todos os pontos isolados, é obtido um conjunto de elementos com alta densidade. Posteriormente, escolhe k objetos de alta densidade como sendo os centros iniciais dos *clusters*, em que a distância entre os objetos de dados é a maior. Segundo os pesquisadores, os resultados experimentais mostram que o algoritmo *k-means* aprimorado pode melhorar a estabilidade e precisão do agrupamento de texto, porém a metodologia teve uma melhora pequena nos resultados dos agrupamentos e que em trabalhos futuros será analisado a complexidade e o tempo de execução do algoritmo.

Abualigah, Khader e Al-Betar (2016) propuseram uma técnica baseada em multi-objetivos, ou seja, combinaram medidas de distância e similaridade para melhorar o processo de clusterização de documentos. Segundo os autores, o método de *clustering* multiobjetivos é combinado com dois critérios de avaliação que emergem como uma alternativa robusta em várias situações. Assim, eles combinaram a distância euclidiana com a similaridade de cosseno e o desempenho da função multiobjetivos foi investigado usando o *k-means*. Os pesquisadores conduziram os experimentos usando sete conjuntos de textos e os resultados mostraram que o método baseado em multiobjetivos proposto supera as outras medidas em relação ao desempenho da metodologia no agrupamento de textos.

Yang, Wu e Cheng (2017) fizeram uma pesquisa objetivando melhorar o processo de *text clustering*. Para isso, os autores pesquisaram sobre agrupamento de texto distribuído com base em conjuntos de itens frequentes. Eles enfocaram a questão da velocidade e precisão na clusterização de documentos usando computação distribuída. Dessa forma, eles propuseram um método de *clustering* baseado em itens frequentes. No processo de agrupamento, elaborou-se uma matriz de similaridade do conjunto de objetos e, em seguida, construiu-se um vetor com base na frequência dos elementos e, por fim, os autores usaram o algoritmo *k-means* para o agrupamento. Como o processo de aglomeração envolve um grande número de cálculos de matrizes e várias iterações, no caso de usar um único computador, os requisitos do hardware são muito altos. Para resolver essa questão, os autores usaram *o Hadoop* e o *Map Reduce* para realizar a computação distribuída e reduzir radicalmente o tempo de *clustering*.

Em virtude do que foi apresentado, observa-se que existem muitos trabalhos relacionados a *clustering e* muitas outras abordagens podem ser encontradas na literatura no campo de agrupamento de textos devido às suas potenciais aplicações. Igualmente importantes são os novos desafios intelectuais que apresenta essa área para a comunidade de pesquisa, visto a importância de organizar as informações disponibilizadas na web, principalmente as relacionadas às notícias publicadas no idioma português.

Por conseguinte, a pesquisa realizada sobre os trabalhos que já foram publicados sobre o tema possibilita um mapeamento de quem já escreveu e o que já foi escrito sobre o assunto. Além disso, obter informações sobre a situação atual do problema pesquisado, de forma a conhecer publicações existentes e os aspectos que já foram abordados, é de extrema valia para o avanço de novas investigações nessa área. Nesse contexto, este trabalho visa usar as técnicas de aprendizado não supervisionado para *clustering* de notícias coletadas dos principais jornais *on-line*, publicadas no idioma português, pois são poucos os trabalhos que exploram esse tipo de corpus. Desse modo, durante os experimentos, principalmente na fase de pré-processamento, as técnicas serão testadas e adaptadas para processarem as notícias em português.

4 METODOLOGIA

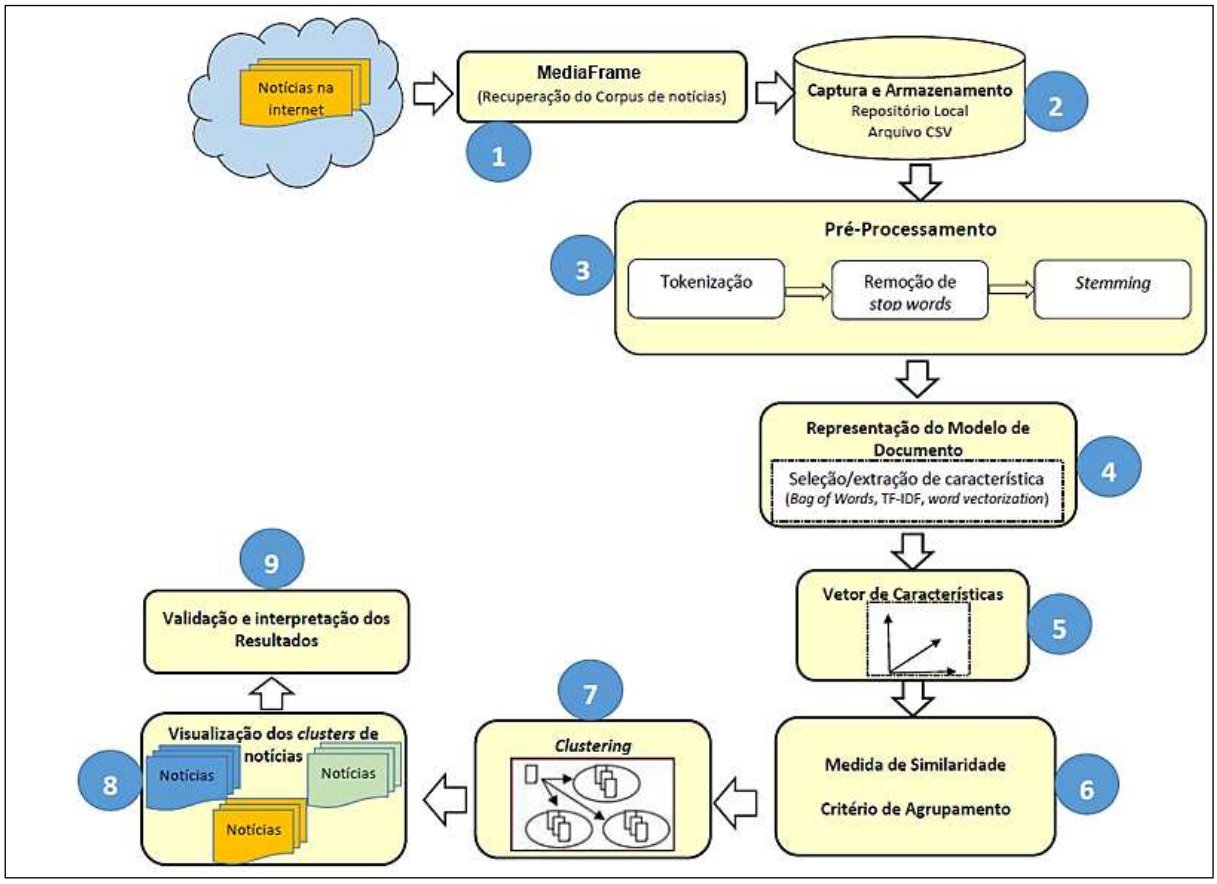
Este capítulo discute a metodologia e a estratégia de projeto a ser implementada para responder à questão de pesquisa deste estudo. Para Kothari (2004), a pesquisa compreende a definição e redefinição de problemas, formulação de hipóteses, passando pelas etapas de coleta, organização e avaliação dos dados, fazendo deduções e chegando às conclusões. Além de testar os resultados para confirmar se as hipóteses são verdadeiras.

Trata-se de uma pesquisa quali-quantitativa, por usar métodos quantitativos, como técnicas estatísticas, e também qualitativos, por se caracterizar pela qualificação dos dados coletados de forma a fazer uma análise mais profunda sobre o tema estudado. Segundo Fiel (2017), as pesquisas qualitativas são voltadas para os aspectos subjetivos. É uma análise onde não se utiliza atributos numéricos e não se mensura os dados, como por exemplo as emoções e o estudo de sentimentos, sensações e opiniões. Já as pesquisas qualitativas utilizam técnicas estatísticas para quantificar os dados analisados.

Para a realização dos experimentos desta pesquisa, utilizou-se a linguagem de Programação Python, por se tratar de uma linguagem *open source*, que pode lidar muito bem com dados textuais e pelo fato dela possuir bibliotecas para carregamento de dados, visualização, estatísticas, Processamento de Linguagem Natural, Recuperação da Informação e *Machine Learning,* que podem ser usadas para encontrar padrões e resolver problemas relacionados à análise de documentos. Essa vasta caixa de ferramentas fornece aos cientistas de dados uma grande variedade de funcionalidades para fins gerais e especiais. Uma das principais vantagens de usar o Python é a capacidade de interagir diretamente com o código, visto que o Aprendizado de Máquina e a Análise de Dados são processos fundamentalmente iterativos, nos quais os dados conduzem a análise. É essencial que esses processos tenham instrumentos que permitem interação fácil e rápida. Por isso, essa linguagem foi escolhida para analisar a coleção de notícias deste trabalho, pois os pacotes, prontamente disponíveis, reduzem o tempo e os esforços necessários para o desenvolvimento.

A Figura 19 apresenta o fluxograma da metodologia proposta para o agrupamento de notícias disseminadas pelos meios de comunicação em massa:

FIGURA 19: Fluxograma da metodologia



Fonte: elaborado pela autora

As etapas, ilustradas no fluxograma, serão descritas nas próximas subseções:

4.1 Captura do Corpus de notícias utilizando o *Media Frame*

Esta etapa teve como objetivo coletar, dos principais sites de notícias (Extra, O Globo, G1, Folha de São Paulo, Estadão etc.), uma amostra de documentos para a realização do processamento desejado. Ou seja, adquirir uma coleção de notícias dos principais jornais *on-line* para a realização dos experimentos.

Assim, neste trabalho, utilizou-se o *Media Frame* para coletar os três corpora que foram utilizados nos testes, conforme ilustrado no Quadro 8: Para o primeiro experimento, usou-se um corpus de notícias relacionadas ao tema "**politica**"; para o segundo, uma coleção com temas relacionados à **política**, **economia**, **educação** e **saúde**. E, por fim, uma amostra com assuntos mais diversificados, ou seja, **biologia**, **eletricidade**, **futebol** e **economia**.

QUADRO 8: Temas nas notícias x quantidade

Experimento	Temas das notícias	Quantidade de notícias
1	Política	123
2	política, economia, educação e saúde	107
3	biologia, eletricidade, futebol e economia	50

Fonte: Elaborada pela autora

Assim, coletaram-se as notícias para testar a metodologia desta pesquisa e verificar se os algoritmos de *clustering* são capazes de separar a coleção por subcategorias. Além disso, utilizaram-se três coleções diferentes para analisar se o tamanho e o tipo de corpus interferem nos resultados. Para assegurar a validade das conclusões tiradas delas, não se deve buscar apenas a maior similaridade dos textos com os quais os algoritmos de agrupamento deveriam estar lidando em uma situação do mundo real, como também padrões de notícias devem ser sempre usados, na medida do possível.

Para a recuperação das notícias, primeiramente, informaram-se as palavras-chave para a busca (temas das notícias), escolheram-se alguns jornais *on-line* para realizar a coleta e descobrir os *feeds* associados a cada fonte de mídia e, também, determinou-se o período desejado em que as notícias foram publicadas. Posteriormente, o coletor permitiu rastrear cada uma das fontes para descobrir os informes relacionadas ao assunto pesquisado. Em seguida, fez-se o download das coleções recuperadas em formato csv, possibilitando, deste modo, o armazenamento dos arquivos. Assim, o corpus foi armazenado em um repositório local para posteriormente ser processado.

4.2 Pré-processamento do corpus de notícias

Devido à natureza textual não estruturada, os documentos necessitam de um pré-processamento para serem submetidos aos algoritmos de aprendizagem. A transformação dos textos em uma representação mais adequada é uma etapa de suma importância, visto que isso tem uma influência fundamental em quão bem um algoritmo de aprendizado poderá generalizar a partir dos exemplos (SEBASTIANI, 2002). Nesta fase, as notícias recuperadas pelo *Media Frame* foram submetidas à inúmeras operações para serem representadas estruturalmente, pois antes de realizar qualquer análise, é necessário normalizar os documentos. Diante disso, os conteúdos das notícias, acompanhados de seus títulos, foram pré-processados usando as técnicas tokenização, remoção de *stop words*, *stemming* etc. descritas a seguir:

a) Tokenização

A tokenização ou *tokenization* é o primeiro passo de uma operação de pré-processamento. Para isso, primeiramente, dividiram-se as notícias em unidades menores, procedimento que decompõe uma unidade de documento em pedaços denominados *tokens*.

b) Remoção das stop words

Nesta etapa, realizou-se a eliminação das *stop words*, que são palavras que têm pouca ou nenhuma significância. Os artigos, os pronomes, as preposições e as interjeições são considerados *stop words*. Elas geralmente são removidas do texto durante o processamento, de modo a reter as palavras mais relevantes. Como não existe um conjunto universal de *stop words* e cada domínio ou idioma pode ter seu próprio, neste trabalho, adotou-se a lista de *stop words* para o idioma Português disponibilizada pelo NLTK e novas palavras, que são comuns em artigos de notícias, como publicação, *copyright,* direitos reservados etc., foram adicionadas à lista com o objetivo de melhorar o resultado da clusterização.

E para reduzir ainda mais os termos de pouca relevância nos textos, atualizou-se a lista de *stop words* com várias novas palavras, tais como verbos genéricos ou substantivos sem muita relevância, que foram cuidadosamente selecionadas depois de analisar o corpus.

Portanto, a fase de remoção das *stop words* é importante para reduzir o número de palavras envolvidas nos processos posteriores de análise, de modo a conseguir um melhor desempenho sem perda significativa de informação útil.

c) *Stemming*

O passo seguinte foi identificar e unificar palavras que possuem o mesmo significado semântico. Muitas vezes, prefixos e sufixos são anexados a um tronco de palavras para mudar seu significado ou criar uma nova palavra. O objetivo dessa etapa é remover esses afixos e retornar as palavras em sua forma básica, ou seja, reduzir a palavra a sua raiz. Para isso, utilizaram-se os *stemmers* RSLP, Porter e Snowball.

d) N-gramas

Aplicou-se também a técnica n-gramas, usando n=2, de modo que uma sequência contígua de duas palavras, que aparecem sequencialmente nos textos, seja unida. Com isso, é possível buscar na coleção, os termos compostos com um único significado semântico. Essa técnica ajuda a reduzir a quantidade de palavras presentes no corpus.

4.3 Representação do modelo de documentos

Uma importante etapa no processo de clusterização é representar o conteúdo do texto sob a forma de expressão matemática para posterior análise e processamento. A representação de documentos pode ser feita usando diversas abordagens, entre elas, a *Bag-of-Words*.

Portanto, neste passo, com o corpus de notícias normalizado, uma matriz de característica foi construída. Para isso, mantiveram-se os *tokens* dos textos nos documentos normalizados e as características foram extraídas, com base no modelo TF-IDF, de modo que cada atributo ocorra em pelo menos 25% e no máximo 85% das notícias que foram recuperadas dos principais jornais *on-line*. Para controlar a porcentagem, são usadas as frequências mínima e máxima dos termos no documento.

4.4 Criação de vetores de documentos

Para dividir um conjunto de notícias em *clusters*, é necessário um método que permita compará-los. Infelizmente, dois artigos não podem ser comparados diretamente, portanto, eles devem ser representados de forma que um computador possa fazer essas comparações. Há várias maneiras de representar um texto para que a máquina possa compará-lo com outros documentos. Dessa forma, nesta etapa, as notícias são representadas como um vetor, que é uma sequência de atributos e de pesos. E a importância do termo é calculada conforme um dado critério, como a frequência do termo em um documento ou o número de documentos que contêm o termo. Neste processo, utilizou-se o modelo de espaço vetorial, também chamado de modelo vetorial. É um modelo algébrico popular usado para representar documentos textuais em forma de vetores.

Assim, nesta fase, empregou-se o modelo de espaço vetorial para representar a coleção de notícias usando a Frequência dos termos, a Frequência Inversa dos Termos ou esquema de ponderação TF-IDF. Ademais, o Modelo de Espaço Vetorial é um dos métodos mais eficientes para representação de documentos como vetores usando o peso da frequência do termo (LAMA, 2013). Dessa forma, toda a coleção de notícias foi representada usando esse modelo.

4.5 Medida da Similaridade e identificação dos aglomerados

Neste passo, usaram-se as métricas de distância para medir e analisar a semelhança entre as notícias para, posteriormente, identificar os aglomerados. As medidas de

Similaridade do Cosseno, Distância de Manhattan e Distância Euclidiana foram utilizadas nesta etapa.

Após a escolha da medida de similaridade a ser utilizada, o passo seguinte consiste em organizar a coleção de notícias em *clusters*. Assim, empregou-se a técnica de *clustering* para dividir as notícias dos principais jornais *on-line* em grupos menores. Na análise de agrupamentos, diferentemente da análise discriminante (classificação), os *clusters* não são pré-definidos, ao invés disso, a técnica é usada para identificar os grupos. A análise de *clustering* geralmente envolve pelo menos três passos (HAIR et al., 2005. p. 35): O primeiro é a medida, de alguma forma, da similaridade ou associação entre os objetos para determinar quantos grupos realmente existem na amostra. O segundo passo é o real processo de agrupamento, em que entidades são particionadas em grupos (agrupamentos). O último passo é analisar o perfil das variáveis para determinar sua composição. Muitas vezes, isso é possível pela aplicação da análise discriminante aos *clusters* identificados pela técnica de agrupamento.

Desse modo, para a realização desta etapa, utilizou-se a técnica de agrupamento exclusivo, usando os algoritmos *K-means, Affinity Propagation* e também os algoritmos de agrupamento hierárquico. O *clustering* é realizado sobre a coleção de notícias, através do particionamento do conjunto de textos em subconjuntos ou *clusters*, de forma que documentos dentro de um mesmo *cluster* sejam similares e dentro de *clusters* diferentes sejam distintos. Além disso, nesta fase, realizaram-se, também, adaptações nos algoritmos e vários testes, com parâmetros diferentes e com amostras diversificadas, na tentativa de obter os melhores resultados.

4.6 Visualização dos clusters de notícias

Neste passo, os resultados obtidos no processo de *clustering* foram apresentados. De fato, o resultado são os grupos de notícias formados. Porém, são grandes os desafios associados à visualização de *clusters*. Isso ocorre pelo fato de lidar com espaços de recursos multidimensionais e dados de texto não estruturados. Os próprios vetores de características numéricos podem não ter nenhum sentido para os leitores se fossem diretamente visualizados. Assim, existem algumas técnicas, como Análise de Componentes Principais (ACP), em inglês, *Principal Component Analysis* (PCA) ou Escala Multidimensional, em inglês, *Multidimensional Scaling* (MDS) (SARKAR, 2016) que podem ser utilizadas para reduzir a dimensionalidade, de modo que os *clusters* possam ser visualizados. Neste trabalho, usaram-se dendogramas, diagramas de dispersão e o MDS para visualização dos *clusters*.

4.7 Análise, validação e interpretação dos resultados

O resultado do agrupamento deve ser analisado e validado para verificar se os *clusters* não ocorreram por acaso, pois qualquer algoritmo de *clustering* pode encontrar grupos, independentemente se existe ou não similaridade entre os objetos. Sendo assim, a análise da qualidade de agrupamento é uma questão importante, principalmente, por ser uma técnica não supervisionada. Desse modo, além de usar medidas relativas para comparar os diferentes grupos obtidos através de diferentes configurações de parâmetros para o mesmo algoritmo, utilizou-se também o Coeficiente da Silhueta como medida interna para avaliar a qualidade dos aglomerados. Por fim, usou-se no terceiro experimento uma amostra menor de notícias de forma que foi possível analisar e interpretar os *clusters* com o intuito de averiguar se realmente os textos de um mesmo grupo são similares. Para facilitar a análise dos assuntos mais importantes das notícias, utilizou-se o módulo de Modelagem de Tópicos do software *open source* Orange Canvas[5] para extrair os tópicos mais relevantes da coleção de notícias em estudo e, assim, facilitar a interpretação do conteúdo das mesmas. Para isso, aplicou se a técnica de Modelagem de Tópicos no corpus já pré-processado para identificar a estrutura temática oculta nos documentos. O Orange Canvas, além de apresentar um campo para o utilizador informar o número de tópicos, possibilita realizar a modelagem usando a Indexação Semântica Latente, a Alocação de Dirichlet Latente ou o Processo de Dirichlet Hierárquico (*Hierarchical Dirichlet Process* -PDH). O PDH é uma evolução do LDA em que o usuário não precisa informar o número de tópicos.

Logo, a ferramenta Orange Canvas auxiliou na descoberta dos tópicos ocultos presentes na coleção de notícias através de *clusters* de palavras encontradas em cada documento e em suas respectivas frequências. Tendo em vista que os documentos, em geral, contêm vários tópicos em proporções diferentes, a ferramenta também informa o peso do tópico por documento e dessa forma, ajuda na análise e interpretação dos resultados do processo de agrupamento.

[5] https://orange.biolab.si/

5 ANÁLISE DOS RESULTADOS

Esta pesquisa buscou olhar para técnicas e problemas que foram anteriormente negligenciados na literatura existente sobre o agrupamento de textos no idioma português. Consequentemente, realizou-se uma série de experimentos para testar na prática intuições e *insights*, a fim de obter um conhecimento real de sua adequação e aplicabilidade.

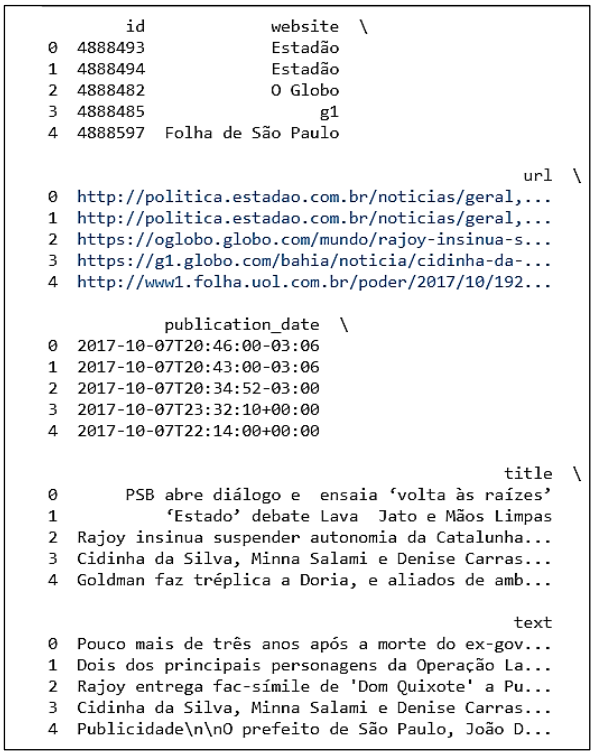
Na realização dos testes, os algoritmos de agrupamento precisam ser controlados por alguns parâmetros que devem ser definidos adequadamente quando usados nos experimentos. Diante disso, em alguns casos, os valores dos parâmetros foram ajustados para obter os melhores resultados em cada conjunto de dados. Ao passo que, em outras situações, evitou-se fornecer aos algoritmos informações que não estariam disponíveis em um cenário real para se ter uma ideia melhor do esforço do sistema. A escolha entre um ou outro depende da natureza da tarefa que está sendo abordada. De qualquer forma, seja qual for o método escolhido, teve-se o cuidado ao fazer o ajuste dos parâmetros de maneira justa a todos os métodos comparados.

Por conseguinte, para testar a metodologia, foram realizados três experimentos com corpora diferentes, conforme descrito nas próximas subseções.

5.1 Experimento 1: Teste com um corpus de 123 notícias relacionadas à política

Para realização deste experimento, utilizou-se o *Media Frame* para a aquisição do corpus. Assim, coletaram-se 123 notícias dos principais jornais *on-line* (Extra, O Globo, G1, Folha de São Paulo, Estadão) usando na consulta a palavra-chave “politica” e filtrando as matérias que foram publicadas nos dias 06/10/17 e 07/10/17. A Figura 20 mostra os cinco primeiros documentos da coleção.

FIGURA 20: Cinco primeiras notícias do corpus



```
         id             website  \
0  4888493             Estadão
1  4888494             Estadão
2  4888482             O Globo
3  4888485                  g1
4  4888597  Folha de São Paulo

                                                 url  \
0  http://politica.estadao.com.br/noticias/geral,...
1  http://politica.estadao.com.br/noticias/geral,...
2  https://oglobo.globo.com/mundo/rajoy-insinua-s...
3  https://g1.globo.com/bahia/noticia/cidinha-da-...
4  http://www1.folha.uol.com.br/poder/2017/10/192...

           publication_date  \
0  2017-10-07T20:46:00-03:06
1  2017-10-07T20:43:00-03:06
2  2017-10-07T20:34:52-03:00
3  2017-10-07T23:32:10+00:00
4  2017-10-07T22:14:00+00:00

                                               title  \
0      PSB abre diálogo e  ensaia 'volta às raízes'
1          'Estado' debate Lava  Jato e Mãos Limpas
2  Rajoy insinua suspender autonomia da Catalunha...
3  Cidinha da Silva, Minna Salami e Denise Carras...
4  Goldman faz tréplica a Doria, e aliados de amb...

                                                text
0  Pouco mais de três anos após a morte do ex-gov...
1  Dois dos principais personagens da Operação La...
2  Rajoy entrega fac-símile de 'Dom Quixote' a Pu...
3  Cidinha da Silva, Minna Salami e Denise Carras...
4  Publicidade\n\nO prefeito de São Paulo, João D...
```

Fonte: Elaborada pela autora

Com as notícias recuperadas, a tarefa seguinte foi o pré-processamento, ou seja, preparar a coleção para que elas possam ser compreendidas pelos algoritmos de Aprendizagem de Máquina.

5.1.1 Pré-processamento

Antes de testar qualquer algoritmo de *clustering*, o primeiro passo é o pré-processamento. Deste modo, com objetivo de agrupar as notícias em subcategorias, realizaram-se as seguintes etapas:

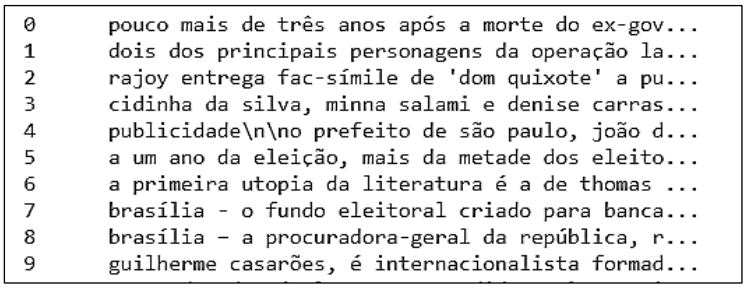
I) Normalização do corpus

Para normalizar o corpus, cumpriram-se os seguintes passos:

a) Padronização dos textos para minúsculos.

Para facilitar as próximas tarefas, os textos foram convertidos em minúsculos. A Figura 21 ilustra as dez primeiras notícias após a conversão.

FIGURA 21: Notícias transformadas em minúsculo



```
0      pouco mais de três anos após a morte do ex-gov...
1      dois dos principais personagens da operação la...
2      rajoy entrega fac-símile de 'dom quixote' a pu...
3      cidinha da silva, minna salami e denise carras...
4      publicidade\n\no prefeito de são paulo, joão d...
5      a um ano da eleição, mais da metade dos eleito...
6      a primeira utopia da literatura é a de thomas ...
7      brasília - o fundo eleitoral criado para banca...
8      brasília – a procuradora-geral da república, r...
9      guilherme casarões, é internacionalista formad...
```

Fonte: Elaborada pela autora

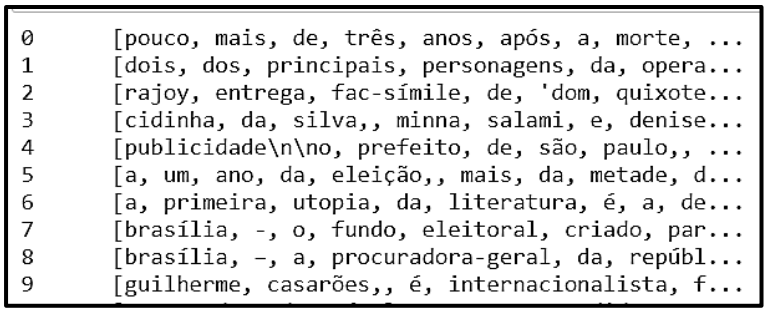
b) Eliminação de caracteres indesejados

Posteriormente, utilizou-se do conceito de Expressão Regular para realizar uma limpeza nos textos e, assim, eliminar os caracteres indesejados e os números, conforme exemplificado entre parênteses e separados pela barra vertical, (\|.|,|:|;|!|\?|\(|\)|\||\+|'|"|‘|’|“|”|'|\'|…|\-|–|—|\$|&|*|>|<@). Além disso, removeram-se as URLs e usou um analisador HTML (*Parse* HTML) para detectar *tags* HTML e considerar apenas o texto dessas marcações.

II) Tokenização

A próxima etapa foi a tokenização, ou seja, a coleção foi dividida em unidades menores, denominada *tokens.* Esse procedimento se faz necessário para a realização das demais operações do pré-processamento. A Figura 22 ilustra as notícias transformadas.

FIGURA 22: Notícias convertidas em *tokens*



```
0      [pouco, mais, de, três, anos, após, a, morte, ...
1      [dois, dos, principais, personagens, da, opera...
2      [rajoy, entrega, fac-símile, de, 'dom, quixote...
3      [cidinha, da, silva,, minna, salami, e, denise...
4      [publicidade\n\no, prefeito, de, são, paulo,, ...
5      [a, um, ano, da, eleição,, mais, da, metade, d...
6      [a, primeira, utopia, da, literatura, é, a, de...
7      [brasília, -, o, fundo, eleitoral, criado, par...
8      [brasília, –, a, procuradora-geral, da, repúbl...
9      [guilherme, casarões,, é, internacionalista, f...
```

Fonte: elaborada pela autora

III) Remoção das *stop words*

O passo seguinte foi a remoção das *stop words*, que são palavras comuns, que repetem muitas vezes, mas não acrescentam e nem retiram informações relevantes. A Figura 23 apresenta dez palavras consideradas *stop list,* disponibilizada pela biblioteca NLTK do Python.

FIGURA 23: *Stop words*

```
>>> stopwords = nltk.corpus.stopwords.words('portuguese')
>>> stopwords[:10]
['a', 'ao', 'aos', 'aquela', 'aquelas', 'aquele', 'aqueles',
 'aquilo', 'as', 'até']
 >>>
```

Fonte: Elaborada pela autora

Além da lista de *stop words* disponibilizada pelo NLTK, uma nova lista foi adicionada após análise dos textos de modo a melhorar os resultados.

As nuvens de palavras ilustradas nas Figuras 24 e 25 exibem os termos mais frequentes da coleção, antes e depois da remoção das *stop words*, respectivamente.

FIGURA 24: Nuvem de palavras com *stop words*

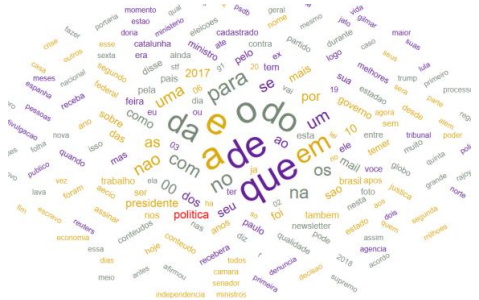
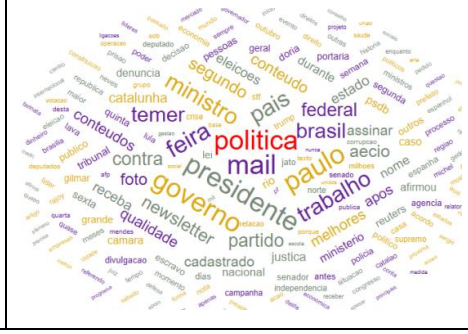


FIGURA 25: Nuvem de palavras sem as *stop words*



Fonte: Elaborada pela autora

Conforme mostraram as nuvens de palavras, a pesquisa na área de análise de textos se torna inviável sem a remoção das *stop words.* As imagens ilustram bem essa questão, pois os artigos, preposições, conjunções etc. repetem muitas vezes, não têm significados relevantes, mas se presentes no corpus, prejudicam o resultado. Assim, a remoção desses termos é importante para que o agrupamento seja feito usando as palavras de maior peso da coleção para que o resultado seja relevante.

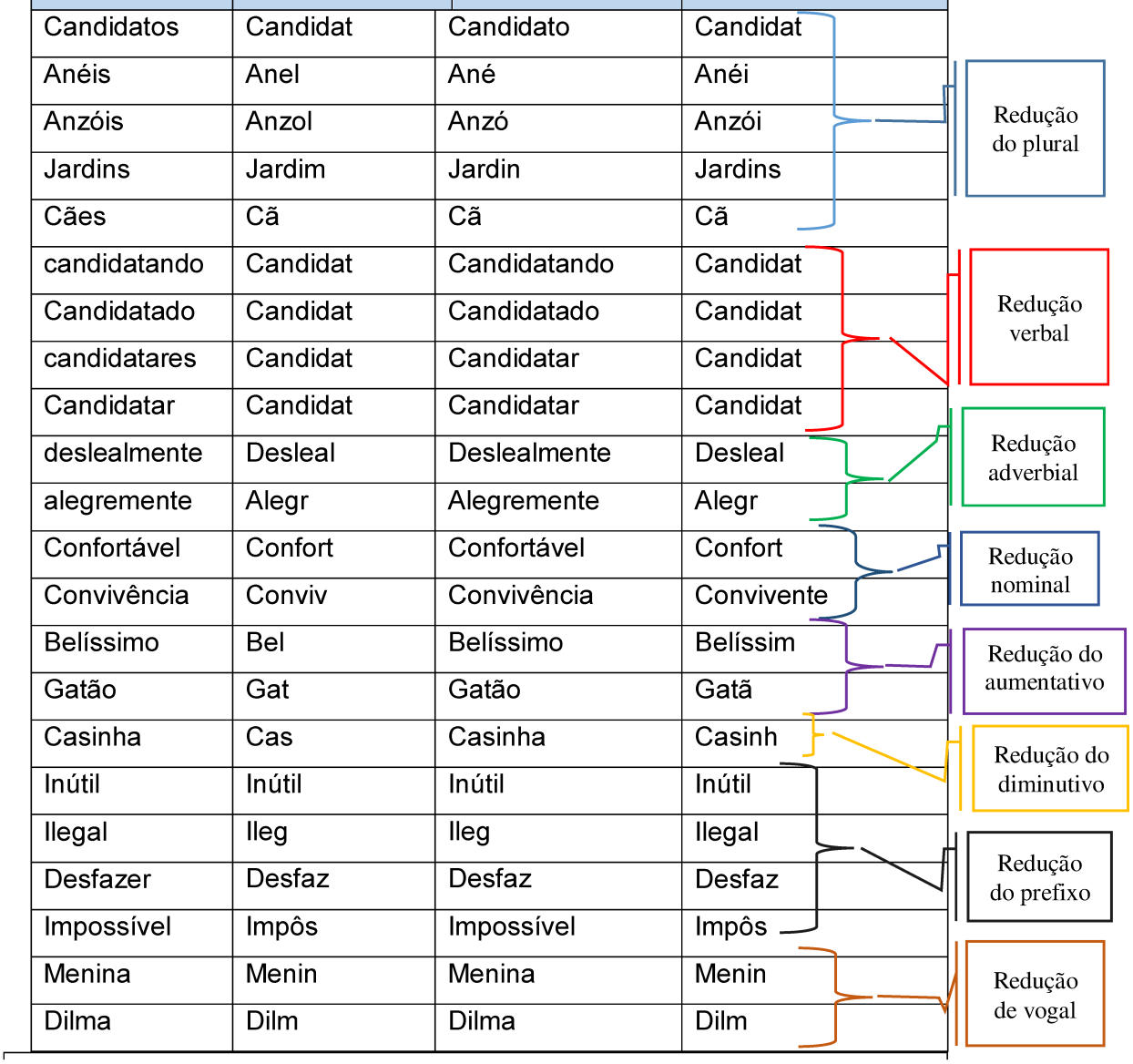
5.1.2 Teste dos algoritmos de *Stemming*

Primeiramente, realizou-se um teste com os algoritmos de *stemming* RSLP, Porter e Snowball, com o objetivo de verificar qual deles apresenta melhor desempenho ao ser alimentado com textos em português. Para isso, analisaram-se as taxas de erros e acertos quanto à redução verbal, nominal, adverbial, do plural, do aumentativo, do prefixo e da vogal. Dessa forma, os *stemmers* que conseguirem reduzir melhor as palavras às suas raízes, apresentando um maior número de acertos, são os que tiveram uma melhor performance.

O Quadro 9 mostra o resultado da aplicação da técnica de *stemming* em algumas palavras.

QUADRO 9: Comparação dos algoritmos de *stemming* RSLP, Porter e *Snowball*

Palavra	RSLP Stemmer	Porter Stemmer	Snowball Stemmer
Candidatos	Candidat	Candidato	Candidat
Anéis	Anel	Ané	Anéi
Anzóis	Anzol	Anzó	Anzói
Jardins	Jardim	Jardin	Jardins
Cães	Cã	Cã	Cã
candidatando	Candidat	Candidatando	Candidat
Candidatado	Candidat	Candidatado	Candidat
candidatares	Candidat	Candidatar	Candidat
Candidatar	Candidat	Candidatar	Candidat
deslealmente	Desleal	Deslealmente	Desleal
alegremente	Alegr	Alegremente	Alegr
Confortável	Confort	Confortável	Confort
Convivência	Conviv	Convivência	Convivente
Belíssimo	Bel	Belíssimo	Belíssim
Gatão	Gat	Gatão	Gatã
Casinha	Cas	Casinha	Casinh
Inútil	Inútil	Inútil	Inútil
Ilegal	Ileg	Ileg	Ilegal
Desfazer	Desfaz	Desfaz	Desfaz
Impossível	Impôs	Impossível	Impôs
Menina	Menin	Menina	Menin
Dilma	Dilm	Dilma	Dilm



Fonte: Elaborada pela autora

Para o cálculo do erro, foi considerado que o algoritmo acertou a redução da palavra caso o retorno tenha sido o radical correto ou a palavra de origem.

Observa-se que, em relação à redução do plural, o RSLP apresentou o melhor desempenho, pois conseguiu acertar quatro palavras em um total de cinco, enquanto Porter errou todas e o *Snowball* acertou apenas uma. Quanto à redução adverbial, os algoritmos foram testados com duas palavras, das quais eles deveriam remover o sufixo **mente**. RSLP e *Snowball* acertaram nos dois casos e Porter não acertou em nenhuma das situações. No caso

da redução nominal, os algoritmos deveriam remover os sufixos **ável** e **ência**. Os algoritmos RSLP e *Snowbal* acertaram a redução da palavra **confortável** e somente o RSLP acertou a remoção do sufixo da palavra **convivência**. Porter errou nos dois casos. Em relação à redução do aumentativo, diminutivo e superlativo, RSLP acertou todas as três palavras utilizadas para o teste e outros dois algoritmos erraram. Quanto à redução de vogal, os algoritmos RSLP e *Snowball* conseguiram remover corretamente a letra **a** da palavra **menina**, porém ambos erraram ao remover essa letra do nome próprio **Dilma**. Já o algoritmo Porter não utiliza a regra de remoção de vogal. Por fim, nenhum dos algoritmos usam regras para remoção dos prefixos, isso poderia ser implementado, porém não é o objeto de estudo desta pesquisa.

A Tabela 4 mostra o percentual de erros dos radicalizadores, apresentando separadamente a porcentagem de *overstemming* e de *understemming*. Os dados para o cálculo foram retirados do Quadro 9.

TABELA 4: Porcentagem de erros dos algoritmos

Stemmer	% erro	% Overstemming	% Understemming
RSLP	31,0	4,5	0
Porter	86,4	9,1	72
Snowball	63,6	4,5	27,3

Fonte: Elaborada pela autora

Ao comparar os *Stemmers*, o RSLP apresentou melhor desempenho para palavras na língua portuguesa. Porter não teve boa performance, pois foi projetado para trabalhar com textos no idioma inglês, porém, a sua versão para o português (*Snowball*) conseguiu melhorar o desempenho, mas mesmo assim foi bem inferior ao RSLP.

As Figuras 26 e 27 são recortes da coleção de notícias após a aplicação das técnicas de *stemming* usando os algoritmos *Porter* e *RSLPStemme*r, respectivamente.

FIGURA 26: Notícias submetidas ao algoritmo *Porter Stemmer*

```
print 'Título:', media_titles[0]
print 'Notícia:', media_text[0][:1000]

Título: psb abr diálogo ensaia ‘volta raízes’
Notícia: pouco trê ano após mort ex-governador pernambuco eduardo campo plena campanha presidencial, psb abriu can
ai diálogo ciro gome (pdt), marina silva (rede) geraldo alckmin (psdb) 2018 ensaia “volta origens” centro-esquer
da fazendo oposição governo pmdb. cúpula legenda aguarda próxima semana debandada deputado governistas. dirigent
calculam 12 18 36 deputado devem deixar partido continuarem base governo michel temer. senha psb reforçar discu
rso voltou origen centro-esquerda. psb rompeu então president dilma rousseff 2013, lançou eduardo campos. 2015 a
poiou impeach petista. viabilizar candidatura planalto, 2014, campo inchou partido levando nome pouca nenhuma afi
nidad histórica ideológica legenda. é ala psb rebel cúpula partidária decidiu indicaria nome governo temer. pdt
fez oferta psb apoi ciro gome considerada generosa cúpula partido: aliança horizontal. newslett política
```

Fonte: gerada pela autora

FIGURA 27: Notícias submetidas ao algoritmo *RSLPStemmer*

```
print 'Título:', media_titles[0]
print 'Notícia:', media_text[0][:1000]
```

```
Título: psb abr diálog ensa ‘volt raízes’
Notícia: pouc trê ano após mort ex-govern pernambuc eduard camp plen campanh presidencial, psb abr canal diálog
cir gom (pdt), marin silv (rede) gerald alckmin (psdb) 2018 ensa “volt origens” centro-esquerd faz opos govern
pmdb. cúpul legend aguard próx seman deband deput governistas. dirig calcul 12 18 36 deput dev deix part cont
inu bas govern michel temer. senh psb reforç discurs volt orig centro-esquerda. psb romp ent presid dilm rouss
eff 2013, lanç eduard campos. 2015 apoi impeachment petista. viabil candidat planalto, 2014, camp inch part lev
  nom pouc nenhum afin histór ideológ legenda. é ala psb rebel cúpul partidár decid indic nom govern temer. p
dt fez ofert psb apoi cir gom consider gener cúpul partido: alianç horizontal. newslett polít receb e-mail con
teúd qual assin e-mail cadastrado! log receb melhor conteúd e-mail. funcion assim: psb apoi cir gom pd
```

Fonte: gerada pela autora

Por conseguinte, RSLP foi considerado o melhor algoritmo de *stemming* para textos no idioma português, portanto, ele foi usado nos experimentos desta pesquisa.

5.1.3 Representação do Modelo de Documento

Com todo o trabalho inicial de pré-processamento já realizado, o passo seguinte foi a geração da *Bag-of-Word*, cujo objetivo é fazer a contagem das palavras para cada instância de dados (notícias). Para isso, realizou-se a contagem da Frequência dos Termos e a Frequência dos Documentos (FD). A TF pode receber como parâmetros:

- Contagem: número de ocorrências de uma palavra em um documento
- Binário: a palavra aparece ou não aparece no documento
- Sublinear: logaritmo da frequência do termo (contagem)

Neste experimento, utilizou-se o *count* (contagem) como parâmetro de TF. E para o cálculo da FD, usou-se o IDF. Como resultado, construiu-se o vetor de características. Para a extração dos atributos, utilizou-se o modelo TF-IDF e fez o controle de modo que cada característica ocorresse no mínimo em 25% e no máximo em 85% dos documentos. Para controlar essas porcentagens, usaram-se as frequências mínima e máxima dos Termos do Documento, processo conhecido como *Pruning* (poda). Com o uso dessa técnica, palavras raras, que aparecem com menos frequência do que um determinado limite predefinido, devem ser removidas. O mesmo deve acontecer com os termos mais frequentes do que o limite determinado.

Além disso, fez-se uso da técnica de n-gramas para extrair características simples (*n=1*) ou compostas (*n=2*), de modo que, neste caso, duas palavras que aparecessem sequencialmente nas notícias ficassem unidas.

A Figura 28 ilustra o *Bag-of-Word* gerado a partir da coleção de notícias recuperada pelo *Media Frame*. Nesse exemplo, o peso foi baseado na frequência das palavras.

FIGURA 28: *Bag-of-Word* gerada a partir do corpus de notícias

Pouco mais de três anos após a morte d...	id=4888493, website=Estadão, afirmou=2.000, antes=1.000, apos=1.000, assinar=1.000, ...
Rajoy entrega fac-símile de 'Dom Quixot...	id=4888482, website=O Globo, acordo=1.000, caso=2.000, contra=1.000, crise=3.000, d...
Publicidade	id=4888597, website=Folha de São Paulo, afirmou=2.000, antes=1.000, brasil=1.000, bra...
A um ano da eleição, mais da metade d...	id=4888441, website=g1, acordo=1.000, alguns=1.000, antes=1.000, apenas=4.000, apes...
Brasília - O fundo eleitoral criado para b...	id=4888407, website=Estadão, acordo=1.000, apenas=3.000, apesar=1.000, assinar=1.00...
Guilherme Casarões, é internacionalista ...	id=4888308, website=Estadão, alguns=1.000, antes=2.000, apos=1.000, assinar=1.000, b...
Em meados do século 20, os candidatos...	id=4888228, website=Estadão, alguns=1.000, antes=3.000, apenas=1.000, assinar=1.000,...
MONTREAL (Reuters) - O Federal Reserv...	id=4888182, website=Reuters, apos=1.000, federal=1.000, politica=2.000, presidente=1....
Publicidade	id=4888646, website=Folha de São Paulo, afirmou=4.000, antes=1.000, atual=1.000, bras...
Publicidade	id=4888346, website=Folha de São Paulo, alguns=1.000, antes=1.000, apesar=1.000, con...
Macri e Obama: reunião privada em sele...	id=4888100, website=O Globo, antes=1.000, apos=2.000, casa=1.000, estado=1.000, feir...
Milhares de pessoas vão às ruas em Ma...	id=4888043, website=g1, acordo=1.000, alguns=1.000, apesar=1.000, brasilia=1.000, co...
BRASÍLIA – O líder do PT na Câmara dos...	id=4888025, website=Estadão, assinar=1.000, atual=2.000, brasileira=1.000, brasilia=1.00...
No centro de Moscou, apoiadores de N...	id=4887935, website=g1, apesar=1.000, dias=1.000, eleicoes=1.000, outras=1.000, pais=...
SÃO PAULO (Reuters) - Apoiadores do lí...	id=4887896, website=Reuters, apesar=1.000, dias=1.000, eleicoes=1.000, outras=1.000, ...
10h24	id=4887847, website=O Globo, apesar=1.000, dias=1.000, eleicoes=1.000, outras=1.000, ...

Fonte: Gerada pela autora

Observa-se que todas as palavras da coleção de notícias foram associadas a um índice numérico. Portanto, nesta etapa, as notícias pré-processadas foram convertidas em vetores numéricos.

5.1.4 Medida de Similaridade

Para encontrar a similaridade entre os documentos, as notícias foram primeiramente pré-processadas e vetorizadas usando o modelo TF-IDF. Uma vez que se tem as representações vetoriais da coleção, pode-se calcular a semelhança entre os documentos usando alguma medida de distância ou de similaridade. As medidas de Similaridade do Cosseno, Distância de Manhattan e Distância Euclidiana foram usadas nos experimentos deste trabalho para calcular a semelhança entre os documentos de texto, ou seja, entre as notícias. A Figura 29 ilustra um recorte da matriz de similaridade N x N gerada a partir do corpus usado nesta pesquisa, sendo N a quantidade notícias.

FIGURA 29: Matriz de similaridade

	0	1	2	3	4	5	6	7	8	9	...
0	1.000000	0.559975	0.482199	0.267122	0.572259	0.297291	0.261501	0.299889	0.334151	0.815508	...
1	0.559975	1.000000	0.495424	0.276313	0.747727	0.484642	0.396569	0.545280	0.690999	0.488763	...
2	0.482199	0.495424	1.000000	0.237066	0.417089	0.681318	0.423683	0.640504	0.497783	0.552901	...
3	0.267122	0.276313	0.237066	1.000000	0.239216	0.186067	0.168454	0.310417	0.208009	0.260942	...
4	0.572259	0.747727	0.417089	0.239216	1.000000	0.619284	0.395347	0.513364	0.633694	0.586454	...
5	0.297291	0.484642	0.681318	0.186067	0.619284	1.000000	0.511186	0.737821	0.659812	0.445523	...
6	0.261501	0.396569	0.423683	0.168454	0.395347	0.511186	1.000000	0.433963	0.399988	0.319063	...
7	0.299889	0.545280	0.640504	0.310417	0.513364	0.737821	0.433963	1.000000	0.753642	0.294286	...
8	0.334151	0.690999	0.497783	0.208009	0.633694	0.659812	0.399988	0.753642	1.000000	0.346676	...
9	0.815508	0.488763	0.552901	0.260942	0.586454	0.445523	0.319063	0.294286	0.346676	1.000000	...

Fonte: Gerada pela autora

A próxima subseção descreve o processo de agrupamento. Testaram-se os algoritmos *k-means*, *Affinity Propagation* e os modelos hierárquicos.

5.1.5 Processo de agrupamento

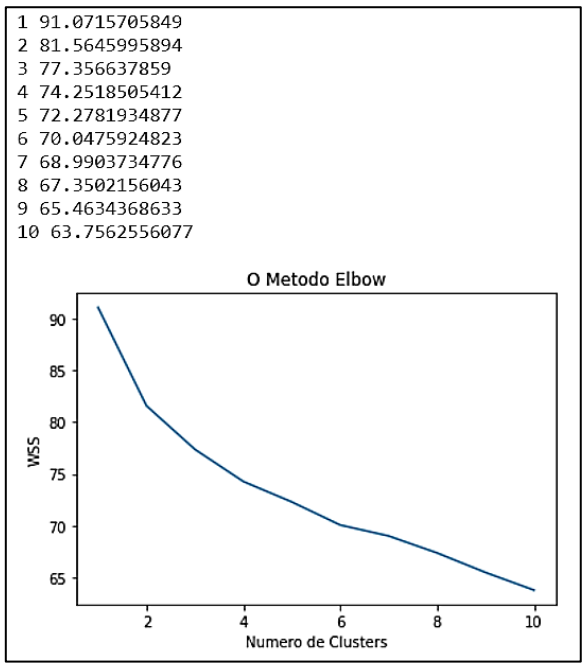
É na etapa de *clustering* que os grupos são formados. Assim, com o intuito de verificar o desempenho das técnicas, vários testes foram realizados, começando com o *k-means*. Como esse algoritmo exige que o usuário forneça o valor de k, utilizou-se o método *Elbow* para descobrir o número ideal de *clusters*.

5.1.5.1 *Clustering* das notícias usando o algoritmo *k-means*

I) Cálculo do valor de K

O método Elbow foi usado para encontrar o melhor valor de k, realizou-se uma repetição de 10 interações e foi calculado a cada interação da repetição, a variância dos dados inseridos no conjunto. A Figura 30 mostra o gráfico gerado pelo método *Elbow*.

FIGURA 30: Gráfico gerado pelo método *Elbow*



Fonte: Elaborada pela autora

O eixo Y representa o WSS (*Within Sum of Square*) e o eixo X, o Número de *Clusters*. A Figura 30 mostra que não houve grande variação na curva do gráfico, a maior variância está na quantidade de *clusters* de 1 para 2. Isso se justifica pelo fato da amostra de notícias ser relacionada a apenas um assunto, ou seja, política. Mesmo assim, forçou-se o teste com valores de k diferentes do que foi determinado pelo método *Elbow* para tentar subdividir as notícias relacionada à política em subcategorias e assim, analisar os resultados.

Mesmo com o *Elbow* indicando um número muito baixo de clusters (1 ou 2), neste experimento, optou-se por testar outros valores para, assim, analisar os resultados. Desse modo, testou-se o algoritmo *k-means* três vezes, usando diferentes configurações para k (número de agrupamentos) e n (número de termos). No primeiro teste, utilizaram-se k=4 e n=5, no segundo, k=3 e n=4 e no último, mantiveram-se as quatro características e o número de *clusters* foi reduzido para dois. Quanto ao valor de n, no primeiro teste escolheu aleatoriamente o valor cinco (5) por acreditar que essa quantidade de características pode representar bem um assunto, mas a partir dos testes, observou-se que um número muito alto para n pode apresentar características pouco significativas e um número muito baixo pode ter pouco poder de discriminação.

II) Teste do *k- means* (k=4 e *n*=5)

O Quadro 10 mostra as características extraídas para cada *cluster*.

QUADRO 10: Características extraídas pelo *k-means* – 1º experiemento (k=4, n=5)

Clusters	**Características**
Cluster 0	brasil, rio, país, lei, mundo
Cluster 1	presid, governo, país, pública, cidad
Cluster 2	candidato, deputado, partido, eleitor, presid
Cluster 3	mail, conteúdo, acordo, empresa, receber

Fonte: Elaborado pela autora

Ao analisar as características (*features*) extraídas para a formação dos *clusters*, nota-se que ao forçar o algoritmo a distribuir a coleção de notícias em grupos maiores do que o ideal, alguns atributos se repetiram, como por exemplo, a palavra "**país"** apareceu nos *clusters* 0 e 1 e a palavra "**presidente"** apareceu nos grupos 1 e 2. Isso se deve ao fato dessas palavras aparecerem simultaneamente em várias notícias, mas a questão da repetição do termo "país" foi resolvida ao reduzir o número de características para quatro (4). Porém, o vocábulo "presidente" apareceu nos *clusters* 2 e 3. O Quadro 11 apresenta os termos de cada grupo ao reduzir o valor de n para quatro (4).

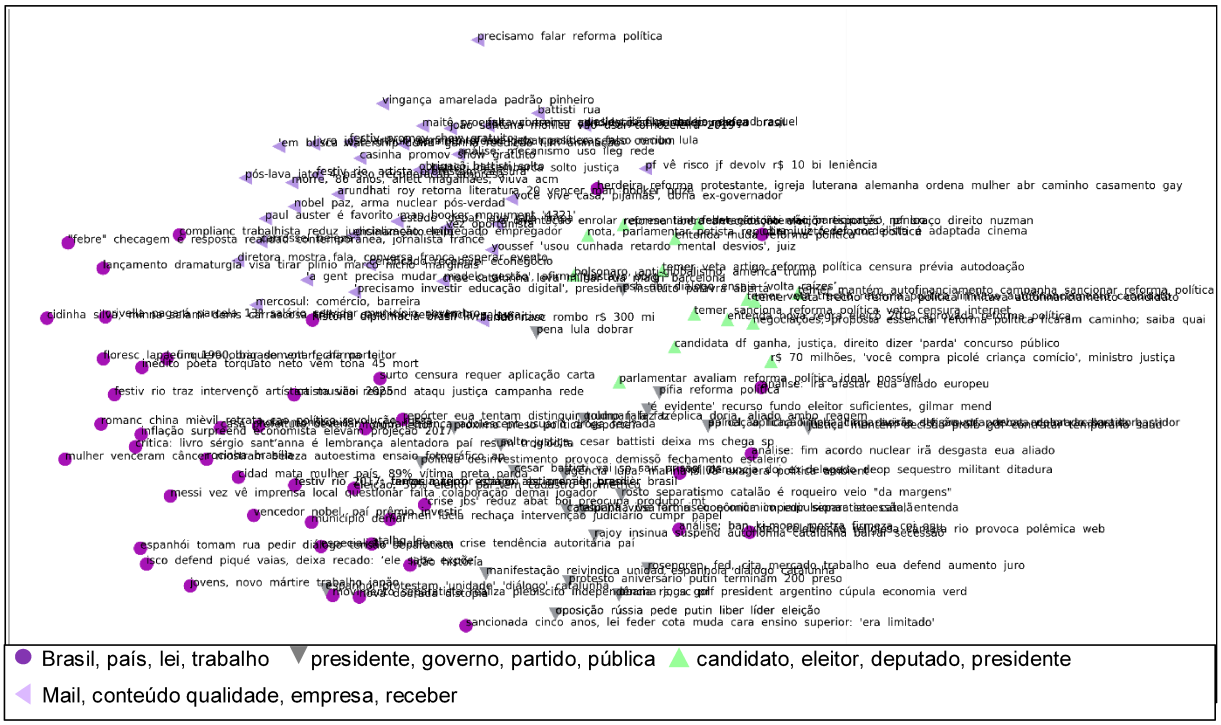
QUADRO 11: Características extraídas pelo *k-means* – 1º experimento (k=4, n=4)

Clusters	**Características**
Cluster 1	Brasil, lei, país, trabalho
Cluster 2	Presidente, governo, partido, pública
Cluster 3	Candidato, eleitor, deputado, presidente
Cluster 4	Mail, conteúdo qualidade, empresa, receber

Fonte: Elaborado pela autora

Ao diminuir a quantidade de *factures*, nem sempre elas continuam as mesmas, mas é claramente observável que as palavras de cada grupo são relacionadas, ou seja, tem a ver com o mesmo assunto. A Figura 31 esboça os quatros *clusters* formados pelo algoritmo *k-means* ao ser alimentado com um corpus composto por 123 notícias.

FIGURA 31: Diagrama de dispersão dos *clusters* de notícias (k=4 e n=4)



Fonte: Elaborada pela autora

A legenda apresenta as 4 características de cada aglomerado (*labels*).

A Figura 31 ilustra o diagrama de dispersão em que os pontos são agrupados com base na similaridade das notícias. O algoritmo formou quatro (4) *clusters*, conforme configuração, porém, os pontos no gráfico ficaram bastante espalhados mostrando que os elementos de grupos diferentes não são tão heterogêneos e objetos de um mesmo grupo não são tão semelhantes. A maior concentração de documentos foi no *Cluster 1*. Esse grupo apresentou como descritores as palavras “Brasil”, “lei”, “pais” e “trabalho”. As notícias que relatam “**mercado de trabalho**” e “**leis trabalhistas**” ficaram alocadas nesse grupo. No entanto, esse aglomerado foi o que mais apresentou assuntos diversificados, mas todos relacionados com o tema do corpus, ou seja, política. Já o *Cluster 2* englobou matérias cujos termos “**eleitor**”, “**eleição**”, “**protesto**”, “**oposição**” e “**manifestação**” estavam presentes no conteúdo dos informes. O *Cluster 3* concentrou as notícias sobre “**Reforma Política**” e “**presidente Michel Temer**” pois essas expressões apareceram muitas vezes nos textos desse aglomerado. Já o *Cluster 4* uniu as notícias relacionadas a assuntos **artístico** e **cultural**. Expressões como “Nobel da Paz”, “Prêmio Internacional Man Booker”, “livro”, “escritor”, “romances”, “literatura”, “Maitê Proença”, “Lançamento dramaturgia” e “intervenção artística” foram evidenciadas nas notícias desse grupo. Nesse caso, o algoritmo teve um bom

desempenho ao agrupar por similaridade, porém as características não foram tão representativas em relação às notícias desse *cluster*.

Por conseguinte, por se tratar de textos jornalísticos, a notícia necessita ser relevante, ou seja, tratar de assuntos que sejam atraentes para o leitor e cujo objetivo é informá-lo de algum acontecimento da atualidade que seja digno de ser noticiado. Assim sendo, pode-se dizer que a notícia é um texto de caráter temporal, diretamente ligado aos fatos recentes. Isso faz com que ao pesquisar por matérias, publicadas em um determinado período, sobre um tema geral como, por exemplo, "política", mesmo se tratando do termo usado na busca e que estava em alta no momento da pesquisa, os informes recuperados vão tratar de assuntos diversificados. Essa falta de padronização, pertinente com esse tipo de texto, dificulta a formação de grupos com alta similaridade entre os documentos de cada cluster. Fato que aconteceu ao tentar subdividir o corpus com notícias sobre um único assunto em quatro (4) *clusters*. Desta forma, realizou-se um novo teste reduzindo o valor de *k* para 3, pois os *clusters* 2 e *3* deveriam ser mesclados, pois as características de ambos mostraram uma forte semelhança das notícias desses dois aglomerados.

III) Teste do k- means (k=3 e *feature*=4)

Ao reduzir o número de grupos para três, mantendo as quatro características, observa-se que os *clusters* 1 e 2, conforme apresentado no teste anterior, podem ser unificados se for levado em consideração apenas os descritores extraídos. No entanto, uma leitura do conteúdo das notícias se faz necessária para ter uma conclusão mais assertiva. O Quadro 12 apresenta com mais detalhes as *features* extraídas.

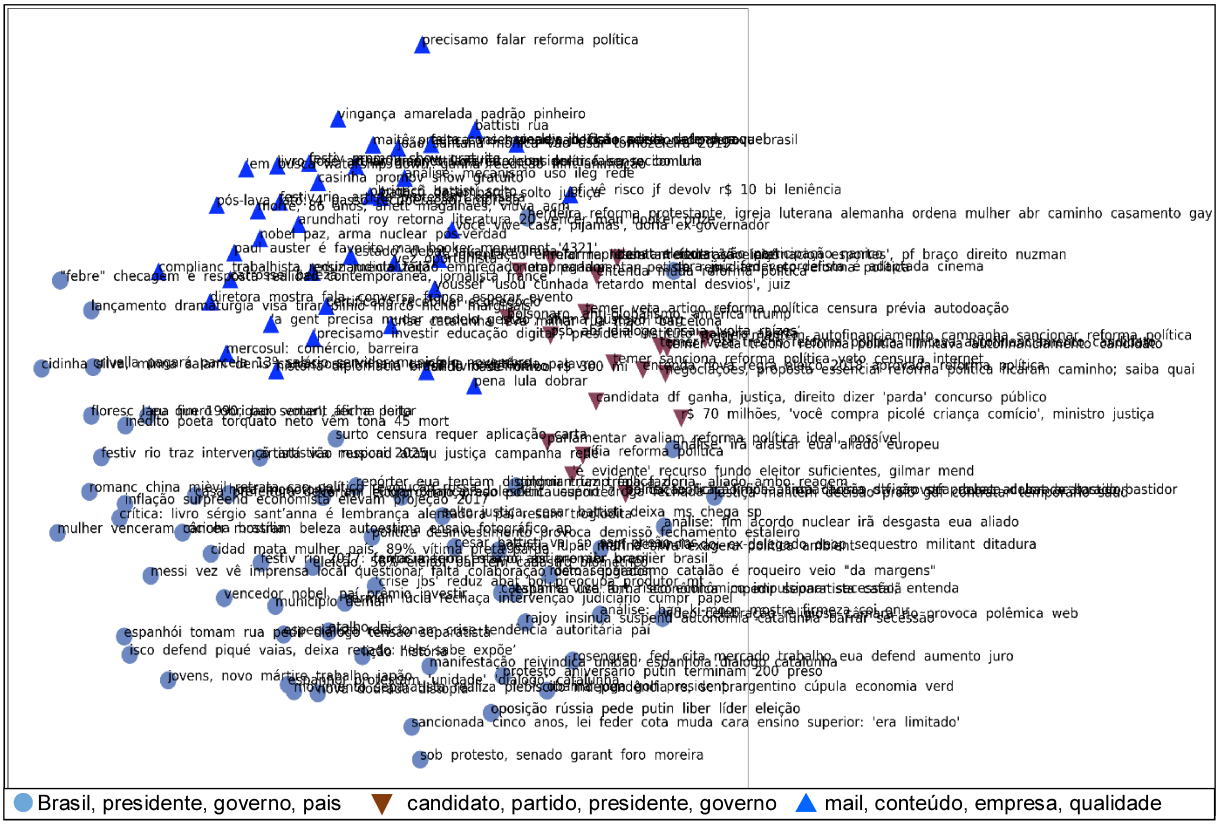
QUADRO 12: Características extraídas pelo *k-means* – 1º experimento (k=3, n=4)

Clusters	**Características**
Cluster 1	Brasil, presidente, governo, pais
Cluster 2	candidato, partido, presidente, governo
Cluster 3	mail, conteúdo, empresa, qualidade

Fonte: Elaborado pela autora

A Figura 30 mostra que os pontos no gráfico de dispersão ficaram menos espalhados, se comparados com o diagrama da Figura 32.

FIGURA 32: Diagrama de dispersão dos *clusters* de notícias (k=3 e n=4)



Fonte: Elaborada pela autora.

O gráfico de dispersão, Figura 32, mostra os três *clusters* formados, quanto mais próximos estão os elementos de um grupo, mais semelhantes eles são.

É importante ressaltar que os grupos se formaram automaticamente, sem intervenção do usuário, sem considerar previamente as características dos documentos e sem usar uma coleção de testes com rótulos conhecidos para direcionar o agrupamento, conforme ocorre na classificação. Assim, a técnica agrupou as notícias com base nas informações existentes, de modo que os informes de um grupo sejam os mais semelhantes possíveis. No caso desse teste, ao dividir a coleção em três grupos, observou-se que o algoritmo conseguiu agrupar, porém, pontos pertencentes a um mesmo grupo ficaram bem distantes. Isso se deve à natureza do corpus, pois, por se tratar de apenas um assunto, o programa encontrou dificuldades em extrair características representativas para formar 3 *clusters*. Por conseguinte, optou-se por realizar o experimento novamente reduzindo o número de grupos.

IV) Teste do *k- means* (k=2 e n=4)

Por fim, para avaliar o número ideal de *cluster* calculado pelo *Elbow*, neste experimento, utilizou-se *k=2* e mantiveram-se as quatro características. O objetivo foi verificar se o *k-means* apresenta um bom resultado ao separar as notícias relacionadas à política em

duas subcategorias. Observa-se no Quadro 13 que as características extraídas para cada grupo têm forte relação entre si e se comparadas em *clusters* diferentes, elas são pouco relacionadas.

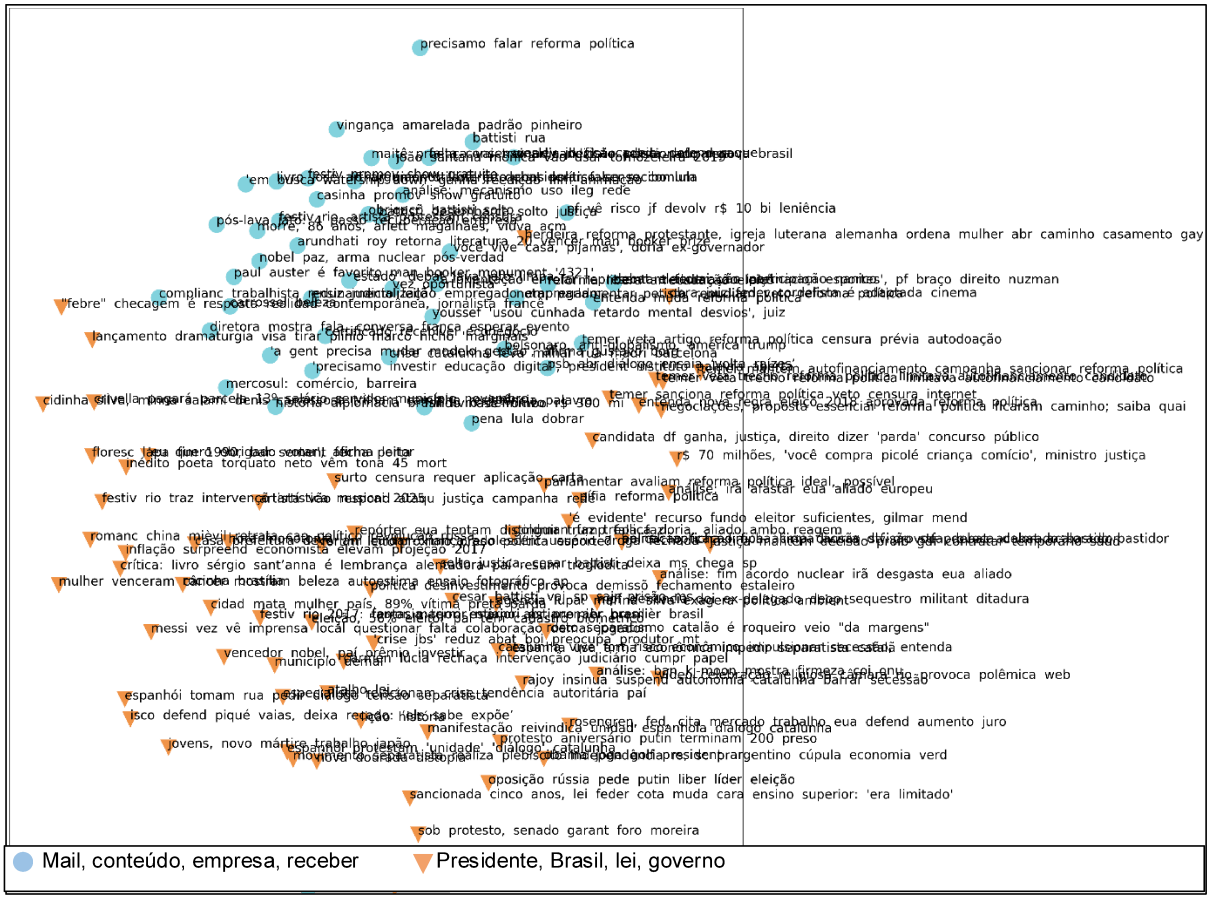
QUADRO 13: Características extraídas pelo *k-means* – 1º experimento (k=2, n=4)

Clusters	**Características**
Cluster 1	mail, conteúdo, empresa, receber
Cluster 2	presid, país, candidato, governo

Fonte: Elaborado pela autora

Os termos extraídos para o *Cluster* 2 são palavras correlacionadas ao tema política, já no *Cluster* 1, as características não são tão representativas em relação ao assunto do corpus, porém ambos os grupos são formados por notícias sobre política. A Figura 33 apresenta o agrupamento formado usando k=2 e n=4.

FIGURA 33: Diagrama de dispersão dos *clusters* de notícias (k=2, n=4)



Fonte: Elaborada pela autora

Conforme aconteceu ao tentar agrupar a coleção em quatro (4) e em três (3) grupos, o resultado não foi muito diferente ao usar *k=2*. As características extraídas para o *Cluster 1* não foram representativas, mas o algoritmo fez a divisão incluindo no *Cluster 2* as notícias que apresentavam um maior número de termos que são fortemente relacionados à política, tais como “presidente”, “Brasil”, “governo”, “eleição”, “parlamento” e “reforma política”. E o *Cluster 1* englobou as notícias que, apesar de também ter relação com esse tema, os termos de maior frequência têm uma relação mais fraca com o assunto do corpus como, por exemplo, as palavras “e-mail”, “empresa”, “trabalhista”, “trabalhador” e “empregado”.

Em virtude dos fatos mencionados e para uma melhor análise, os algoritmos *Affinity Propagation* e os modelos hierárquicos também serão testados com essa amostra, mas novos experimentos serão realizados utilizando uma coleção de notícias mais diversificada.

5.1.5.2 C*lustering* das notícias usando o algoritmo *Affinity Propagation*

Usando o corpus já pré-processado anteriormente para o agrupamento com o *k-means*, o *Affinity Propagation*, que calcula o número de k sem intervenção do usuário, subdividiu as notícias em 18 *clusters*. O Quadro 14 apresenta as características extraídas.

QUADRO 14: Características extraídas pelo *Affinity Propagation*

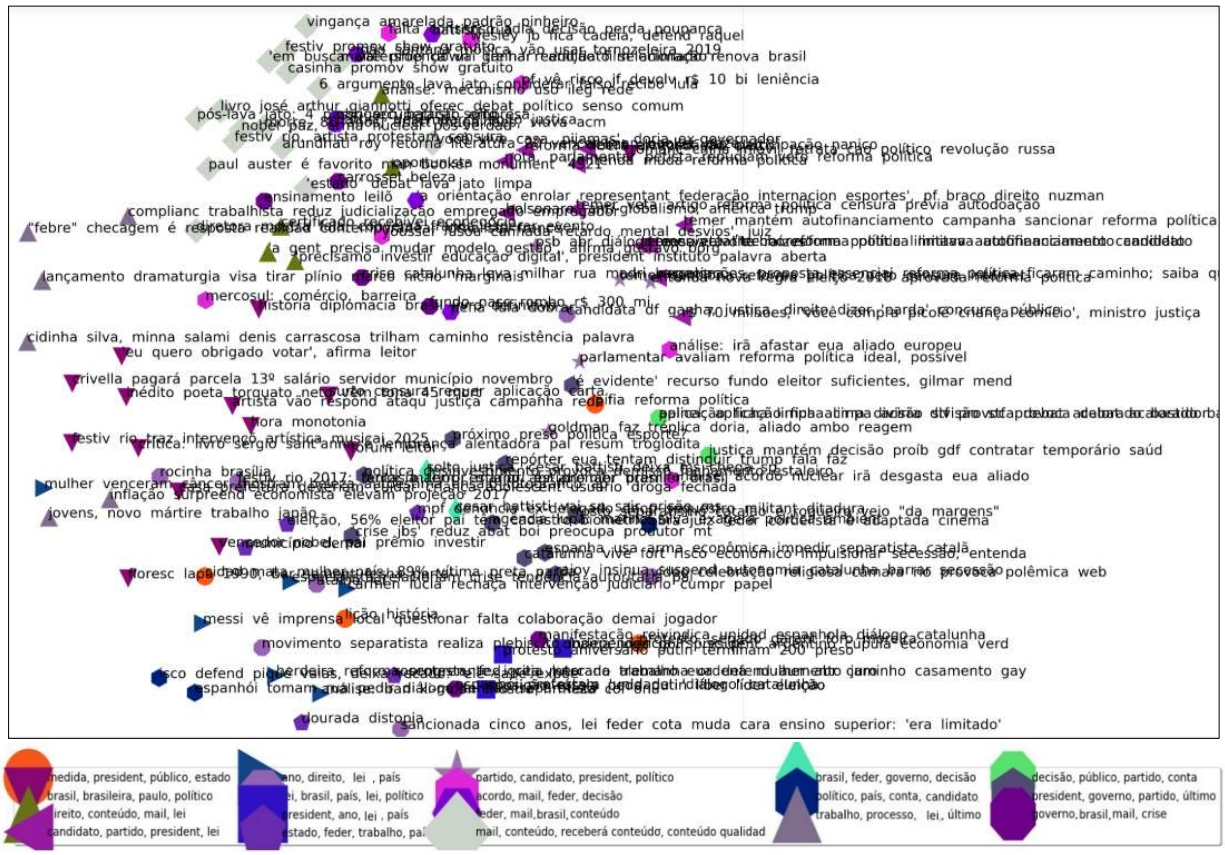
	Características
Cluster 1	Medida, presidente, público, estado
Cluster 2	Brasil, brasileiro, Paulo, política
Cluster 3	Direito, conteúdo, mail, lei
Cluster 4	Candidato, partido, presidente, lei
Cluster 5	Ano, direito, lei, país
Cluster 6	Lei, Brasil, político, país
Cluster 7	Presidente, ano, lei, política
Cluster 8	Estado, federal, trabalho, país
Cluster 9	Partido, candidato, presidente, político
Cluster 10	Acordo, email, federal, decisão
Cluster 11	Federal, mail, Brasil, conteúdo
Cluster 12	Mail conteúdo, receberá conteúdo, conteúdo qualidade
Cluster 13	Brasil, federal, governo, decisão
Cluster 14	Político, país, conta, candidato
Cluster 15	Trabalho, processo, lei, último
Cluster 16	Decisão, público, partido, conta
Cluster 17	Presidente, governo, partido, último
Cluster 18	Governo, Brasil, mail, crise

Fonte: Elaborada pela autora

Fazendo uma análise somente das características extraídas, nota-se que o *Affinity Propagation* dividiu o corpus em muitos grupos não conseguindo separar as notícias em subcategorias bem definidas visto que muitos descritores repetiram em vários *clusters*.

A Figura 34 ilustra os dezoitos (18) *clusters* formados pelo *Affinity Propagation*.

FIGURA 34: **Diagrama de dispersão usando o *Affinity Propagation***



Fonte: Elaborada pela autora

O número de grupos identificado por esse algoritmo foi muito superior ao valor encontrado pelo método *Elbow*. Além disso, sabe-se que a coleção é formada por notícias referentes a apenas um assunto, o que não justifica uma quantidade grande de aglomerados. Porém, percebe-se que há certa regularidade nos termos agrupados, bem como em suas ocorrências. Mas, apesar de, aparentemente, o *Affinity Propagation* ter realizado o agrupamento, os *clusters* encontrados foram irregulares, pontos de grupos diferentes misturam-se e pontos de um mesmo grupos ficaram distantes. Além disso, notou-se que muitos *clusters* possuem alto potencial para serem mesclados. Para uma melhor avaliação dessa técnica, é necessária uma análise mais detalhada do conteúdo das notícias. Portanto, novos testes serão feitos usando um corpus menor, formado por uma amostra de notícias com assuntos mais variados.

5.1.5.3 Cluster Hierárquico

Neste experimento, utilizou-se o algoritmo *Hierarchical Clustering* com os métodos aglomerativos *Single Linkage*, *Average Linkage*, C*omplete Linkage* e o *Ward* para verificar o comportamento dos textos utilizando uma abordagem hierárquica. Como a coleção de notícias já estava representada no modelo Espaço Vetorial, etapa realizada nas experiências anteriores, realizaram-se os testes variando apenas a medida de similaridade e o método de agrupamento.

I) Calculo da Distância

Utilizaram-se nos experimentos desta pesquisa as medidas de similaridade do **Cosseno, Manhattan** e **Euclidiana** para calcular a distância entre os documentos.

II) *Clustering*

Para agrupar as notícias, testaram-se 4 métodos de agrupamento hierárquico:

***- Single linkage*:** calcula a distância entre os elementos mais próximos dos dois *clusters*;

***- Average linkage*:** calcula a distância média entre os elementos dos dois *clusters*;

***- Complete linkage*:** calcula a distância entre os elementos mais distantes dos *clusters*;

- Ward's method: a medida de distância entre dois *clusters* é a soma das distâncias ao quadrado entre os dois aglomerados. Minimiza a variação total dentro do *cluster* e em cada etapa, os pares de *clusters* com a distância mínima entre os grupos são mesclados.

O Quadro 14 apresenta as combinações realizadas entre os métodos de agrupamentos hierárquicos e as medidas de similaridade.

QUADRO 15: Método agrupamento x medida de similaridade

Experimento	Método de agrupamento	Medida da Similaridade	Nº de grupos formados
1º Teste	*Single Linkage*	Distância Euclidiana	Não agrupou
		Similaridade de Cosseno	Não agrupou
		Distância de Manhattan	Não agrupou
2º Teste	*Average Linkage*	Distância Euclidiana	4 *clusters* primários
		Similaridade de Cosseno	Não agrupou
		Distância de Manhattan	Não agrupou
3º Teste	*Complete Linkage*	Distância Euclidiana	2 *clusters* primários
		Similaridade de Cosseno	Não agrupou
		Distância de Manhattan	3 *clusters* primários
4º Teste	*Ward*	Distância Euclidiana	2 *clusters* primários
		Similaridade de Cosseno	3 *clusters* primários
		Distância de Manhattan	3 *clusters* primários

Fonte: Elaborado pela autora

Conforme apresentado no Quadro 15, no 1º teste, o *Single Linkage* não conseguiu formar grupos, independente da medida de similaridade adotada. O *Average Linkage* compôs 4 grupos primários usando a Distância Euclidiana e não conseguiu agrupar ao usar a Distância de Manhattam e a Similaridade de Cosseno. No 3º teste, utilizou-se o *Complete Linkage* que formou dois *cluters* primários usando a Distância Euclidiana e três grupos primários ao usar a Distância de Manhattam, porém não conseguiu organizar as notícias em grupos ao usar a Similaridade de Cosseno. No 4º teste, utilizou-se o método Ward que conseguiu compor 3 grupos primários usando a Distância de Manhattam e a Similaridade de Cosseno e 2 grupos no primeiro nível usando a Distância Euclidiana.

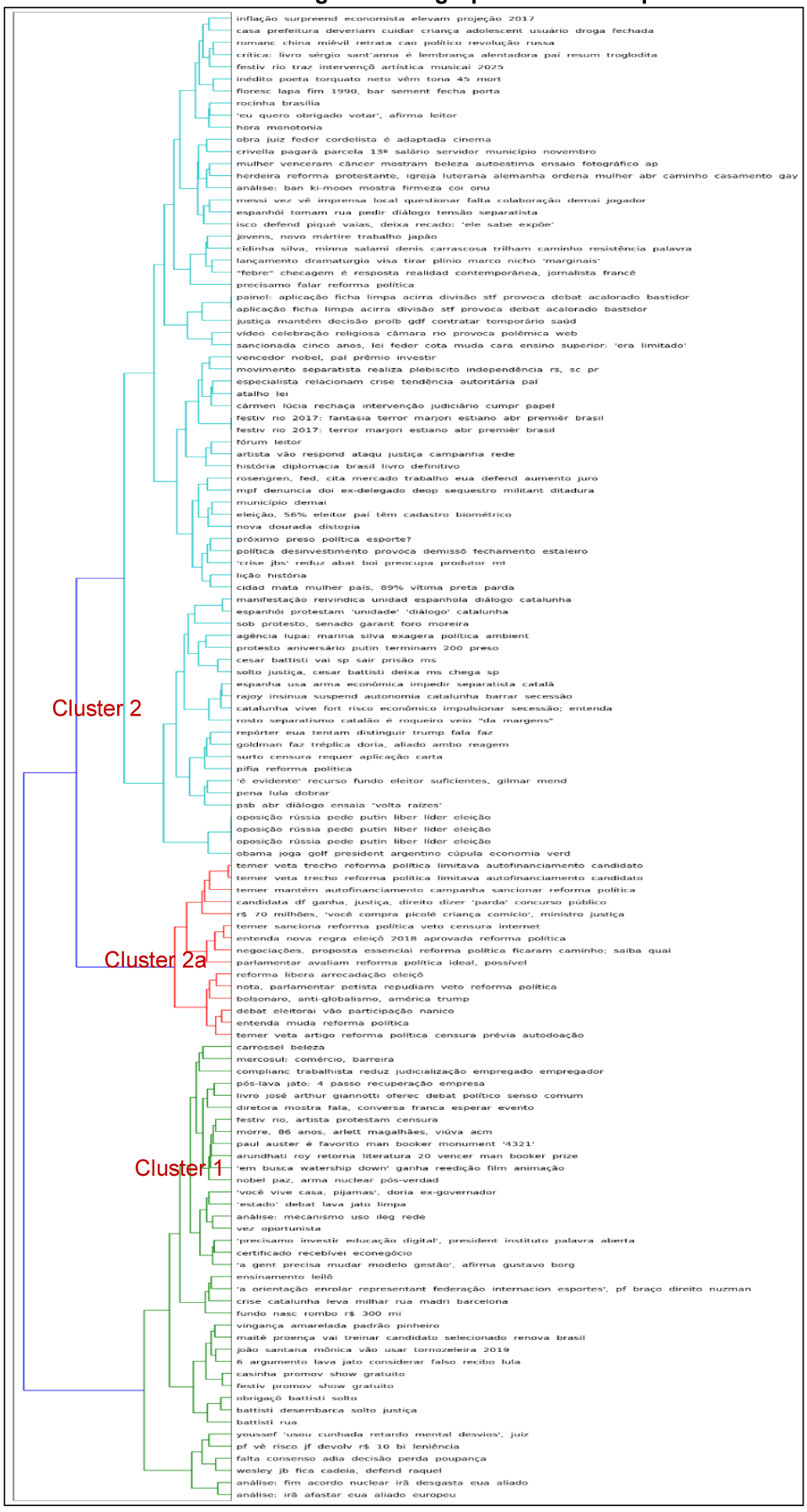
Por conseguinte, neste primeiro experimento, usando uma coleção de 123 notícias relacionadas à política, o método Ward foi o algoritmo hierárquico que obteve o melhor desempenho para esse tipo de corpus, sendo que a medida de similaridade adotada não fez muita diferença no resultado final.

III) Visualização do *Clustering* Hierárquico

A Figura 35 mostra o dendrograma plotado ao realizar o teste usando o método de agrupamento hierárquico *Ward* e a Similaridade de Cosseno, medida usada para calcular a distância entre as notícias.

Percebe-se que termos relacionados à "política" estão presentes em todos os grupos. Os vocábulos “"politica", "presidente", "candidato", "deputado", "governo", "reforma", "eleitor" e "campanha" foram as palavras que aparecerem com mais frequência em toda a coleção. Isso faz sentido porque o corpus em estudo foi coletado no período das campanhas para presidente do Brasil em 2018.

FIGURA 35: Dendograma do agrupamento hierárquico

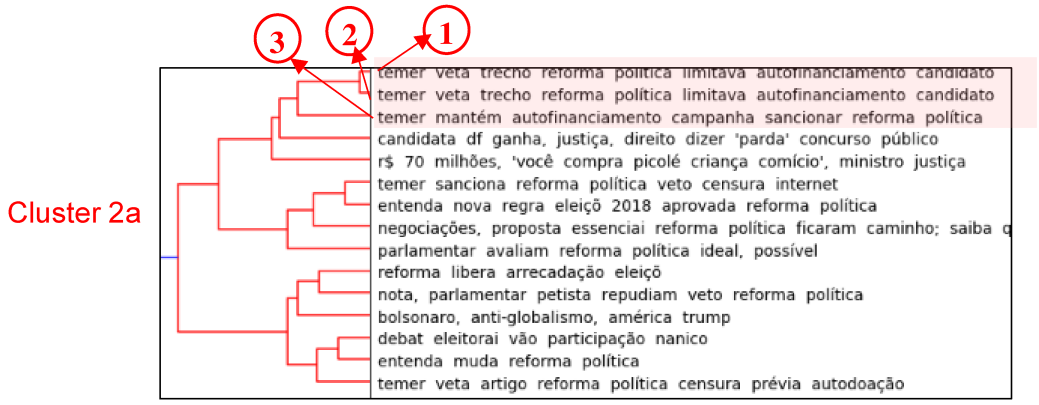


Fonte: Gerada pela autora

Observa-se na Figura 35 que primeiramente o algoritmo dividiu o corpus em três grupos. O método retornou dois *clusters* primários e o maior grupo foi dividido em duas subcategorias principais, sendo que a maior parte das notícias concentrou em apenas um deles. Isso se justifica pelo fato do corpus ser composto por notícias semelhantes, o que dificultou a criação de subcategorias bem definidas.

O menor grupo (*Cluster 2a*) foi o que conseguiu formar um aglomerado com notícias mais similares, pois a maioria das matérias concentradas nele são sobre "**reforma política"**. A Figura 36 expande o *Cluster 2* para melhor visualização dos títulos.

FIGURA 36: Recorte de um cluster formado pelo algoritmo hierárquico



Fonte: Elaborada pela autora

Nota-se que as três primeiras notícias do *Cluster 2a* são muito similares, por isso elas estão próximas. A altura da barra da terceira notícia que faz a ligação com as duas primeiras é maior do que a altura da barra que liga as notícias 1 e 2. Isso é porque a distância entre a notícia 3 e 2 é maior do que a distância entre as notícias 2 e 1. Quanto menor a distância, mais similares são os documentos. Ao mesmo tempo, ao traçar uma linha vertical no último nível da Figura 35, verifica-se uma quantidade grande de grupos formados. Isso se deve ao fato de não existir grupos com uma quantidade maior de notícias semelhantes, pois, apesar do tema geral ser o mesmo, as matérias discutem assuntos diversos. Dessa forma, são necessários mais experimentos usando uma nova coleção para melhor análise.

5.1.6 Validação dos resultados do 1º experimento

Em relação ao primeiro experimento, realizou-se a avaliação do melhor número de *clusters*, usando o Algoritmo *k-means,* através do Coeficiente da Silhueta. Fez-se o teste com **k** variando de 2 a 8. Observa-se na Tabela 5 que o modo de inicialização do algoritmo (K-Means++ ou Inicialização randômica) não fez muita diferença no cálculo.

TABELA 5: Avaliação do *k-means* usando o coeficiente da silhueta – 1º experimento

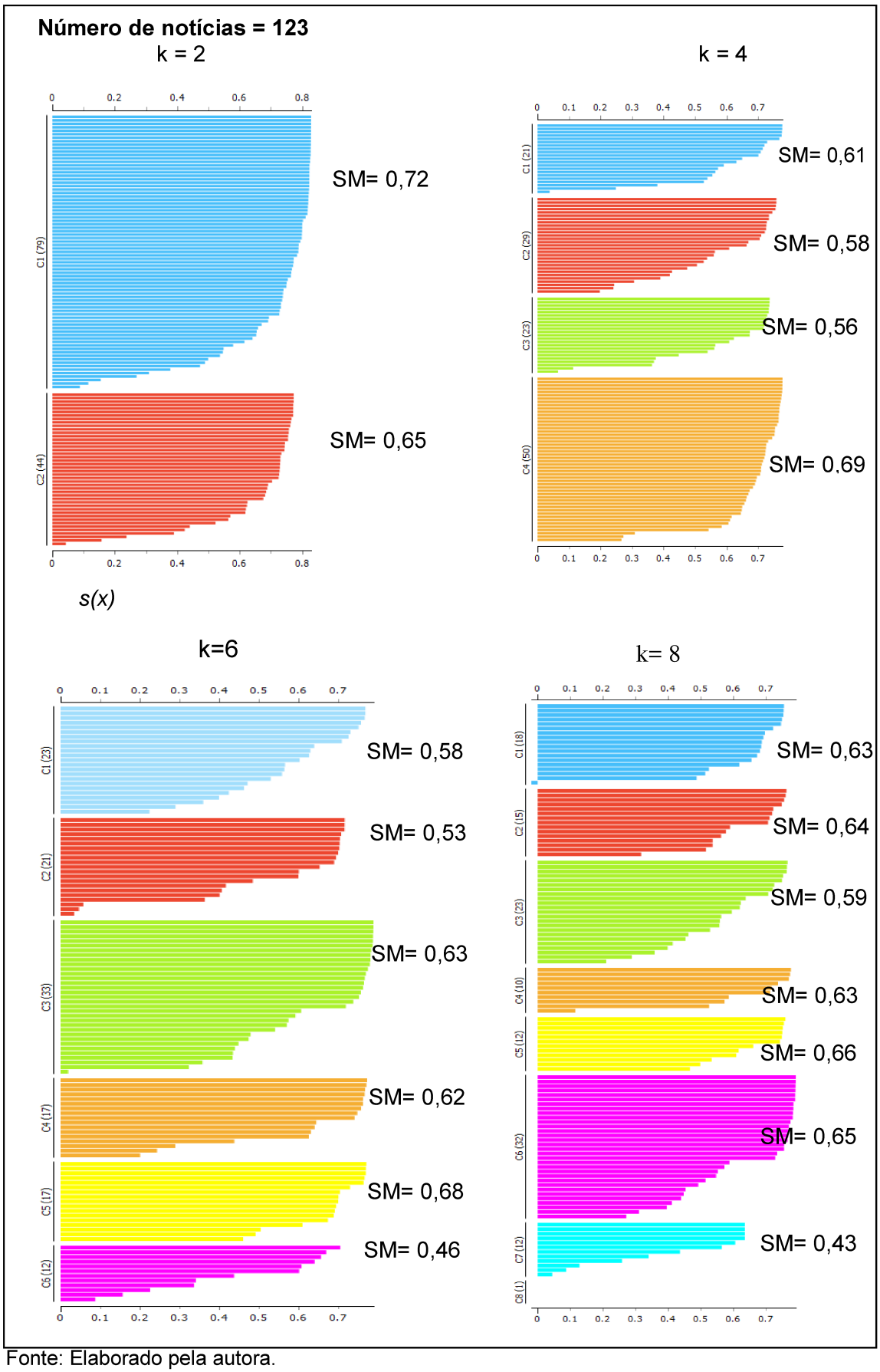
Método de avaliação		
Nº de *clusters*	Silhouette	
	Inicialização com k-Means++	Inicialização randômica
2	0,694	0,694
3	0,621	0,623
4	0,629	0,629
5	0,594	0,592
6	0,590	0,590
7	0,586	0,597
8	0,606	0,592

Fonte: Elaborada pela autora

O gráfico da silhueta é uma forma de avaliar o particionamento. Desse modo, todo o agrupamento é exibido combinando as silhuetas em um único gráfico, possibilitando, assim, a avaliação da qualidade relativa dos *clusters*. Cada observação é representada pelo valor *s(xi),* chamado de silhueta, que é baseado na comparação da consistência e na separação de cada grupo (MAXIMILIANO; CORDEIRO, 2008). A largura média da silhueta fornece uma avaliação da validação do agrupamento e pode ser utilizada para selecionar o número ideal de grupos.

A Figura 37 apresenta os gráficos da silhueta para k=2, k=4, k=6 e k=8. O eixo vertical representa os *n* documentos e o horizontal representa o valor da silhueta para cada objeto (notícia). As notícias estão divididas em grupos e ordenadas em ordem decrescente do valor da silhueta. Fez-se o teste usando a Distância Euclidiana e de Manhattan, porém a medida utilizada, nesse caso, não fez diferença.

FIGURA 37: Gráficos da silhueta para diferentes valores de k – 1º experimento



Fonte: Elaborado pela autora.

Verifica-se no gráfico 37 que, para k=4, encontraram-se as médias dos coeficientes da silhueta para cada grupo:

Cluster 1: 21 notícias – valor médio de silhueta: 0,61

Cluster 2: 29 notícias – valor médio de silhueta: 0,58

Cluster 3: 23 notícias – valor médio de silhueta: 0,56

Cluster 4: 50 notícias – valor médio de silhueta: 0,69

Como o coeficiente foi aproximadamente 0.6, significa que uma estrutura razoável foi encontrada, com base nos valores da silhueta apresentados no Quadro 7.

A verificação manual dos documentos em cada *cluster* revelou que a taxa de acerto do *k-means* foi de apenas 60 %. A precisão não foi significativamente alta, mas por ser um algoritmo não supervisionado e que não tem acesso às informações adicionais sobre a amostra, conforme acontece com as técnicas de aprendizado supervisionado, pode-se afirmar que o resultado foi relativamente bom. Assim, em situações em que as informações de verdade básica são limitadas ou inexistentes, a análise de *clusters* pode fornecer informações valiosas sobre os padrões dos dados.

5.2 Experimento 2: Teste com um corpus de 107 notícias relacionadas aos temas política, educação, saúde e economia

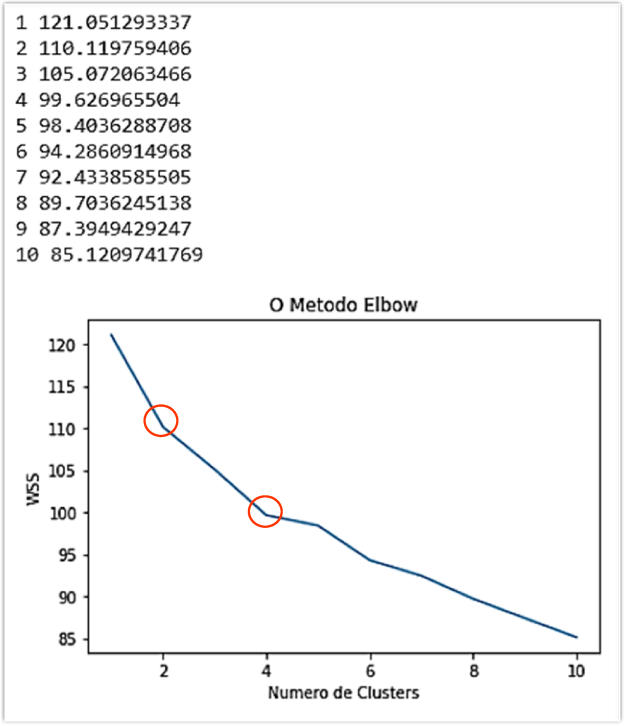
Na realização deste experimento, os termos “política”, “educação”, “saúde” e “economia” foram usados como palavras-chave para recuperação das notícias usando o *Media Frame*. Após coleta do corpus, repetiram-se todos os passos relacionados ao pré-processamento discutido anteriormente. Por isso, fez-se a descrição do segundo teste a partir da fase de *clustering*.

5.2.1 *Clustering* das notícias usando o algoritmo *k-means*

I) Cálculo do valor de K

O primeiro passo foi analisar, através do método *Elbow,* qual seria o valor ideal de k para o corpus em estudo. A Figura 38 mostra que não houve um ponto em que a inércia reduziu bruscamente de forma a determinar o número ideal de c*lusters*, porém, pode-se ver uma queda em k=2 e k=4.

FIGURA 38: Cálculo do valor de K pelo *Elbow* – 2º experimento



Fonte: Elaborada pela autora

O valor ideal do número de *cluster*, visto que a coleção de notícias é composta por quatro temas distintos, seria k = 4. Assim, fez-se o teste com k=2 e k=4, conforme indicado pelo *Elbow*.

I) Teste do *k- means* (k=4 e n=5)

O Quadro 16 mostra as características extraídas para cada *cluster* usando na configuração k = 4 e n = 5.

QUADRO 16: Características extraídas pelo *k-means* – 2º experimento (k=4, n=5)

Clusters	**Características**
Cluster 1	estado, trabalho, projeto, público, educação
Cluster 2	reforma, economia, governo, estado, brasil
Cluster 3	Brasil, acordo, política, brasil, governo
Cluster 4	bolsonaro, governo, presidente, ministro, jair

Fonte: Elaborado pela autora

Ao analisar os termos extraídos para cada grupo, nota-se que eles têm forte relação, o que justifica pertencerem ao mesmo *cluster*. Porém, algumas palavras, conforme aconteceu no primeiro experimento, repetiram em mais de um aglomerado. Mesmo reduzindo o número de atributos para 4, esse problema não foi resolvido. Mas isso pode realmente acontecer, pois

uma palavra relevante pode, realmente, ser representativa em mais de um grupo. O Quadro 17 apresenta as características extraídas para n=4.

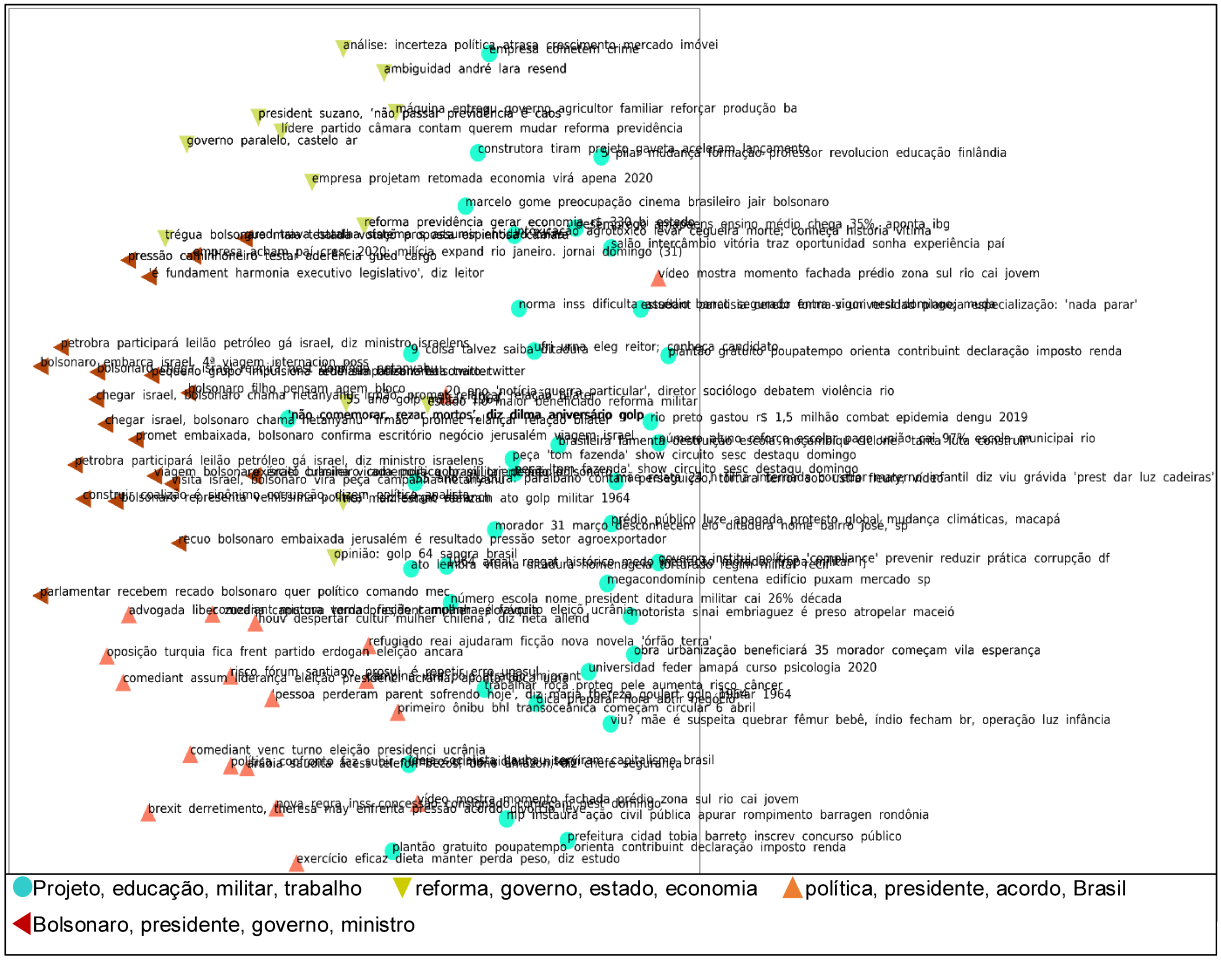
QUADRO 17: Características extraídas pelo *k-means* – 2º experimento (k=4, n=4)

Clusters	Características
Cluster 1	Projeto, educação, militar, trabalho
Cluster 2	reforma , governo, estado, economia
Cluster 3	Política, presidente, acordo, Brasil
Cluster 4	Bolsonaro, presidente, ministro, governo

Fonte: Elaborado pela autora

Ao diminuir a quantidade de características, as palavras de cada grupo continuaram relacionadas a um mesmo assunto, porém, alguns termos ainda se repetiram. A Figura 39 ilustra os quatros *clusters* formados pelo algoritmo *k-means* ao ser alimentado com um corpus composto por 123 notícias referentes aos temas política, economia, educação e saúde.

FIGURA 39: Diagrama de dispersão dos *clusters* de notícias (k=4 e n=4)



Fonte: Elaborada pela autora

A legenda apresenta as 4 características de cada *cluster* (*labels*).

O *Cluster 1*, que teve como principais características os termos “**projeto**”, “**educação**”, “**militar**” e “**trabalho**”, concentrou a maior parte das notícias sobre educação. Além disso, buscou na amostra sobre economia três (3) notícias que comentavam sobre escolas como UFRJ, Universidade do Texas, Fundação Getúlio Vargas e Sesc, e uma (1) outra notícia sobre projetos. Esse grupo buscou também duas (2) matérias sobre ditadura militar e uma (1) sobre projetos na coleção de informes sobre política. Frases como “talvez você não saiba”, “você precisa saber” podem ter influenciado o algoritmo a agrupar essas notícias com as outras sobre educação. E em relação aos rótulos, essas notícias foram adequadas. Porém, apesar do bom desempenho do *k-means* neste caso, essa técnica agrupou erroneamente algumas notícias que não eram sobre educação, assunto de maior concentração desse *cluster*.

O *Cluster 2* obteve como características as palavras “**reforma”,** “**governo**”, “**estado**”, “**economia**” e concentrou a maior parte das notícias relacionadas com o assunto economia. Neste caso, os descritores foram altamente relevantes e pertinentes com o tema do grupo. A notícia de título “Para presidente da Suzano, não passar a Previdência é o caos”, que foi recuperada pelo coletor como sendo da categoria saúde, também foi alocada no segundo grupo. Conforme ilustrado na Figura 40, a notícia realmente relata sobre economia e não sobre saúde.

FIGURA 40: Recorte de notícia

A proposta do governo prevê uma economia de R$ 1,1 trilhão em dez anos. Qual patamar seria o mínimo?

Se sair 10% a 20% abaixo da proposta, já é um gol de placa. A reforma de Temer ia dar uma economia de
a de R$ 400 bilhões. Se a proposta for desidratada, não vai gerar credibilidade, não vai adiantar nada
não vem o investimento.

Se tivermos uma economia de cerca de R$ 500 bilhões, vai ser suficiente?

Fonte: Elaborara pela autora

Embora o algoritmo tenha formado esse grupo com assuntos similares, ele teve como ponto negativo o fato de seis (6) notícias da coleção sobre economia não ter sido alocadas nesse *cluster*. Essas matérias não foram categorizadas em nenhum grupo.

O *Cluster 3*, que teve como descritores os termos “**Política”**, “**presidente”**, “**acordo” e** “**Brasil”**, concentrou a maior parte das notícias relacionadas à política, mas também ‘buscou’ outras cinco (5) matérias da coleção sobre saúde, sendo que três (3) realmente têm relação com o tema política e duas são realmente sobre saúde.

Por fim, o *Cluster 4*, cujas características extraídas foram “**Bolsonaro**”, “**presidente**”, “**ministro**” e “**governo**”, agrupou notícias de todas as categorias: política (9), saúde (3), educação (2) e economia (1). A maior concentração foi de notícias da categoria política e as

demais matérias desse grupo continham os descritores muito presentes no conteúdo dos textos.

Portanto, se tratando de similaridade, o algoritmo foi bastante eficiente. Todavia, algumas notícias não foram alocadas em nenhum *cluster*.

II) Teste do *k- means* (k=2 e n=4)

Apesar das notícias serem coletadas usando palavras-chave diferentes (política, economia, educação e saúde) na pesquisa, esses assuntos são altamente relacionados. Assim, optou-se por fazer um novo teste reduzindo o número de grupos para dois (k=2). O Quadro 18 apresenta os termos mais relevantes dos dois *clusters* formados.

QUADRO 18: Características extraídas pelo *k-means* – 2º experimento (k=2, n=4)

Clusters	**Características**
Cluster 1	bolsonaro, presidente, governo, ministro
Cluster 2	Brasil, militar, acordo, estado

Fonte: Elaborado pela autora

Observa-se que, apesar da coleção de notícias utilizada nos testes ser composta por quatro assuntos diferentes, os *clusters* não se organizaram de acordo com os temas do corpus, conforme esperado. Acredita-se que isso se deve ao fato de todos os temas da amostra, apesar de conterem notícias relacionadas à educação, saúde e economia, serem relacionados à política. Isso fez com que o resultado da segunda experimentação ficasse semelhante ao primeiro. Desta forma, serão necessários novos experimentos usando uma coleção de notícias formada por temas mais diversificados para uma melhor análise dos resultados.

III) Teste do *Affinity Propagation*

Ao alimentar o *Affinity Propagation* com um corpus contendo 107 notícias relacionadas aos temas política, economia, educação e saúde, esse algoritmo encontrou 22 *clusters*, um número bem acima do esperado. Assim, optou-se por analisar essa técnica apenas no terceiro experimento, que será realizado usando uma coleção mais diversificada e com uma quantidade menor de documentos, pois ao utilizar um corpus menor, será possível fazer uma análise de cada notícia e interpretar melhor a formação de cada grupo.

IV) *Cluster* hierárquico

Assim como realizado anteriormente, fizeram-se testes nesta experiência usando os algoritmos de agrupamento hierárquico combinando diferentes medidas de similaridade. O objetivo foi analisar o comportamento dessas técnicas ao serem alimentadas com um corpus formado por 107 notícias relacionadas aos temas política, economia, educação e saúde.

O Quadro 19 ilustra as combinações que foram realizadas e o resultado alcançado.

QUADRO 19: Método agrupamento x medida de similaridade

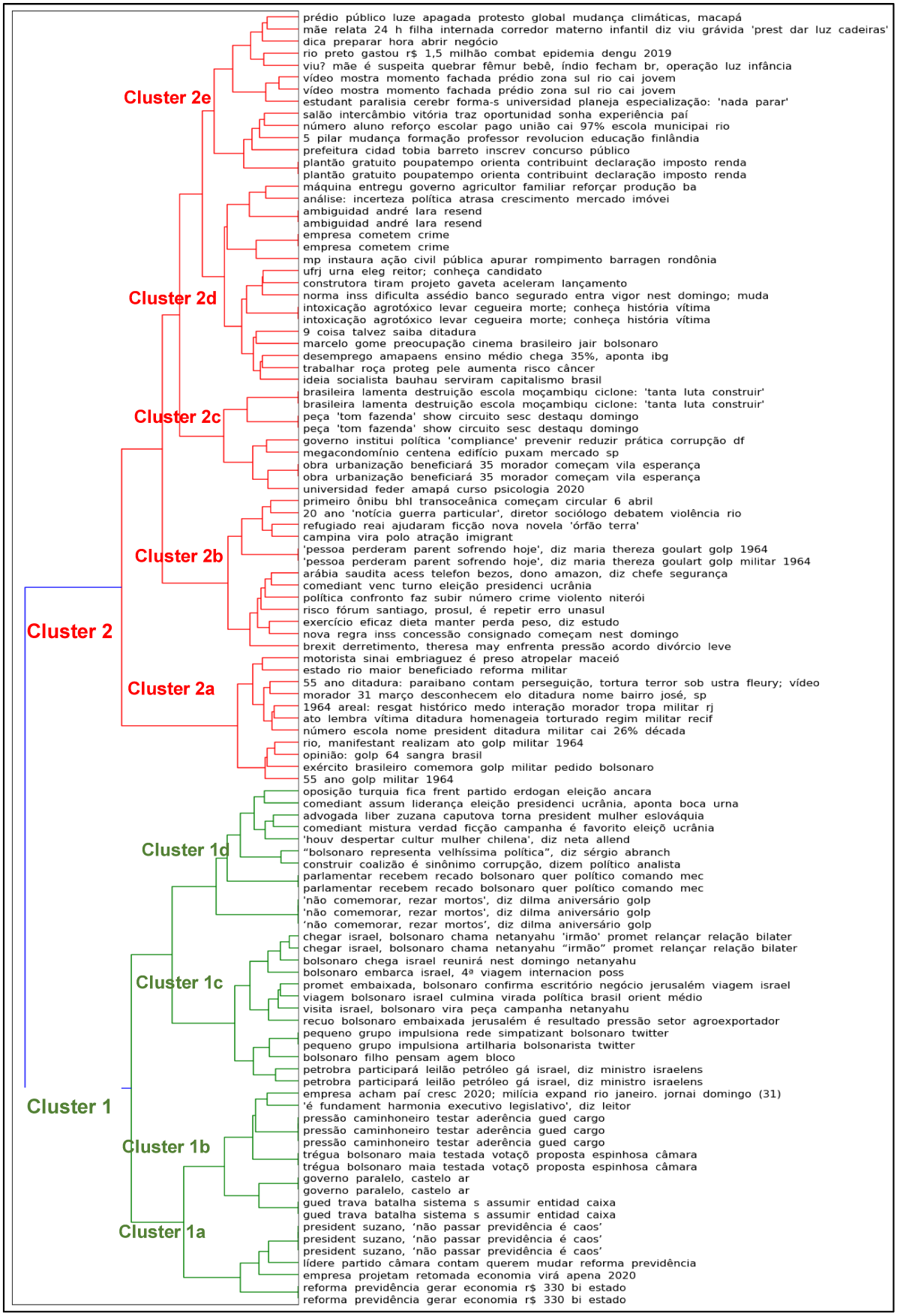
Experimento	Método de agrupamento	Medida da Similaridade	Nº de grupos formados
1º Teste	Single linkage	Distância Euclidiana	2 *clusters* primários, 3 secundários
		Similaridade de Cosseno	Não agrupou
		Distância de Manhattan	2 *clusters* primários, 3 secundários
2º Teste	Average linkage	Distância Euclidiana	2 *clusters* primários, 3 secundários
		Similaridade de Cosseno	Não agrupou
		Distância de Manhattan	2 *clusters* primários, 3 secundários
3º Teste	Complete linkage	Distância Euclidiana	2 *clusters* primários, 3 secundários
		Similaridade de Cosseno	Não agrupou
		Distância de Manhattan	2 *clusters* primários, 3 secundários
4º Teste	Ward	Distância Euclidiana	2 *clusters* primários, 4 secundários
		Similaridade de Cosseno	Não agrupou
		Distância de Manhattan	2 *clusters* primários, 3 secundários

Fonte: Elaborado pela autora

Conforme ilustrado no Quadro 19, todos os algoritmos apresentaram resultados semelhantes ao utilizar as medidas Distância Euclidiana e Distância de Manhattan e não conseguiram formar *cluster*s usando a Similaridade de Cosseno. Portanto, essa medida não é indicada para o cálculo da similaridade de textos no agrupamento de notícias coletadas dos principais jornais *on-line* usando as técnicas de agrupamento hierárquico.

A Figura 41 ilustra o dendograma formado usando o método *Ward*.

FIGURA 41: *Cluster* hierárquico – 2º experimento



Fonte: Elaborada pela autora

Conforme ilustrado na Figura 41, primeiramente as notícias foram subdivididas em dois (2) grandes grupos. O *Cluster 1* concentrou notícias de todas as quatro (4) categorias, sendo quatorze (14) notícias da coleção sobre economia, dezoito (18) notícias da amostra sobre política, cinco (5) sobre educação e seis (6) sobre saúde, computando um total de 43 matérias alocadas nesse aglomerado. Apesar desse grupo ter concentrado notícias de todas as categorias, houve uma maior concentração de notícias relacionadas ao tema economia e política. Entretanto, as notícias que foram recuperadas pelo *Media Frame* como pertencentes às categorias educação e saúde também são sobre política, porque essas áreas têm forte relação.

Mais especificamente, o *Cluster 1a* agrupou sete (7) notícias sobre economia, sendo quatro (4) realmente oriunda da coleção que contém as matérias sobre esse assunto. As demais vieram das amostras sobre saúde e política, mas também tem relação com esse tema. O *Cluster 1b* também concentrou dez (10) notícias sobre economia, sendo que oito (8) vieram da coleção recuperada pelo coletor usando a palavra-chave “economia” e as outras duas usando a palavra-chave “política”. Já o *Cluster 1c* foi formado por 13 notícias sobre política, sendo que dez (10) delas vieram da amostra sobre esse tema e as demais vieram da coleção sobre saúde. E para finalizar esse bloco, o *Cluster 1d* agrupou 12 notícias também relacionadas à política, sendo que sete (7) vieram da coleção sobre política e as demais do corpus sobre educação.

O *Cluster 2* foi subdivido em 5 grupos menores. O *Cluster 2a* formou-se por notícias de todos os corpora (educação, política, economia e saúde), porém conseguiu formar uma subcategoria com assuntos bem semelhantes cujos principais termos dos textos desse grupo são “**ditadura militar**” e “**golpe militar**”. Diante desse fato, pode-se afirmar que o *Cluster 2a* é constituído por uma subcategoria de notícias relacionadas à ditatura militar.

Os *Clusters 2b* e *2c* subdividiram-se em dois novos subgrupos cada, mas não conseguiram formar grupos bem definidos, pois apesar da maioria tratar de assuntos relacionados à política, eles também concentraram notícias sobre saúde e economia, cujos conteúdos das matérias são pouco relacionados. Quanto ao *Cluster 2d*, dos quatorze (14) informes pertencentes a esse grupo, nove (9) foram sobre economia. Contudo, notícias das outras categorias também foram alocadas nesse aglomerado. Por fim, o *Cluster 2e* foi formado por quatorze (14) notícias, sendo onze (11) sobre educação e três (3) sobre saúde.

Por conseguinte, ao analisar os *clusters* formados pelo método *Ward*, usando a Distância Euclidiana, observa-se que praticamente todas as notícias são relacionadas à política e cujos termos mais importantes são “Bolsonaro”, “presidente”, “governo”, “ministro”, “economia”, “ensino”, “federal”, “educação”, ou seja, as palavras mais relevantes estão relacionadas à política. Isso é porque, apesar das notícias terem sido coletadas como

pertencentes a categorias diferentes, os quatros temas têm forte ligação com o assunto em questão.

5.2.2 Validação dos resultados do 2º experimento

Na segunda experiência, realizou-se a avaliação do melhor número de *cluster*, usando também o algoritmo *k-means,* através do coeficiente da silhueta. Fez-se o teste com k variando de 2 a 8. O resultado é exibido na Tabela 6.

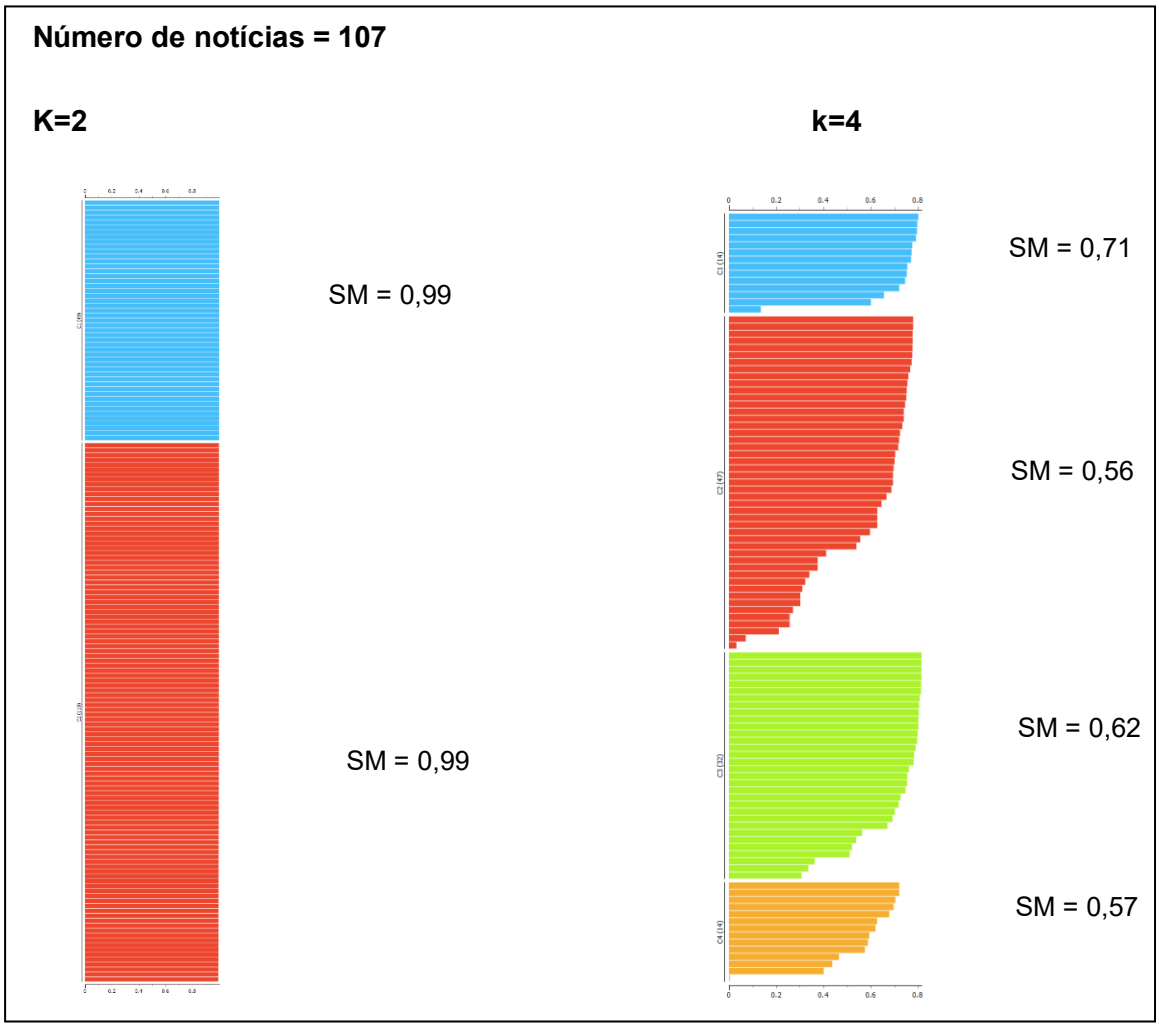
TABELA 6: Avaliação do *k-means* usando o coeficiente da silhueta – 2º experimento

	Método de avaliação	
Nº de *clusters*	Silhouette	
	Inicialização com *k-Means++*	Inicialização randômica
2	0,996	0,996
3	0,690	0,677
4	0,712	0,712
5	0,721	0,714
6	0,633	0,726
7	0,641	0,618
8	0,625	0,634

Fonte: Elaborada pela autora

Após essa simulação, a coleção foi dividida em 2 e em 4 *clusters*. A qualidade dos agrupamentos é mostrada pelos gráficos da silhueta na Figura 42.

FIGURA 42: Gráficos da silhueta – 2º experimento



Fonte: Elaborada pela autora

Fazendo uma análise das notícias do segundo experimento, observa-se que todo o corpus tem relação com o tema política, mesmo com temas referentes à economia, educação e saúde, visto que esses assuntos são muito relacionados. Ao dividir a coleção em dois *clusters*, de acordo com o índice da silhueta, o resultado foi um agrupamento forte. O grupo 1 (C1) foi formado por 29 notícias e o grupo 2 (C2) pelo restante, ou seja, 78 documentos. No segundo agrupamento ficaram concentradas as notícias sobre política, cujas palavras que apareceram com maior peso foram ‘política’, ‘presidente’, ‘Bolsonaro’, ‘reforma’, ‘governo’, ‘parlamento’.

Verifica-se na Figura 37 que, ao usar k=4, encontraram-se as seguintes médias dos coeficientes da silhueta para cada grupo:

Cluster 1: 14 notícias – valor médio de silhueta: 0,71

Cluster 2: 47 notícias – valor médio de silhueta: 0,56

Cluster 3: 32 notícias – valor médio de silhueta: 0,62

Cluster 4: 14 notícias – valor médio de silhueta: 0,57

Como o coeficiente foi aproximadamente 0.6, significa que uma estrutura razoável foi encontrada. Deste modo, optou-se por fazer mais um teste, usando um corpus mais diversificado, com uma quantidade menor de notícias, de forma que o conteúdo das matérias pudesse ser mais facilmente avaliado.

5.3 Experimento 3: Teste com um corpus de 50 notícias relacionadas aos temas economia, biologia, eletricidade e futebol

No terceiro experimento, optou-se por uma amostra mais diversificada e por uma coleção menor, contendo apenas cinquenta notícias, para facilitar a avaliação dos grupos formados. Para isso, coletaram-se, separadamente, 12 notícias usando na busca a palavra-chave biologia, 13 notícias relacionadas ao tema futebol, 12 sobre eletricidade e 13 sobre economia. Posteriormente os corpora foram misturados formando apenas um corpus.

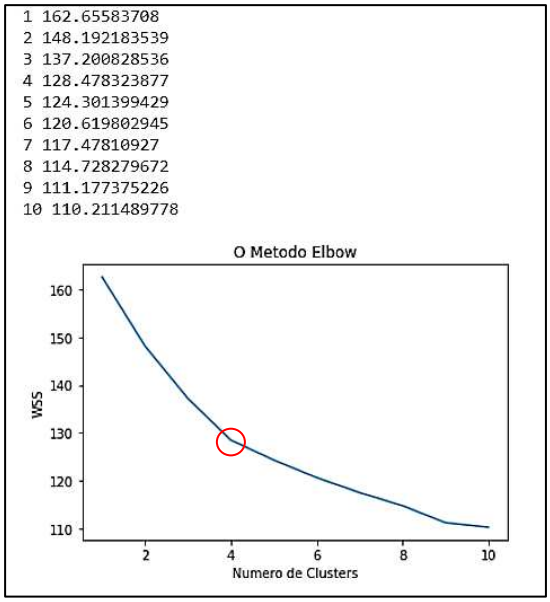
Assim como realizado no segundo teste, repetiram-se todos os passos relacionados ao pré-processamento já discutido no primeiro experimento e em seguida, realizou-se a etapa de *clustering*.

5.3.1 *Clustering* das notícias usando o algoritmo *k-means*

I) Cálculo do valor de K

Novamente, utilizou-se o método *Elbow* para encontrar o valor de **k**. A Figura 43 mostra o gráfico gerado.

FIGURA 43: Cálculo do valor de k pelo *Elbow* – 3º experimento



Fonte: Elaborada pela autora

Observa-se, mais uma vez, que não houve um ponto em que a inércia diminuiu subitamente de forma a ficar nítido o “cotovelo”, ou seja, o ponto que indica o número ideal de *clusters* a serem criados com base no conjunto de notícias. Mas, ainda assim, é possível notar uma inclinação em k=4, valor esperado, visto que a amostra foi composta por notícias referentes a quatro temas diferentes.

II) Teste do *k- means* (k=4 e n=4)

Neste teste, atribui-se o valor quatro, tanto para **k** quanto para o número de *feature*. O Quadro 20 mostra as características extraídas para cada *cluster*.

QUADRO 20: Características extraídas pelo *k-means* – 3º experimento (k=4, n=4)

Clusters	Características
Cluster 1	biologia, vivo, trabalho, atividade
Cluster 2	copa, futebol, seleção, time
Cluster 3	energia, carga elétrica, eletrifio, risco
Cluster 4	presidente, governo, economia, ministro

Fonte: Elaborado pela autora

As características são os termos principais de cada grupo. Assim, foram agrupadas as notícias que, de certa forma, apresentam particularidades em comum, ou seja, o algoritmo de *clustering* encontrou algum padrão ou similaridade entre algumas matérias que as classificaram como pertencentes a um mesmo grupo. Fazendo uma análise dos termos mais frequentes e relevantes de cada um dos aglomerados, conclui-se que esses *clusters* estão mais coesos e relacionados às categorias especificas do que nas experiências anteriores. Observa-se no *Cluster 4* que as principais características extraídas para esse grupo foram os termos “presidente”, “governo”, “economia” e “ministro”. Isso quer dizer que as notícias pertencentes a esse aglomerado apresentam essas palavras com maior frequência. Ao mudar o valor de n para 8, os termos “Bolsonaro”, “Guedes”, “reforma” e “previdência” foram acrescentados à lista de descritores. Portanto, nesse caso, foi interessante aumentar o número de características de forma que termos mais específicos sobre o assunto fossem extraídos.

Como não se tem nenhuma informação sobre as categorias que cada notícia pertence, os descritores do grupo 4 foram altamente relevantes, pois a partir deles, é possível identificar que as notícias são pertinentes ao tema economia ou à política, pois esses dois assuntos são fortemente relacionados.

No *Cluster 1*, usando n=4, foram extraídas as seguintes características: “biologia”, “vivo”, “trabalho”, “atividade”. Não é possível identificar o assunto central do aglomerado de notícias com apenas essas palavras. Assim, fez-se o teste usando n=8 e os termos “humano”,

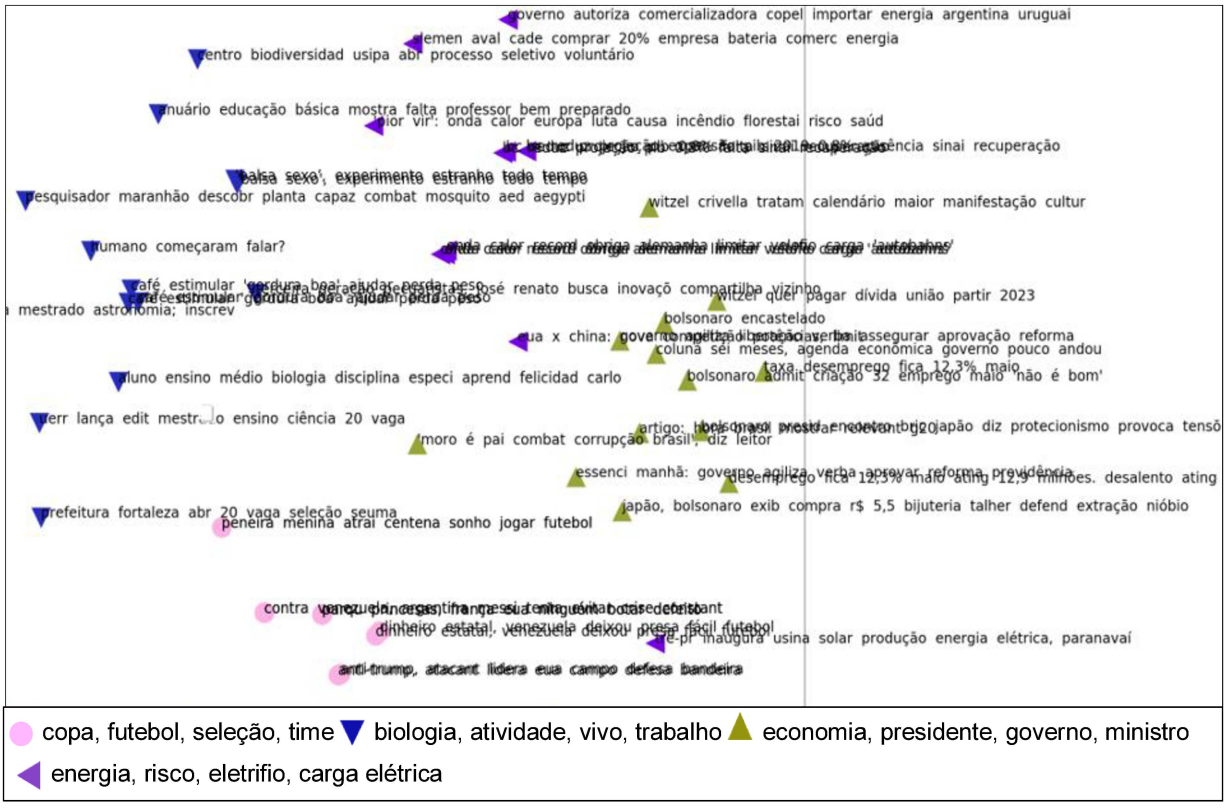
"gordura", "corpo", "pesquisa" foram acrescentados à lista de descritores. Apesar desses vocábulos terem ligação com o tema biologia, fica difícil para o usuário, que não tem nenhum conhecimento sobre o conteúdo dos documentos, identificar qual é o assunto central das matérias tendo apenas as *features* extraídas como referências. A vantagem é que o processo de *clustering* divide o corpus em grupos, o que facilita para o usuário fazer uma leitura rápida e identificar o assunto, já que o número de informes em cada aglomerado é bem menor, o que ajuda o leitor a encontrar apenas as informações que são relevantes para ele.

Assim, quando a coleção é composta por assuntos mais gerais, as chances de identificar o tema central das notícias, tendo como base apenas as características extraídas, são menores, pois textos mais específicos tendem a ter melhores descritores.

Por conseguinte, em relação à quantidade ideal de características, testes são necessários para descobrir o número que melhor representa os grupos, pois um número muito pequeno pode não ser suficiente para representar o aglomerado e se a quantidade for muito grande, pode acontecer das características não retratarem realmente cada *cluster*.

A Figura 44 apresenta os aglomerados formados usando o algoritmo *k-means* com k=4.

FIGURA 44: Diagrama de dispersão – 3º experimento (k=4 e n=4)



Fonte: Elaborada pela autora

Ao analisar cada grupo, observa-se que o *Cluster 1*, que agrupou as notícias pertinentes ao tema biologia, conseguiu aglomerar um total de 13 textos. Como, inicialmente, foram coletadas somente 12 notícias relacionadas a esse tema, realizou-se uma análise das matérias para verificar se realmente foi adequada a alocação realizada pelo algoritmo de *clustering.* Enfim, das 13 notícias classificadas no grupo 1, 11 realmente pertenciam ao grupo biologia. As outras duas matérias, que também foram agrupadas no primeiro *cluster*, apesar de coletadas como pertencentes ao tema eletricidade, elas não têm relação com esse assunto e também não pertencem aos temas futebol e economia. Como essas duas notícias tratavam de assuntos relacionados à antropologia, estudo do corpo humano, relações sexuais, estudo científico com macacos etc., o algoritmo acertou ao classificar essas notícias como pertencentes ao grupo biologia, visto que, entre os 4 temas, ele encontrou o *cluster* mais pertinente. Mas por que o coletor de notícias recuperou essas duas notícias ao pesquisar pelo tema eletricidade? As matérias tratavam do mesmo assunto, porém recuperadas de jornais diferentes. No decorrer do texto, apareceram várias vezes os termos “laboratório de carga”, local usado para estudar o comportamento humano. Isso justifica o fato delas terem sido recuperadas como pertencentes ao tema eletricidade.

Em relação ao grupo eletricidade, composto por 12 documentos, o *k-means* agrupou 10 informes como pertencentes a esse assunto. Como duas notícias, que foram coletadas no grupo de eletricidade, foram aglomeradas no grupo biologia, parece que o algoritmo acertou 100% as demais. Porém, das 10 notícias pertencentes ao grupo eletricidade, uma foi recuperada da amostra de economia. Essa notícia realmente descrevia sobre economia, o algoritmo errou nesse caso. Porém, como ele não leva em consideração o contexto, algumas palavras que apareceram nas frases, como por exemplo, ‘a China é uma **potência**’, ‘**tensões** comerciais’, ‘empresas de **semicondutores**’, fizeram com que o *k-means* agrupasse essa notícia no *cluster* sobre eletricidade.

Dando continuidade à análise, o algoritmo agrupou muito bem as notícias relacionadas à economia. Das 13 notícias, ele acertou 12. Uma das matérias desse tema ficou no aglomerado sobre eletricidade, conforme comentado anteriormente e uma outra notícia, pertencente a categoria futebol, foi alocada no grupo sobre economia. Porém, o algoritmo não errou nesse caso, pois o título da notícia era “Moro é o pai do combate à corrupção”. Com base no título, esse documento não poderia ter sido recuperado pelo coletor cuja palavra-chave usada na pesquisa foi futebol. Deste modo, para averiguar, foi necessária uma leitura completa da matéria. Assim, observou-se que a notícia realmente falava sobre Moro, porém, o autor fez uma comparação das ações do juiz com o futebol, que é um esporte movido por sentimentos e paixões. Nessa narração, a palavra futebol apareceu várias vezes, isso fez com que o coletor recuperasse essa notícia juntamente com as demais relacionadas ao tema

futebol. Assim, não se pode afirmar que o algoritmo de *clustering* errou nesse caso, pois a notícia sobre Moro também continha parágrafos sobre a Reforma da Previdência e outros termos relacionados à economia.

Quanto às notícias cujos assuntos são referentes à futebol, o *k-means* agrupou 6 matérias como pertencentes a essa categoria e uma outra, também relacionada a esse tema, foi para o grupo sobre economia. No total, 8 notícias não foram agrupadas, sendo 6 pertencentes a amostra de notícias sobre futebol. A Tabela 7 apresenta os acertos e erros do algoritmo *k-means*.

TABELA 7: Acerto x erro do *k-means* – 3º experimento

	Total de notícias relacionadas ao assunto	**Notícias agrupadas**	**Acerto**	**Erro**	**Agrupou de/em outras categorias**
Cluster 1 Biologia	12	13	11/13	2/13	2 de eletricidade
Cluster 2 Futebol	13	6	6/6	0	1 em economia
Cluster 3 Eletricidade	12	10	9/10	1/10	1 de economia, , 2 em biologia
Cluster 4 Economia	13	13	12/13	1/13	1 de futebol, 1 em eletricidade
Total	**50**	**42**	**38**	**4**	

Fonte: Elaborada pela autora

Ao analisar a quantidade de notícias agrupadas, o *k-means* conseguiu uma taxa acima de 90% de acerto e agrupou 84% dos textos. Considerando as particularidades das notícias, esse algoritmo obteve praticamente 100% de exatidão, ponderando apenas as matérias que foram congregadas. Por fim, ao fazer uma análise com base na leitura dos informes, identificou-se que nem todas as notícias foram agrupadas corretamente, porém, em alguns casos, o erro foi do coletor e não do algoritmo de *clustering*.

Contudo, a análise dos aglomerados auxilia na identificação do conteúdo das notícias e das suas relações. Isso é relevante quando se trabalha com uma grande quantidade de textos porque permite ao leitor identificar os grupos que contêm os documentos que mais o interessa. Além disso, por se tratar de uma técnica não supervisionada, pode-se considerar que o algoritmo teve um excelente desempenho.

III) Teste usando o algoritmo *Affinity Propagation*

O algoritmo *Affinity Propagation* tenta construir os agrupamentos com base nas propriedades dos dados sem qualquer pressuposto sobre o número de *clusters*. Assim, ao ser alimentado pelo corpus usado neste experimento, essa técnica encontrou seis grupos, ou

seja, k=6. Um valor diferente do que o esperado, visto que a coleção de notícias é composta por quatro assuntos distintos, mas ao mesmo tempo não muito discrepante do valor encontrado pelo método *Elbow*. Posteriormente, será realizada uma avaliação usando o Coeficiente da Silhueta para uma melhor análise do número de grupos. O Quadro 21 mostra as características extraídas pelo *Affinity Propagation*.

QUADRO 21: Características extraídas pelo *Affinity Propagation* (k=6, n = 4)

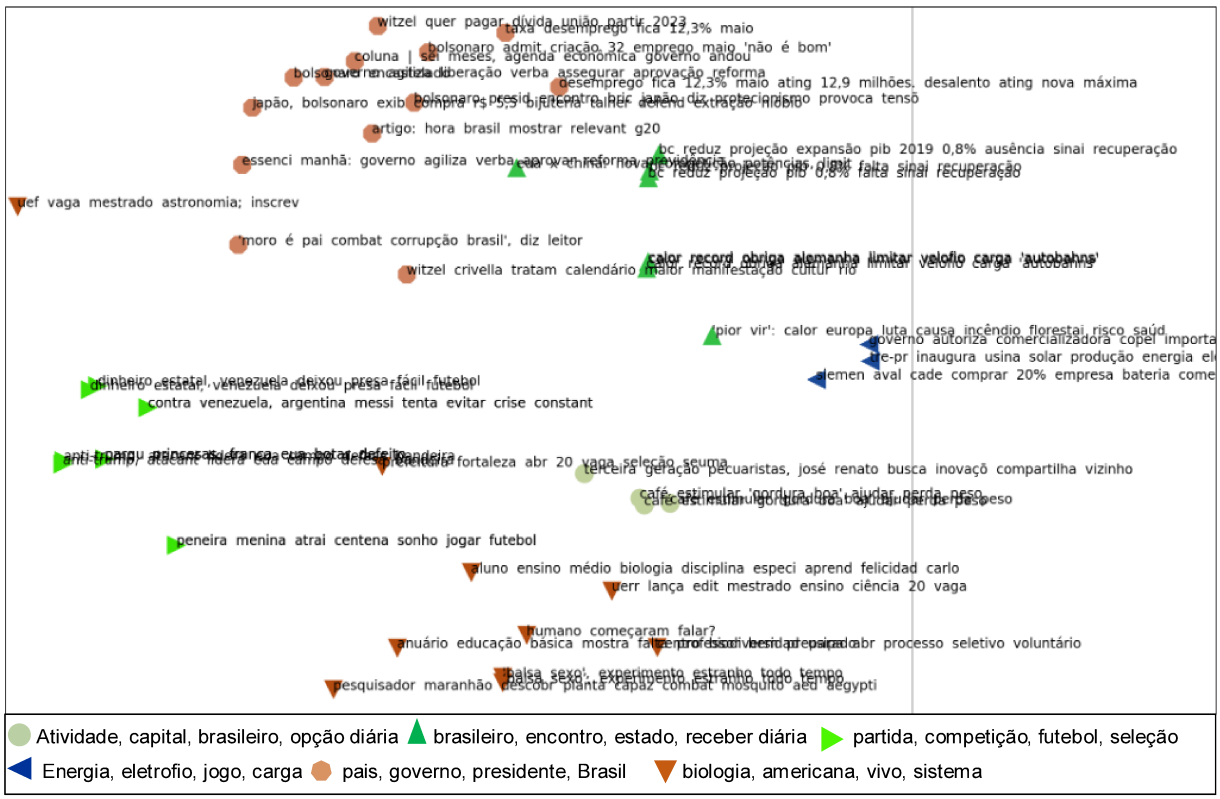
	Características
Cluster 1	atividade, capital, brasileiro, opção diária
Cluster 2	brasileiro, encontro, estado, receber diária
Cluster 3	partida, jogo, futebol, seleção
Cluster 4	energia, eletrifio, empresa, carga
Cluster 5	pais, governo, presidente, Brasil
Cluster 6	biologia, americana, vivo, sistema

Fonte: Elaborado pela autora

Observando apenas as características, sem analisar as notícias, o *Affinitty Propagation* misturou alguns termos de categorias diferentes e não extraiu descritores altamente relacionados, diferentemente do que aconteceu com o *k-means* que apresentou termos mais fortemente correlacionados para cada grupo.

Um ponto importante que se percebe ao analisar as características extraídas pelo *Affinity Propagation* é que nem todas as palavras-chave descrevem o assunto real do referido *cluster.* Assim, seria interessante usar outras técnicas de análise de textos, como por exemplo, nuvens de palavras ou modelos de tópicos, para extrair informações de cada grupo para assim, melhor representá-lo. A Figura 45 ilustra os aglomerados formados por esse algoritmo.

FIGURA 45: Diagrama de Dispersão - *Affinity Propagation* - 3º experimento



Fonte: Elaborada pela autora

O Quadro 22 apresenta um resumo contendo as principais características extraídas para cada *cluster*, a quantidade de notícias que foi alocada em cada grupo, além de mostrar, também, a quantidade de notícias por assunto que foi agrupada em cada aglomerado.

QUADRO 22: Notícias agrupadas por grupo

	Características extraídas	Quantidade de notícias no grupo	Notícias agrupadas
Cluster 1 ●	atividade, capital, brasileiro, opção diária	4	4 biologia
Cluster 2 ▲	Brasileiro, encontro, estado, receber diária	8	7 notícias relacionadas à eletricidade 1 relacionada à economia
Cluster 3 ▶	partida, competição, futebol, seleção	12	12 notícias relacionadas à futebol
Cluster 4 ◀	Energia, eletrifio, jogo, carga	3	3 notícias relacionadas à eletricidade
Cluster 5 ●	pais, governo, presidente, Brasil	13	12 notícias relacionadas à economia 1 notícia relacionada à futebol
Cluster 6 ▼	biologia, americana, vivo, sistema	10	8 notícias relacionadas à biologia 2 notícias relacionadas à eletricidade
Total		50	

Fonte: Elaborada pela autora

Ao analisar os valores dispostos no Quadro 22, observa-se que o *Affinity Propagation* teve como vantagens o fato de ter agrupado todas as notícias e ter formado grupos com assuntos semelhantes. Porém, não conseguiu acertar o número de grupos, visto que o corpus era formado por quatro temas e o algoritmo encontrou seis (6) *clusters*, alocando as notícias sobre eletricidade em 3 grupos. Além disso, o algoritmo não apresentou um bom desempenho em relação às características extraídas, visto que muitos dos termos não têm valor significativo em relação às notícias do grupo relacionado e, em contrapartida, outras palavras de peso pertencentes a uma determinada categoria foram alocadas em outros *clusters* como, por exemplo, a palavra **jogo** que foi alocada no grupo de notícias relacionas à eletricidade e não à futebol. Entretanto, acredita-se que se realizar uma análise no corpus após a coleta e retirar as notícias que foram recuperadas erroneamente, essa técnica de agrupamento teria uma melhor taxa de acerto.

IV) *Cluster* Hierárquico

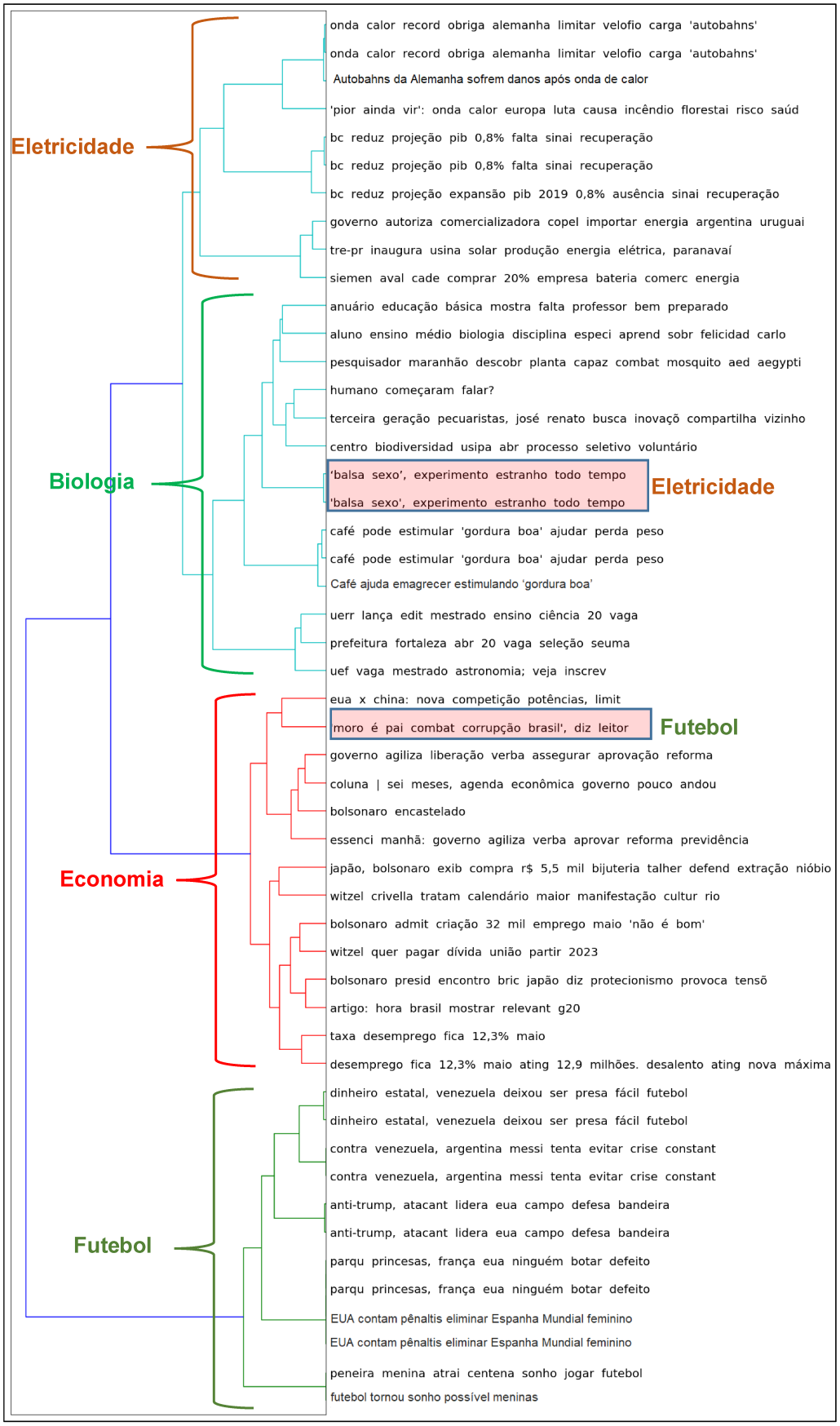
Conforme realizado no primeiro e no segundo experimento, fizeram-se alguns testes nesta experiência usando os algoritmos de agrupamento hierárquico combinando diferentes medidas de similaridade. O intuito foi analisar o comportamento dessas técnicas ao serem alimentadas com um corpus formado por 50 notícias relacionadas aos temas biologia, eletricidade, futebol e economia. O Quadro 23 mostra as combinações que foram realizadas e o resultado alcançado usando essa técnica, conforme ilustrado na Figura 46.

QUADRO 23: Método agrupamento x medida de similaridade

Experimento	Método	Medida da Similaridade	Nº de grupos formados
1º Teste	Single linkage	Distância Euclidiana	2 *clusters* primários, 4 secundários
		Similaridade de Cosseno	Não agrupou
		Distância de Manhattan	Não agrupou
2º Teste	Average linkage	Distância Euclidiana	2 *clusters* primários, 3 secundários
		Similaridade de Cosseno	Não agrupou
		Distância de Manhattan	2 *clusters* primários, 3 secundários
3º Teste	Complete linkage	Distância Euclidiana	2 *clusters* primários, 4 secundários
		Similaridade de Cosseno	Não agrupou
		Distância de Manhattan	2 *clusters* primários, 3 secundários
4º Teste	Ward	Distância Euclidiana	2 *clusters* primários, 3 secundários
		Similaridade de Cosseno	Não agrupou
		Distância de Manhattan	2 *clusters* primários, 3 secundários

Fonte: Elaborado pela autora

FIGURA 46: Agrupamento hierárquico – 3º experimento



Fonte: Elaborado pela autora

O método *Ward* e a Distância Euclidiana formaram a combinação que obteve melhor resultado. Conforme ilustrado na Figura 46, nota-se que primeiramente o algoritmo subdividiu as notícias em dois grupos. Em um deles ficaram unidas as matérias relacionadas à futebol e no outro, de tamanho maior, concentraram-se as demais notícias. Esse modelo considera primeiramente a coleção de textos como sendo um único grupo que posteriormente é recursivamente dividido para produzir um bom agrupamento no final.

Assim, na segunda divisão, o grupo maior foi divido em três (3) *clusters*: economia, biologia e eletricidade. Embora essa técnica tenha apresentado dificuldades nos experimentos anteriores, ela conseguiu um excelente desempenho com um corpus menor e mais diversificado. Praticamente todas as notícias foram agrupadas corretamente. Apenas uma matéria foi categorizada erroneamente, pois foi recuperada pelo *Media Frame* como pertencente ao grupo eletricidade, mas foi alocada no grupo sobre Biologia. Essa notícia apareceu duas vezes, provavelmente foi coletada de dois jornais diferentes.

De fato, a notícia de título "*A 'balsa do sexo', um dos experimentos mais estranhos de todos os tempos*" foi alocada no grupo sobre biologia. Provavelmente, termos como "*testes com animais*", "*estudo do corpo humano*", "*estudos científicos com macacos*, "*sexualidade*" e "*fêmeas ovulando*" fizeram com que o algoritmo agrupasse essa notícia no *cluster* sobre biologia. Contudo, os termos "*fios*", "*laboratório de carga*", "*eletrifios*" e "*tensões*", também presentes nessa matéria, fizeram com que o coletor a recuperasse quando realizou a busca usando a palavra-chave "*eletricidade*". No entanto, ao fazer uma leitura desse informe, observou-se que a notícia não deveria pertencer à nenhuma dessas duas categorias.

Além desse caso, a notícia "*Moro é o pai do combate à corrupção no Brasil*" foi recuperada pelo coletor como pertencente à categoria futebol e foi agrupada no *cluster* de economia. Isso também aconteceu com o *k-means*, conforme justificado anteriormente nos testes com esse algoritmo.

Em virtude dos fatos mencionados, e para evitar que imprevistos como esses aconteçam, é importante que tanto os Sistemas de Recuperação da Informação quanto os algoritmos de agrupamento levem em consideração a semântica contida nos textos, de modo que sejam analisados não somente o significado de cada palavra, mas também o contexto em que ela está inserida.

6.3 Validação dos resultados do 3º experimento

Na terceira experimentação, realizou-se a avaliação do melhor número de *cluster*, usando também o algoritmo *k-means,* através do coeficiente da silhueta. Fez-se o teste com k variando de 2 a 8. O resultado é exibido na Tabela 8.

TABELA 8: Avaliação do *k-means* usando o coeficiente da silhueta

Nº de *clusters*	Método de avaliação	
	Silhouette	
	Inicialização com *k-Means++*	Inicialização randômica
2	0.580	0.580
3	0.625	0.625
4	0.668	0.668
5	0.693	0.693
6	0.701	0.701
7	0.675	0.675
8	0.666	0.682

Fonte: Elaborada pela autora

Como os índices variaram entre 0,5 e 0,7, significa que uma estrutura razoável foi encontrada. O valor maior do coeficiente da silhueta foi para k=6, indicando que esse corpus seria melhor organizado se as notícias fossem subdivididas em seis grupos.

O experimento 2 conseguiu um índice melhor do que o experimento 1, de acordo com o valor da silhueta. Assim, conclui-se que a diversidade do conteúdo também influencia nos resultados. A Tabela 9 apresenta a comparação dos valores da silhueta obtidos nos experimentos 1 e 2.

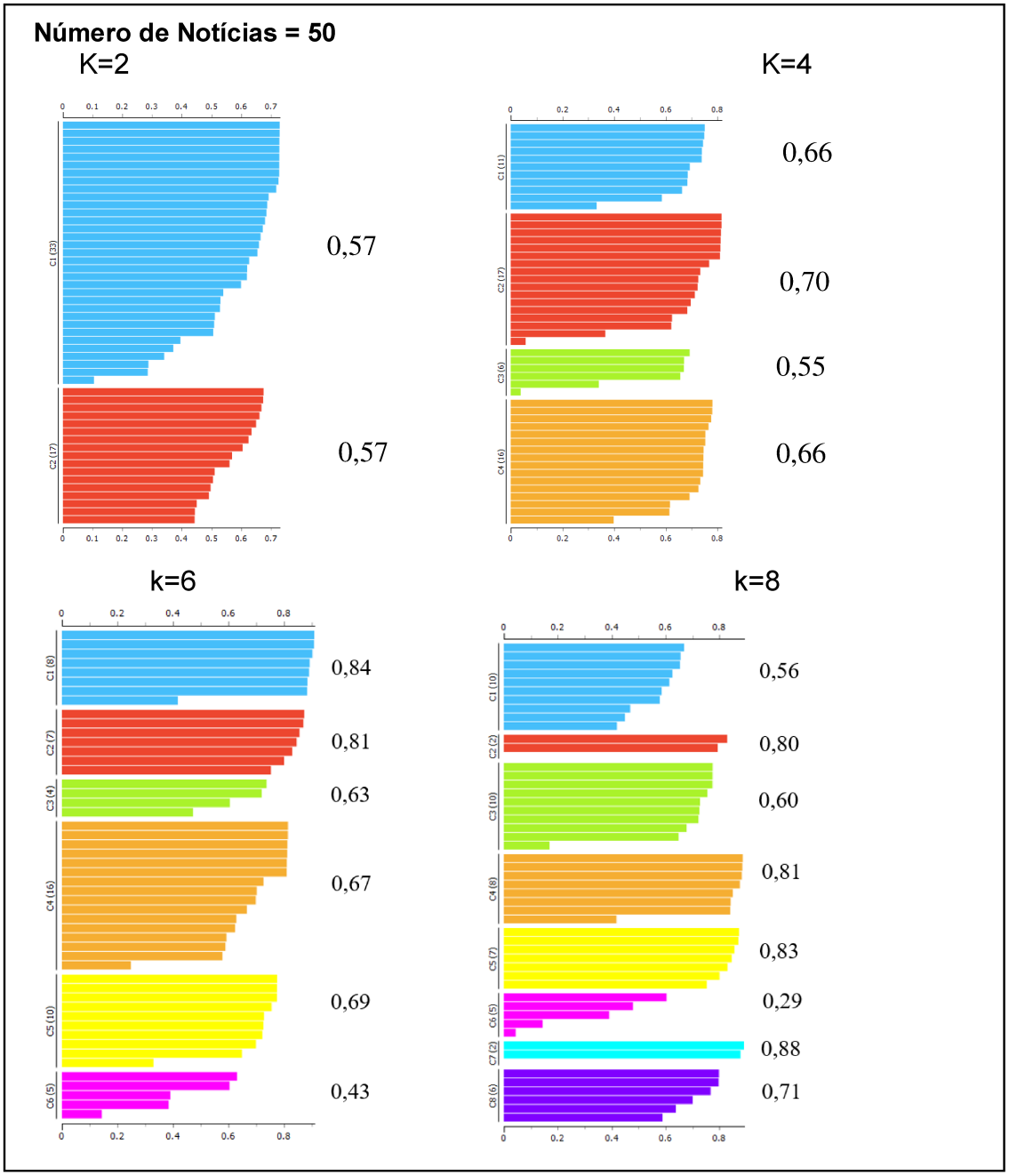
TABELA 9: Comparação dos coeficientes da silhueta obtidos nos experimentos 1 e 2

Nº de *clusters*	Método de avaliação	
	Silhouette	
	1º experimento	2º experimento
2	0,694	0.996
3	0,621	0.690
4	0,629	0.712
5	0,594	0.721
6	0,590	0.633
7	0,586	0.641
8	0,606	0.625

Fonte: Elaborada pela autora

Após a simulação realizada com o *k-means*, variando o número de *clusters*, elaboraram-se os gráficos das silhuetas para melhor visualização do agrupamento, conforme apresentados na Figura 47.

FIGURA 47: Gráficos da silhueta para diferentes valores de k – 3º experimento



Fonte: Elaborada pela autora

Após a obtenção dos aglomerados, realizou-se um estudo para saber qual é o perfil de cada *cluster*. Assim, analisou-se o agrupamento formado pelo *k-means*, usando k=4, para verificar se as notícias foram agrupadas corretamente, de acordo com as categorias. Apesar

de cada grupo ter predominância de notícias de um determinado assunto, todos os *clusters* apresentaram algumas divergências.

Por conseguinte, para melhorar a qualidade do agrupamento é necessária, também, uma análise das notícias recuperadas, visto que muitos informes relatam mais de um assunto em seu conteúdo, dificultando, assim, a tarefa de *clustering.* Mas, ao mesmo tempo, a principal vantagem desta técnica é a possibilidade de descobrir agrupamento não óbvio.

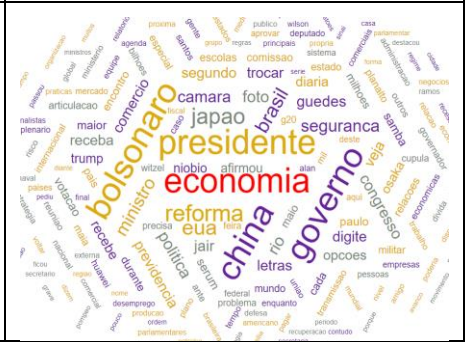
5.4 Discussão e interpretação do melhor resultado

Conforme apresentado no resultado dos experimentos, o *k-means* foi o algoritmo que apresentou melhor desempenho ao ser alimentado com um corpus de notícia no idioma português. No 3º experimento, utilizou-se um corpus com uma quantidade menor de documentos de forma que a análise manual do conteúdo fosse possível, pois é importante interpretar os resultados e discutir se os métodos automatizados são adequados e suficientes para análise de textos.

Mimno (2012) discute como um pesquisador pode examinar as palavras e seu relacionamentos dentro de um texto. Especificamente, ele faz várias observações sobre coerência, incluindo a identificação de palavras intrusas, ou seja, termos que estão fora do contexto em relação ao assunto em discussão. Além disso, documentos intrusos também podem ser identificados, ou seja, textos que foram alocados em um grupo, mas estão fora do contexto dos demais elementos do *cluster*.

Desse modo, essa abordagem pode auxiliar os seres humanos a interpretarem modelos de *clustering* com base em duas tarefas de avaliação: avaliar explicitamente a qualidade dos *clusters* inferidos pelo modelo e quão bem o modelo atribui rótulos aos grupos. O Quadro 24 apresenta as notícias recuperadas pelo *Media Frame* como pertencentes à amostra sobre “economia” e seus principais termos.

QUADRO 24: Notícias recuperadas pelo *Media Frame* e principais características

Título das notícias recuperadas pelo *Media Frame* (economia)	Palavras-chave	peso	Nuvem de palavras
0 Desemprego fica em 12,3% em maio e atinge 12,9... 1 Artigo: Hora de o Brasil se mostrar relevante ... 2 Witzel quer pagar dívidas à União só a partir ... 3 Witzel e Crivella tratam como calendário a mai... 4 Bolsonaro preside encontro dos Brics no Japão ... 5 Coluna | Em seis meses, agenda econômica do go... 6 EUA X China: uma nova era de competição entre ... 7 Bolsonaro encastelado 8 O essencial da manhã: Governo agiliza verbas p... 9 Taxa de desemprego fica em 12,3% em maio 10 No Japão, Bolsonaro exibe compra de R$ 5,5 mil... 11 Bolsonaro admite que criação de 32 mil emprego... 12 Governo agiliza liberação de verbas para asseg...	Economia Presidente Governo Bolsonaro China Reforma Japão EUA Brasil Ministro	43 41 39 37 36 23 22 21 19 17	

Fonte: Elaborado pela autora

Dada uma coleção de notícias, espera-se que palavras-chave como “presidente”, “governo” e “ministro” estejam associados ao assunto “política”. Por outro lado, tema específico como “futebol”, provavelmente, terá termos como “bola” e “jogador”, enquanto termos gerais como “porque” e “então” teriam as mesmas chances de aparecerem em ambos os casos. Desse modo, é importante retirar as palavras sem significância na fase de pré-processamento para que os rótulos sejam representativos.

O Quadro 25 mostra as notícias que foram agrupadas automaticamente como pertencentes à categoria “economia” e seus respectivos rótulos. Como esse aglomerado conseguiu agrupar praticamente todas as notícias recuperadas pelo *Media Frame*, optou-se por uma análise mais profunda do conteúdo das matérias para verificar se os rótulos realmente descrevem o teor dos documentos.

QUADRO 25: Notícias do grupo economia e suas principais características

Notícias alocadas no grupo economia	Rótulos gerados
0 Desemprego fica em 12,3% em maio e atinge 12,9... 1 Artigo: Hora de o Brasil se mostrar relevante ... 2 Witzel quer pagar dívidas à União só a partir ... 3 Witzel e Crivella tratam como calendário a mai... 4 Bolsonaro preside encontro dos Brics no Japão ... 5 Coluna | Em seis meses, agenda econômica do go... 6 EUA X China: uma nova era de competição entre ... 7 Bolsonaro encastelado 8 O essencial da manhã: Governo agiliza verbas p... 9 Taxa de desemprego fica em 12,3% em maio 10 No Japão, Bolsonaro exibe compra de R$ 5,5 mil... 11 Bolsonaro admite que criação de 32 mil emprego... 12 Governo agiliza liberação de verbas para asseg... Moro é o pai do combate à corrupção	presidente, governo, economia, ministro, Bolsonaro, Guedes, reforma, previdência, estado, congresso

Fonte: Elaborado pela autora

Tendo como base a metodologia de Mimno (2012), nota-se que nenhum rótulo do *cluster* que agrupou as notícias sobre economia é intruso, pois todos os termos têm relação com o assunto do grupo. E são palavras relevantes que realmente qualificam o aglomerado formado. Observa-se que os rótulos (Quadro 25) são mais representativos do que as palavras-chave (Quadro 24) quando analisados isoladamente, pois os descritores são bem representativos, principalmente, quando apresentados em conjunto. No entanto, alguns dos vocábulos têm pouco significado quando vistos de forma isolada ou como parte de um resumo de dez palavras. Não há contexto suficiente para determinar a definição de alguns termos. Desse modo, é necessária uma revisão adicional para os rótulos dos *clusters* como, por exemplo, apresentar os títulos dos informes de cada grupo, tópicos ou resumo que melhor os descrevem.

Também não houve notícias intrusas no aglomerado que agrupou as matérias sobre economia. Esse *cluster* concentrou praticamente todas as notícias recuperadas pelo *Media Frame* ao usar a palavra-chave “economia” na busca, perdendo apenas uma para o *cluster*

eletricidade, pelo fato dessa notícia apresentar termos característicos desse assunto, conforme comentado anteriormente. Apesar da evasão de uma matéria para outra categoria, esse *cluster* herdou uma outra notícia de título "*Moro é o pai do combate à corrupção*" da amostra recuperada pelo coletor como pertencente ao tema futebol. Porém essa notícia não pode ser considerada intrusa, pois após análise de seu conteúdo, observou-se que é mais pertinente o informe ser categorizado como pertencente ao grupo sobre economia do que futebol. A Figura 48 mostra um fragmento dessa notícia.

FIGURA 48: Recorte de notícia

Trump focado em reduzir o déficit comercial com a China, com muitos economistas dizendo que a estratégia não faz muito sentido. Mas as tarifas adicionais e a incerteza generalizada em torno das relações econômicas levando algumas empresas a repensarem se devem manter suas operações na China. Também tendo efeito a estratégia de colocar empresas chinesas, notadamente a Huawei, a gigante do setor de tecnologia de comunicação, em uma lista de entidades que não terão mais acesso a componentes americanos.

Fonte: Elaborada pela autora

A partir da leitura dessa notícia, percebe-se a importância da clusterização na separação de uma coleção de notícias em grupos menores e com assuntos mais específicos, pois conforme o caso apresentado, o algoritmo de *clustering* foi mais assertivo do que o Sistema de Recuperação das informações. O fato é que o algoritmo de aprendizado não supervisionado não recebe informação sobre o conteúdo do documento, ele aprende com o que há nos dados. Desse modo, ele é capaz de automatizar tarefas redundantes, detectar e analisar padrões ocultos. Diferentemente do que acontece com os sistemas de classificação que são treinados através de aprendizado supervisionado. Isso significa que os humanos devem rotular e classificar os dados, o que pode ser uma tarefa trabalhosa e propensa a erros. Apesar disso, esses modelos também são muito eficientes na automatização de tarefas.

Ao fazer uma leitura sucinta nas matérias pertencentes ao *Cluster 4* (economia), formado pelo algoritmo *k-means*, é possível descobrir o que os principais jornais estão publicando sobre o assunto com pouco esforço e sem gastar muito tempo. O Quadro 26 apresenta os títulos das notícias do *Cluster 4* com as principais frases de cada matéria.

QUADRO 26: Título x Tópicos das notícias

Título das notícias do *Cluster 4* (economia)	Tópicos ou frases relevantes do *Cluster 4*
Desemprego fica em 12,3% em maio e atinge 12,9 milhões. Desalento atinge nova máxima	Taxa de desemprego; pessoas procuram emprego; Economia fraca; por que Brasil não cresce? mercado de trabalho mostra sinais precarização; sem sinais de crescimento; retomada da economia somente 2020.
Hora de o Brasil se mostrar relevante no G20	Organização Mundial do Comércio; instituições centrais; sistema internacional; livre comércio; processos integração regional; Brasil precisa resistir pressões americanas; risco soberania nacional; negociações; evitar danos imagem externa; discurso presidencial não incorpore "globalismo", "marxismo cultural"; rejeição à globalização.
Witzel quer pagar dívidas à União só a partir de 2023.	Ministro Paulo Guedes antecipar revisão Regime Recuperação Fiscal; suspensão pagamento serviço dívida; prevejo o caos; situação financeira precária; empurrar frente 13 bilhões.
Witzel e Crivella tratam como calendário a maior manifestação cultural do Rio.	Privatização carnaval carioca; carnaval atravessa momento crise; cenário grave ameaça desfiles; Evento promove desenvolvimento local.
Bolsonaro preside encontro dos Brics no Japão e diz que protecionismo provoca tensões	Enfrenta momentos difíceis; presidente criticou práticas econômicas protecionistas; redução de medidas distorcivas; comércio agrícola; abrir mercado gás natural; oferta de energia barata; modernização leis trabalhistas.
Em seis meses, agenda econômica do governo pouco andou.	Reforma Previdência empurrada; tensão entre presidente da Câmara e ministro da economia; Ministério da Economia não informou cronograma plano arrecadar bilhões com privatizações; reforma tributária ficou num limbo; empréstimos estados quebrados não andou.
Bolsonaro encastelado	Reestruturação político-administrativa do Planalto não estava funcionando; articulação política teria de ser reconcebida; movimento na contramão; articulação do governo com o Congresso precisa mudar; dificuldades aprovação reforma mais ampla.
O essencial da manhã: Governo agiliza verbas para aprovar reforma da Previdência	Promessa repassa milhões projetos escolhidos por deputado até votação texto no plenário; Defesa vê falha grave embarque cocaína avião da FAB; governo movimenta para acalmar tensão com parlamentares; presidente comissão especial analisa reforma; relatório será votado na próxima semana.
Taxa de desemprego fica em 12,3% em maio	Desemprego era de 12,4%; recuperação lenta do mercado de trabalho; falta de investimentos; economia não cresce de forma consistente e linear, retomada da economia em 2020.
No Japão, Bolsonaro exibe compra de R$ 5,5 mil em bijuteria e talheres para defender extração de nióbio	chefe Planalto defende maior investimento explora ção de nióbio no Brasil.
Bolsonaro admite que criação de 32 mil empregos em maio 'não é bom'	Presidente admitiu que a criação de 32 mil empre gos com carteira assinada em maio "não é bom, poderia ser melhor."; indicação que economia não vai bem; melhora depende aprovação reforma Previdência.

Governo agiliza liberação de verbas para assegurar aprovação da reforma	Casa Civil pediu integrantes das bancadas para indicar projetos que possam receber recursos; governo prometeu milhões a cada parlamentar; Votação antes do recesso; Guedes indica sistema de capitalização; relatório 90% concluído; governo precisa resolver problema do seu partido que pret ende suavizar as regras para profissionais da área de segurança; parecer manterá a proposta do gov erno com idade mínima de 55 anos.
Moro é o pai do combate à corrupção	Lula X Moro, STF decidir sobre a validade de conchavos entre julgador e acusação; necessidade instituições íntegras, fortes, imparciais; Supremo mantém Lula preso adia julg amento. Lula depende da apuração da veracidade do material do site The Intercept Brasil? Moro previne da possibilidade de confirmação da validade do material; Futebol é matemática; esporte movido por paixões e sentimentos deve ser mediado pela objetividade.

Fonte: Elaborado pela autora

Para a extração das frases que resumissem cada notícia, utilizou-se o software *Orange Canvas*, um software *open source* que possui um módulo para Modelagem de Tópicos que é capaz de explorar grandes quantidades de dados textuais e encontrar grupos de palavras similares além de descobrir tópicos importantes nos textos. Porém, usou-se essa ferramenta apenas como auxílio para encontrar as frases mais relevantes nas notícias. Modelagem de tópicos é um outro tema de pesquisa que se combinado com as técnicas de *clustering* poderá produzir resultados interessantes, mas é uma questão para estudos futuros.

Assim, leram-se as notícias na procura das frases que pudessem resumir o conteúdo dos informes. Os tópicos extraídos anteriormente serviram como guias para esse processo. Frases como "*Economia fraca*", "*Por que país não cresce?*", "*taxa alta de desemprego*", "*políticas regressivas*", "*sem condições pagar dívidas*" etc. levam o leitor a concluir, sem necessidades de ler a coleção inteira, que no período em que as notícias foram coletadas, a economia do país não estava indo muito bem.

Logo, seria interessante acrescentar a cada *cluster*, além dos rótulos e dos títulos dos textos, um conjunto de frases relevantes. Isso ajudaria ainda mais o leitor a filtrar apenas a informação desejada, pois conforme apresentado no Quadro 25, as frases resumem bem o conteúdo de cada matéria. Já os rótulos do agrupamento, que muitas vezes são úteis para a identificação do assunto, exigem maior familiaridade com o domínio da coleção. Assim, um leitor que não é especialista no domínio pode ter dificuldades em interpretar os descritores e não conseguir identificar os conceitos essenciais que representam o assunto em discussão.

Contudo, para conseguir melhores resultados, é interessante utilizar as principais técnicas existentes que envolvem o uso de abordagem não supervisionada juntamente com as técnicas manuais. Quando não se tem um conhecimento especializado sobre o tema em

discussão, os recursos não supervisionados visam tornar o processo mais rápido e podem, automaticamente, encontrar padrões e relações em um conjunto de dados. Já as práticas manuais são tradicionais, todavia exigem conhecimentos de especialistas, são mais propicias a erros, o processo é mais trabalhoso e demorado e necessita de mais recurso tanto pessoal quanto de tempo. No entanto, as duas técnicas usadas em conjunto podem extrair dos documentos informações úteis, confiáveis e que representam fielmente o conhecimento contido em uma base textual.

CONCLUSÃO

A descoberta e a análise de aglomerados textuais são processos importantes para a estruturação, organização e a recuperação de informações. Através dos *clusters,* é possível analisar os dados e conhecer melhor o seu conteúdo e suas relações. Assim, explorar e desenvolver uma solução que possa extrair informações de texto e organizá-las por semelhança é um grande desafio devido às dificuldades encontradas na análise de linguagens naturais e na quantidade de informações que precisa ser processada. Apesar disso, o processo de automatizar a descoberta de conhecimento em textos possui um grande potencial para auxiliar as organizações no processo de tomada de decisão, além de ter aplicabilidade em diversas área do saber.

A partir da revisão de literatura, observou-se que os trabalhos relacionados à língua portuguesa são escassos e a maioria dos esforços é direcionada para a língua inglesa. Essa lacuna encontrada nas pesquisas na área de agrupamento de documentos impulsionou o interesse em fazer novas experimentações que levassem em consideração as características do idioma brasileiro. Nesse contexto, este estudo buscou testar e adaptar uma metodologia de aprendizado não supervisionado com o intuito de agrupar, automaticamente, notícias publicadas no idioma do Brasil, postadas na grande mídia. Para isso, identificaram-se as técnicas que são usadas no processo de *clustering* de textos e as aplicaram nas coleções recuperadas pelo *Media Frame*.

Assim, pode-se afirmar que uma das contribuições deste trabalho foi a análise do desempenho de alguns algoritmos de agrupamento de textos aplicados às notícias recuperadas de jornais *on-line* e publicadas na língua brasileira. Optou-se por analisar os algoritmos de *clustering* (*K-Means*, *Affinity Propagation* e os algoritmos hierárquicos *Single Linkage*, *Average Linkage*, *Complete Linkage* e o *Ward's method)*. Para tal fim, testaram-se esses recursos em três diferentes corpora, o que permitiu discutir o sucesso dos métodos em cada caso, além de averiguar a possibilidade efetiva de clusterização dos informes e analisar as dificuldades encontradas.

Uma outra contribuição desta pesquisa foi explorar as particularidades da língua brasileira durante todo o processo de agrupamento das notícias. Para tanto, conforme apresentado nos experimentos, durante a fase de pré-processamento, fez-se um esforço para reduzir a dimensionalidade dos dados, um dos maiores desafios no processo de análise de textos. Para isso, através de expressões regulares, fez-se uma limpeza nas notícias retirando todos os caracteres especiais, números e pontuações. Além disso, removeram-se todas *stop words*, que são palavras que têm pouca ou nenhuma significância no texto.

Outra etapa que é importante na fase de pré-processamento é a tokenização. Essa técnica tem como ponto negativo não levar em consideração o contexto, pois os documentos

são quebrados em *tokens* e a sequência das palavras não é levada em consideração. Para minimizar esse problema, utilizou-se, nos experimentos desta pesquisa, a técnica n-gramas, com n=2, para gerar atributos com palavras compostas, de modo que termos que apareceram sequencialmente nos documentos foram unidos, considerando as palavras adjacentes que ocorreram com uma determinada frequência e, assim, criou-se uma melhor representação semântica do texto completo. Mesmo com as limitações dessa técnica, percebeu-se uma melhora na qualidade das características extraídas e dos rótulos dos *clusters* formados. Portanto, a possibilidade de representar as notícias nos vetores de características usando termos compostos, ao invés de somente palavras simples, permite uma melhor representação vetorial de textos em português.

Outra técnica usada para reduzir a dimensionalidade dos textos foi o *stemming.* Assim, para verificar qual *stemmer* se comportaria melhor com um corpus em português, testaram-se o RSLP, o Porter e o Snowball. O resultado dos testes mostrou que o RSLP apresentou melhor desempenho para esse tipo de texto.

Além disso, utilizou-se a técnica de *pruning* (poda), pois os termos muito frequentes e os que aparecem poucas vezes não colaboram para discriminação e semelhança entre textos. Consequentemente, eles não contribuem na formação de grupos apropriados. Isso ajudou também a evitar a formação de vários *clusters* com poucos documentos, pois a presença de muitas palavras pode contribuir para a formação de grupos contendo apenas uma ou duas notícias. A partir dos resultados dos experimentos, observou-se que essa técnica não teve grande influência nos resultados dos agrupamentos. Assim, optou-se por usar também a Frequência do Termo-Inverso da Frequência nos Documentos (TF-IDF) para ponderar a relevância das palavras em relação ao corpus, pois uma palavra que aparece, por exemplo, cem vezes em um texto não significa que ela seja cem vezes mais relevante.

Além dos aperfeiçoamentos realizados durante a fase de pré-processamento, utilizou-se o método Elbow para calcular o número ideal de grupos para cada amostra de notícias usada nos experimentos, pois o algoritmo *k-means* necessita que o valor de k seja informado pelo usuário. Dessa forma, conhecendo antecipadamente o número de grupos, obtêm-se melhores resultados no processo de *clustering*. Os resultados dos testes apontaram que mais notícias foram agrupadas corretamente e os rótulos ficaram mais bem definidos quando atribuiu a k o valor da quantidade de assuntos variados que continham na amostra.

Por conseguinte, tais técnicas poderão, inclusive, ser avaliadas futuramente em outros tipos de informação como, por exemplos, emails, artigos científicos, textos jornalísticos, documentos jurídicos, documentos gerenciais e organizacionais e, assim, contribuir para o surgimento de novas pesquisas na área.

Observou-se durante os experimentos que a etapa de pré-processamento exige um esforço especial para garantir a qualidade dos dados. A complexidade da língua portuguesa, a necessidade de atualização da lista de *stop words,* a detecção de quais características são mais importantes e, em geral, a complexidade dos problemas relacionados à alta dimensionalidade dos dados foram evidenciados durante todo o processo desta pesquisa.

As medidas de distância também desempenham um papel importante no processo de *clustering*. Porém, não foi encontrada uma que seja mais adequada para todos os tipos de problemas de agrupamento. Portanto, é importante testar um conjunto de medidas para verificar qual combinação produz melhores resultados, pois a escolha delas deve ser feita com base no conjunto de dados e no algoritmo escolhido.

Os resultados mostraram que a precisão obtida pela técnica de *clustering* está relacionada à qualidade dos dados, ou seja, a falta de características em comum nos textos dificulta a identificação de semelhanças entre as matérias. Assim, é importante que o sistema de recuperação das notícias também seja eficaz na coleta, pois a qualidade da amostra influência diretamente nos resultados.

Nesta pesquisa, realizaram-se alguns experimentos utilizando três métodos de *clustering* (particionado, propagação por afinidade e hierárquico) para verificar a separabilidade das notícias publicadas no idioma português. Apesar de alguns *clusters* possuírem homogeneidade entre seus elementos, observou-se que em outros grupos ocorreu uma mistura entre diferentes assuntos. Fato que pode ter acontecido devido a necessidade de mais amostras para compor as métricas ou por uma seleção de atributos mais abrangentes que possam agregar mais informação aos dados.

O algoritmo *k-means* conseguiu os melhores resultados pois obteve bom desempenho nos três experimentos. No último teste, considerando a quantidade de notícias agrupadas, ele conseguiu uma taxa acima de 90% de acerto e agrupou 84% dos textos. Se considerar as particularidades das notícias, esse algoritmo obteve praticamente 100% de acerto, ponderando apenas as matérias que foram congregadas. Enquanto o *Hierarchical Clustering* apresentou dificuldades nos dois primeiros experimentos, visto que notícias semelhantes foram alocadas em grupos diferentes. Porém, essa técnica teve um bom desempenho ao usar uma amostra pequena e com assuntos mais diversificados. Já o algoritmo *Affinity Propagation* apresentou divergência quanto ao número ideal de *clusters*, sendo um resultado que deve ser levado em consideração em trabalhos futuros para verificar a separabilidade entre outras amostras de textos, pois essa técnica apresentou bom desempenho ao agrupar por semelhança, porém subdividiu a coleção em vários grupos, ou seja, um número bem maior do que o valor de k encontrado pelo método *Elbow* e pelo Coeficiente da Silhueta.

Concluiu-se que o tamanho da amostra também influencia nos resultados e que quanto maior for o número de *clusters*, mais forte é a estrutura encontrada intragrupo. Além disso, os algoritmos conseguem melhor desempenho quanto mais diversificado for o corpus e quanto mais bem definidas forem as características dos textos.

Portanto, para melhorar a qualidade do agrupamento é necessária uma análise das notícias recuperadas, visto que muitos informes relatam mais de um assunto em seu conteúdo, dificultando, assim, a tarefa de *clustering*. Mas, ao mesmo tempo, a possibilidade de descobrir agrupamentos não evidentes é a principal vantagem desta técnica.

A partir da revisão da literatura, constatou-se que existem muitos algoritmos de *clustering*, porém não existe um universal que seja capaz de revelar todas as variedades de estruturas que podem estar presentes em uma coleção de textos (MENDES, 2017). Hartigan (1985 apud Faceli *et. al*, 2017, p. 204) afirma que “diferentes agrupamentos são corretos para diferentes propósitos, assim, não podemos dizer que um agrupamento é melhor”. Além disso, é complicado estabelecer uma boa métrica de desempenho para essa técnica, pois, por ser um método não supervisionado, fica difícil avaliar a exatidão dos resultados produzidos pelos algoritmos. Além do mais, medidas de avaliação de *clusters* podem produzir resultados diferentes e, como consequência, a melhor solução de agrupamento pode variar, dependendo da métrica escolhida. Desse modo, para uma melhor avaliação, é necessária a presença de um especialista, ou seja, uma pessoa que tenha o conhecimento sobre o domínio do problema.

Ao mesmo tempo, as técnicas de *clustering* têm a potencialidade de facilitar o acesso às notícias disponibilizadas pela grande mídia, na medida que subdivide uma coleção em grupos menores, o leitor pode escolher o grupo que seja mais de seu interesse, não sendo necessário pesquisar em vários jornais *on-line* para encontrar as informações mais significantes.

Por conseguinte, considerando a importância da área da análise de textos para facilitar e agilizar a extração de informações, principalmente as disponibilizadas na web, pesquisadores podem conduzir novos estudos a partir desta pesquisa, tendo em vista os poucos trabalhos que exploram documentos disponibilizados no idioma português. Além disso, considera-se que pesquisas na área de agrupamento de notícias são importantes, na medida em que aumenta a quantidade de documentos disponibilizados na internet, técnicas para detecção automática de informações relevantes são cada vez mais necessárias.

Todavia, esta pesquisa não esgota o estudo sobre o tema e existem várias perspectivas de trabalhos futuros para dar continuidade aos estudos desta tese. Uma lacuna que está em aberto é o fato de nenhum algoritmo de *stemming* usar regras para remoção dos prefixos, isso é uma questão a ser implementada.

Outro trabalho futuro promissor se refere à integração de ontologias em diferentes etapas do processo de análise das notícias de tal forma que a semântica dos textos também seja levada em consideração. Como todo o experimento desta tese foi realizado em Python e a biblioteca NLTK contém uma interface de acesso ao dicionário WordNet, um primeiro passo seria usar o conhecimento externo do WordNet.Br na fase de pré-processamento. O problema é que WordNet é muito usado para o inglês e o WordNet.Br ainda está em fase de experimentação. Portanto, estudos e testes são necessários para a implementação dessa ideia.

Também seria interessante um estudo mais aprofundado sobre Modelagem de Tópicos de forma que os termos extraídos pudessem ser usados para adicionar informações aos *clusters*, pois essa técnica permite exibir tópicos semanticamente mais coerentes, minimizando a dependência de conhecimentos especializados necessários para a interpretação dos textos.

Além do que já foi exposto, seria importante também avaliar outros algoritmos de *clustering* como, por exemplo, o *Expectation-Maximization* e o *Fuzzy C-Meanse*, pois esses algoritmos trabalham com partições sobrepostas e não rígidas, ou seja, eles têm a capacidade de atribuir a cada documento a probabilidade de pertinência a cada grupo, pois muitas vezes um texto aborda assuntos diversos, o que dificulta a sua alocação em apenas um grupo.

Por fim, também seria relevante testar a metodologia em empresas para categorização e seleção de documentos gerenciais e organizacionais e, assim, verificar o desempenho das técnicas de *clustering* com esse tipo de corpus.

Contudo, espera-se que as técnicas de *clustering* discutidas neste estudo tenham sido bem compreendidas e que os desafios sejam amplamente discutidos na comunidade científica. Que surjam novas oportunidades de pesquisas e que sejam criadas novas vertentes e aplicações inovadoras na área, contribuindo, dessa forma, para a solução dos problemas existentes.

REFERÊNCIAS BIBLIOGRÁFICAS

ABDULSAHIB, A. K.; KAMARUDDIN, S. S**.** Graph Based Text Representation for Document clustering. **Journal of Theoretical and Applied Information Technology** v. 76, n. 1, jun. 2015. Disponível em: https://www.researchgate.net/publication/281944315_Graph_based_text_representation_for_document_clustering. Acesso em: 20 set. 2019.

ABUALIGAH, L.M.; KHADER, A.T.; AL-BETAR, M.A. Multi-objectives-based text clustering technique using K-mean algorithm. *In*: 7TH INTERNATIONAL CONFERENCE ON COMPUTER SCIENCE AND INFORMATION TECHNOLOGY (CSIT). **IEEE**, p. 1-6, 2016. Disponível em: http://ieeexplore.ieee.org.ez27.periodicos.capes.gov.br/document/7549464/. Acesso em 23 mar. 2018.

AFONSO, A. R. **Um Sistema para Indexação e Agrupamento de Artigos Científicos em Português Brasileiro Utilizando Computação Evolucionária**. 2013. 158 f. Tese (Doutorado em Ciência da Informação) – Faculdade de Ciência da Informação - Universidade de Brasília, Brasília, 2013. Disponível em: http://repositorio.unb.br/bitstream/10482/15480/1/2013_AlexandreRibeiroAfonso.pdf. Acesso em: 30 set. 2019.

AGGARWAL, C.C; ZHAO, Y.; YU, P.S. On Text Clustering with Side Information, **ICDE Conference,** 2012. Disponível em: https://www.cs.uic.edu/~yzhao/research/papers/text-icde.pdf. Acesso em 22 mar. 2018.

ALLAHYARI, M. et al. A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques. *In*: PROCEEDINGS OF KDD BIGDAS, 2017, Halifaz, Canada. **arXiv**. ago. 2017. Disponível em: https://arxiv.org/pdf/1707.02919.pdf. Acesso em 30 ago. 2019.

ALMEIDA, L. G. P. de. **Análise de algoritmos de agrupamento para base de dados textuais**. 2007. 139 f. Dissertação. (Mestrado em Modelagem Computacional). Laboratório Nacional de Computação Científica. Petrópolis, Rio de Janeiro, 2007. Disponível em:https://tede.lncc.br/bitstream/tede/75/1/Texto%20completo. Acesso em: 24 mar. 2019.

ALVARES, R. V. **Algoritmos de Stemming e o estudo de Proteomas**. 2014. Tese (Doutorado em Engenharia de Sistemas e Computação) - Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia (COPPE), Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2014. Disponível em: https://www.cos.ufrj.br/uploadfile/1398446767.pdf. Acesso em 16 dez. 2018.

ALVES, A. I.M. **Modelo de representação de texto mais adequado à classificação**. 2010. 92 f. Dissertação. (Mestrado em Engenharia Informática). Instituto Superior do Porto. Disponível em: http://recipp.ipp.pt/handle/10400.22/1908. Acesso em: 12 ago. 2018.

ANDRADE, L.M.S.; BARROS, R.C.; SANTOS, M.A.B. **Processamento de Linguagem Natural (PLN): Ferramentas e Desafios**. Instituto Federal de Educação, Ciência e Tecnologia do Sertão Pernambucano-campus Salgueiro. [s.d.]

APTÉ, C.; DAMERAU, F.; WEISS, S. M. Automated learning of decision rules for text categorization. **ACM Transactions on Information Systems**, New York, v. 12, n.3, p.233–251, jun. 1994. Disponível em: http://users.softlab.ntua.gr/facilities/public/AD/Text%20Categorization/Automated%20Learning%20of%20Decision%20Rules%20for%20TextCategorization.pdf. Acesso em: 30 set. 2019.

ARANHA, C.; PASSOS, E. A Tecnologia de Mineração de Textos. **RESI-Revista Eletrônica de Sistemas de Informação**, v. 2, p. 1–8, 2006. Disponível em: http://www.periodicosibepes.org.br/index.php/reinfo/article/download/171/66. Acesso em: 30 set. 2019.

BAEZA-YATES, R.; RIBEIRO-NETO, B. **Modern Information Retrieval:** The Concepts and Technology behind Search. 2. ed. Harlow, England: Addison-Wesley, 2011.

BAEZA-YATES, R. RIBEIRO-NETO, B. **Recuperação de Informação**: Conceitos e tecnologia das máquinas de busca. Tradução técnica: Leandro Krug Wives, Viviane Pereira Moreira. 2. ed. Porto Alegre: Bookman, 2013.

BANDEIRA, V. Usando o *SpaCy* para um tratamento NPL, **Medium.com**, 2018. Disponível em: https://medium.com/@van3ssabandeira/o-famoso-spacy-90afb683b6fe. Acesso em: 03. Abr. 2020.

BARBOSA, B. Minerando informações de textos. *In:* **Trilha de Machine Learning**. 2017. Disponível em: https://www.slideshare.net/BarbaraBarbosaClaudi/minerando-informaes-de-textos. Acesso em: 28 fev. 2019

BASTOS, V. M. **Ambiente de Descoberta de Conhecimento na Web para a Língua Portuguesa**. 2006. 133 f. Tese (Doutorado em Ciência em Engenharia Civil) – Departamento de Engenharia Civil, Universidade Federal do Rio de Janeiro, Rio de Janeiro. Disponível em: http://www.coc.ufrj.br/pt/documents2/doutorado/2006-2/825-valeria-menezes-bastos-doutorado/file. Acesso em: 30 set. 2019.

BENGFORT, B.; OJEDA, T.; BILBRO, R. **Applied Text Analysis with Python**: Enabling Language-Aware Data Products with Machine Learning. O'Reilly Media, 2018

BIRD, S.; KLEIN, E.; LOPER, E. **Natural language processing with Python**: analyzing text with the Natural Language Toolkit. Sebastopol: O'Reilly, 2009. 502 p

BONETTE, R. P. **Análise de ferramentas de opinion mining aplicadas a redes sociais com foco em inovação de produtos**. 2011. 87 p. Monografia (Graduação em Sistemas de Informação) - Universidade Federal de Lavras, Lavras, 2011.

BORGES, H. B. **Classificador Hierárquico Multirrótulo Usando uma Rede Neural Competitiva**. 2012. 188 p. Tese (Doutorado) – Programa de Pós-graduação em Informática. Pontifícia Universidade Católica do Paraná (PUCPR), Curitiba, Paraná, 2012. Disponível em: https://www.ppgia.pucpr.br/pt/arquivos/doutorado/teses/2012/helyane_borges.pdf. Acesso em: 13 nov. 2018.

BORKO, H**.** Information science: what is it? **American Documentation**, Washington, v. 19, n. 1, p. 3-5, jan. 1968. Disponível em: https://www.marilia.unesp.br/Home/Instituicao/Docentes/EdbertoFerneda/MRI%2001%20-%20Borko,%20H%20-%201968.pdf. Acesso em: 30 set. 2019.

BOURAS, C., TSOGKAS, V. **W-kmeans:** Clustering News Articles Using WordNet. In Proceedings of KES 2010, Part III. v.6268, p. 379-388, 2010. Disponível em: http://ru6.cti.gr/ru6-old/publications/116262780379.pdf. Acesso em 30 set. 2019.

CAMARGO, Y. B. L. de. **Abordagem linguística na classificação automática de textos em português**. 2007. 89 p. Dissertação (Mestrado em Ciências em Engenharia Elétrica) - Programas de Pós-Graduação de Engenharia da Universidade Federal do Rio De Janeiro, Rio de Janeiro. Disponível em: http://www.pee.ufrj.br/index.php/pt/producao-

academica/dissertacoes-de-mestrado/2007-1/2007062502-2007062502/file. Acesso em: 06 mar. 2019.

CAMPOS, R. N. T. **Agrupamento Automático de Páginas Web Utilizando Técnicas de Web Content Mining**.2005. 100 f. Dissertação (Mestrado em Engenharia Informática) - Universidade da Beira Interior, Covilhã.

CARLANTONIO, L. M. di. **Novas Metodologias Para Clusterização de Dados**. 2001. 257 p. Tese (Mestrado em Ciências em Engenharia Civil) – Universidade Federal do Rio de Janeiro, Rio de Janeiro.

CARRILHO JUNIOR, J. R.; PASSOS, E. P. L. (Orientador). **Desenvolvimento de uma Metodologia para Mineração de Textos**., 2007. 96p. Dissertação (Mestrado em Engenharia Elétrica) - Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro. Disponível em: https://www.maxwell.vrac.puc-rio.br/colecao.php?strSecao=resultado&nrSeq=11675@1. Acesso em: 11 jun. 2019.

CARVALHO A. X. Y. et al., Clusterização Hierárquica Espacial com Atributos Binários. **Instituto de Pesquisa Econômica Aplicada (ipea)**, 2009. Disponível em: http://www.ipea.gov.br/portal/images/stories/PDFs/TDs/td_1428.pdf. Acesso em: 24 jan. 2019.

CAVALCANTI, C. R. **Indexação e tesauro**: metodologia e técnica, Brasília, ABDF, 1978.

CESARINO, M. A. N. B. **Sistemas de recuperação da informação**. Revista da Escola de Biblioteconomia da UFMG, v. 14, n. 2, p. 157-168, 1985. Disponível em: <http://www.brapci.inf.br/v/a/9051>. Acesso em: 08 mar. 2018.

CHANDRASEKARAN, B.; JOSEPHSON, J. R.; BENJAMINS, V. R. What Are Ontologies, and Why Do We Need Them? *In:* **IEEE Intelligent Systems**, 14(1):20-6, 1999

CHEESEMAN, P. & J. STUTZ. Bayesian Classification (Auto Class): Theory and Results. *In*: FAYYAD, U. M.; PIATETSKY-SHAPIRO, G.; SMYTH, P.; UTHURUSAMY, R. (Eds.), **Advances in Knowledge Discovery and Data Mining.** American Association for Artificial Intelligence. Menlo Park, CA. 1996, p. 153-180

CHOMSKY, N., 2002. **Syntactic Structures**.2 ed. Berlin, Mouton de Gruyter.

COELHO, G. P.; ZUBEN, F. J. V.; ATTUX, R. R. F. **Comitês de Máquinas: Ensembles e Misturas de Especialistas**. Apresentação. Disponível em: https://slideplayer.com.br/slide/1234786/. Acesso em: 06 mar. 2019.

CORDEIRO, J.P.C. **Extração de Elementos Relevantes em Texto/Páginas da World Wide Web**. 2003. 174 p. Dissertação (Mestrado em Inteligência Artificial e Computação) - Faculdade de Ciências da Universidade do Porto. Portugal.

CORREIA, D. J. B. **Classificação de Dados Biológicos:** Características e Classificadores. 2012. 85 p. Dissertação (Mestrado em Engenharia Informática e Sistemas – Desenvolvimento de Software) – Instituto Superior de Engenharia de Coimbra. Disponível em: https://comum.rcaap.pt/bitstream/10400.26/17234/1/Daniel-Joao-Correia.pdf. Acesso em: 17 mar. 2019.

CURRÁS, E. **Ontologias, taxonomia e tesauros em teoria de sistemas e sistemática**. Tradução Jaime Robredo. Brasília: Thesaurus, 2010. 182 p.

da SILVA CONRADO, M.; FELIPPO, A. D.; SALGUEIRO PARDO, T. A.; REZENDE, S. O. (2014). A survey of automatic term extraction for Brazilian Portuguese. **Journal of the Brazilian Computer Society**, 20 (1):12.

DAS. **Formação Cientista de Dados**. Curso de Machine Learning ofertado por Data Science Academy. E-book. 2017

DEEPTHI, A. L.; PRASAD, J.V.D. Hierarchal clustering and similarity measures along with multi representation. **IJRET: International Journal of Research in Engineering and Technology**, v. 2, n. 8, p. 78-79, 2013. Disponível em: https://www.academia.edu/7527139/HIERARCHAL_CLUSTERING_AND_SIMILARITY_MEASURES_ALONG_WITH_MULTI_REPRESENTATION?auto=download. Acesso em 21 dez. 2019.

DENECKE, K. **Using sentiwordnet for multilingual sentiment analysis**. *In:* ICDE Workshops 2008, pp. 507–512 (2008)

DOBRE, C.; XHAFA, F. Intelligent services for Big data science. **Future Generation Computer Systems**, v. 37, p. 267–281, 2014. Disponível em: https://doi. org/10.1016/j.future.2013.07.014. Acesso em 21 dez. 2019.

DRISCOLL, K.; THORSON, K. Searching and Clustering Methodologies: Connecting Political Communication Content across Platforms. **Annals of the American Academy of Political and Social Science**, v. 659, n. 1, p. 134-148, 2015.

DUMAIS, S.; PLATT, J.; HECKERMAN, D,; Sahami, M. (1998**). Inductive learning algorithms and representations for text categorization**.In Proc. of the 7th International Conference on Information and Knowledge Management (CIKM 98).

EBECKEN, N. F. F.; LOPES, M. C. S.; COSTA, M. C. de A. Mineração de Textos. *In*: REZENDE, S. O. **Sistemas Inteligentes**: Fundamentos e Aplicações, 1. ed. São Paulo: Manole, 2003, cap.13, p. 337-370.

FACELI, Katti. et al. **Inteligência Artificial: uma abordagem de aprendizagem de máquina**. Rio de Janeiro: LTC, 2011.

FAYYAD, U. M. et al. From data mining to knowledge discovery: an overview. *In*: **Advances in knowledge discovery and data mining**. California: AAAI/The MIT, 1996. p.1-34

FALEIROS, T. de P.; LOPES, A. de A. **Modelos probabilísticos de tópicos**: desvendando o Latent Dirichlet Allocation. 2016. Relatório Técnico - Instituto de Ciências Matemáticas e de Computação, USP, São Paulo. Disponível em: http://conteudo.icmc.usp.br/CMS/Arquivos/arquivos_enviados/BIBLIOTECA_158_RT_409.pdf. Acesso em: 10 jan. 2020.

FELDMAN, R.; SANGER, J. T**he Text Mining Handbook**. Cambridge. New York, 2006.

FERNEDA, Edberto. **Ontologia como recurso de padronização terminológica em um Sistema de Recuperação de Informação**. 2013. 97 f. Relatório de Pesquisa (Pós-Doutorado) – Departamento de Ciência da Informação, Centro de Ciências Sociais Aplicadas, Universidade Federal da Paraíba, João Pessoa. Disponível em:https://www.marilia.unesp.br/Home/Instituicao/Docentes/EdbertoFerneda/pos-doutorado.pdf. Acesso em: 08 mar. 2018.

FIEL, C. **O que é Pesquisa Quali-Quantitativa?**, 2017. Disponível em: https://pt.lifeder.com/pesquisa-quali-quantitativa/. Acesso em: 31 mar. 2020.

FOSKETT, D. **Thesauros**. In k. S. Jones and P. Willet, editors, Readings in Information Retrieval, pages 111-134. Morgan Kaufmann Publishers, Inc., 1997.

GAZZOLA, Oliver. Uma aplicação para análise de opiniões com base em representação linguístico-computacional. UNISINOS, São Leopoldo, 2011.

GONÇALVES, M. **Classificação de Textos**. *In:* BAEZA-YATES, R. RIBEIRO-NETO, B. Recuperação de Informação: Conceitos e tecnologia das máquinas de busca. Tradução técnica: Leandro Krug Wives, Viviane Pereira Moreira. 2. ed. Porto Alegre: Bookman, 2013. p. 277-338.

GODFREY, D. et al. A **Case Study in Text Mining: Interpreting Twitter Data From World Cup Tweets**. v. 5, p. 1–11, 2014. Disponível em: https://arxiv.org/pdf/1408.5427.pdf. Acesso em: 08 mar. 2018.

GONZALEZ, M. LIMA, V. L. S. Recuperação de Informação e Processamento da Linguagem Natural. XXIII Congresso da Sociedade Brasileira de Computação, Campinas, 2003. **Anais do III Jornada de minicursos de Inteligência Artificial**, Volume III, p.347-395. Disponível em: https://www.researchgate.net/publication/228608574_Recuperacao_de_Informacao_e_Processamento_da_Linguagem_Natural. Acesso em 02 mar. 2019.

GONZALEZ, M.; TOSCANI, D.; ROSA, L.; DORNELES, R.; LIMA, V.L.S. de. (2003). Normalização de itens lexicais baseada em sufixos. **Workshop em tecnologia da informação e da linguagem humana**. Disponível em: http://nilc.icmc.usp.br/til/til2003_English/oral/gonzalez_21.pdf. Acesso em 08 ago. 2018.

GOLÇALVES, M. L. **Uma abordagem para a Análise de Agrupamentos baseada em Mapas de Kohonen segmentados por Morfologia Matemática e Índices de Validação**. (Relatório Técnico) – Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, 2005. Disponível em: ftp://ftp.dca.fee.unicamp.br/pub/docs/techrep/2005/RT-DCA-FEEC-02-2005.pdf. Acesso em 15 jun. 2019.

GOLDSCHMIDT, R.; PASSOS, E.; BEZERRA, E. **Data mining**: conceitos, técnicas, algoritmos, orientações e aplicações. Rio de Janeiro: Elsevier, 2015.

GUARINO, N. Formal Ontology in Information Systems. *In*: **Proceedings of FOIS'98**, Trento, Italy, p. 3-15, 1998. Disponível em: http://osm.cs.byu.edu/CS652s04/Gua98Formal.pdf. Acesso em: 3 jul. 2017.

HAIR, J. F. *et al*. **Análise multivariada de dados**. Trad. Adonai S. Sant'Anna e Anselmo C. Neto. 5 ed. Porto Alegre: Bookman, 2005.

HALKIDI, M.; BATISTAKIS, Y.; VAZIRGIANNIS, M. On clustering validation techniques. **Journal of intelligent information systems**. v. 17, p. 107-145, 2001. Disponível em: http://web.itu.edu.tr/sgunduz/courses/verimaden/paper/validity_survey.pdf. Acesso em: 08 mar. 2018.

HAN, J.; KAMBER, M.; PEI, J. **Data Mining: Concepts and Techniques**, 3rd edition, Morgan Kaufmann, 2011.

HAAS, Stephanie W, Natural Language Processing: Toward large-scale, robust systems, **Annual Review of Information Science and Technology**, v. 31, p. 83-119, 1996.

HADJIDJ, R.; DEBBABI, M.; LOUNIS, H.; IQBAL, F.; SZPORER, A.; BENREDJEM, D. Towards an integrated e-mail forensi analysis framework. **Digital Investigation**, v. 5 n. 3, p. 124–137, 2009. Disponível em: https://docplayer.net/8121151-Towards-an-integrated-e-mail-forensic-analysis-framework.html. Acesso em: 15 out. 2018

HIPPISLEY, A. Lexical analysis. In **Handbook of natural language processing**. University of Kentucky, 2010.Disponível em: https://uknowledge.uky.edu/cgi/viewcontent.cgi?referer=https://www.google.com/&httpsredir=1&article=1066&context=lin_facpub. Acesso em: 01 mar. 2019.

HIRATA, N. S. T. **Classificador de Bayes**. 2007. Disponível em:http://www.vision.ime.usp.br/~nina/cursos/ibi5031-2007/pr.pdf. Acesso em 06 mar. 2019.

HUANG, **A. Similarity measures for text document clustering**. 6th New Zealand Computer Science Research Student Conference, n. April, p. 49–56, 2008.

ILIASSICH, J. **Clustering Algorithms**: From Start To State Of The Art.

IQBAL, F. et al. Mining writeprints from anonymous e-mails for forensic investigation. **Digital Investigation**, v. 7, n. 1–2, p. 56–64, 2010. Disponível em: https://www.academia.edu/13100817/Mining_writeprints_from_anonymous_e-mails_for_forensic_investigation. Acesso em: 20 jan. 2019.

JACOB, E. K**. Ontologies and the semantic web**. Bulletin of the American Society for Information Science and Technology, Silver Spring, p. 16-18, Apr./May 2003. Disponível em . Acesso em: 09 jul. 2017.

JACOB, E. K. Classification and Categorization: Drawing the Line. **2nd ASIS SIG/CR Classification Research Workshop**, 63-80. doi:10.7152/acro.v2i1.12548, 1991. Disponível em: https://www.researchgate.net/publication/273895978_Classification_and_Categorization_Drawing_the_Line. Acesso em: 17 mar. 2019.

JACOB, E. K. Classification and categorization: a difference that makes a difference". **Library Trends**, vol. 52, nº 3, p. 515-540, 2004. Disponível em: https://www.ideals.illinois.edu/handle/2142/1686. Acesso em: 17 mar. 2019.

JAIN, A., DUBES, R. **Algorithms for Clustering Data**. Prentice-Hall, Englewood Cliffs, NJ. 1988.

JIANG, D.; TANG, C.; ZHANG, A. Cluster analysis for gene expression data: A survey. **IEEE Transactions on Knowledge and Data Engineering**, v. 16, n. 11, p. 1370–1386, 2004.

JINHUAXU; HONGLIU, "**Web User Clustering Analysis based on K-Means Algorithm**", IEEE International Conference on Information, Networking and Automation, 2010.

KOTHARI, C. R. (2004), Research Methodology: Methods and Techniques, (Second Edition), **New Age International Publishers**.

KAUFMAN, L.; ROUSSEEUW, P. J. **Finding Groups in Data: na introduction to Cluster Analysis**, John Wiley and Sons. 1990

KLINCZAK, M. N. M. **Identificação e propagação de temas em redes sociais.** 2016. 151 f. Dissertação (Mestrado em Computação Aplicada) - Universidade Tecnológica Federal do Paraná. Disponível em: http://repositorio.utfpr.edu.br/jspui/bitstream/1/2304/1/CT_PPGCA_M_Klinczak%2C%20Marjori_2016.pdf. Acesso em: 20 dez. 2019.

KONCHADY, M. **Text Mining Application Programming**. Charles River Media, Boston, 2006.

KOWALSKi, G. **Information Retrieval Systems: Theory and Implementation**: Kluwer Academic Publishers. 1997 (Multimedia Information Retrieval: Content-Based Information Retrieval from Large Text and Audio Databases).

KRISHNA, S.; BHAVANI, S. An efficient approach for text clustering based on frequent itemsets. **European Journal of Scientific Research**, v. 42, n. 3, p. 385–396, 2010. Disponível em: https://pdfs.semanticscholar.org/b74a/af14bbbb872fd7c5720d509678d7f83477a8.pdf. Acesso em 10 out. 2017.

KUGLER, M.; TORTATO JÚNIOR, J. T.; LOPES, H. S. Desenvolvimento de uma Rede Neural LVQ em Linguagem VHDL para Aplicações em Tempo Real,**VI Congresso Brasileiro de Redes Neurais**, 2003. Disponível em http://www.cpgei.cefetpr.br/~hslopes/publicacoes/2003/cbrn2003c.pdf.Acesso em: 05 ago. 2019.

LACHI, R. L.; ROCHA, H. V. **Aspectos básicos de clustering**: conceitos e técnicas. 2005, 26 p. Relatório Técnico – Instituto Federal de Computação da Universidade Estadual de Campinas. Disponível em: http://www.ic.unicamp.br/~reltech/2005/05-03.pdf. Acesso em: 30 set. 2019.

LADEIRA, A. P. **Processamento de Linguagem Natural**: Caracterização da produção científica dos pesquisadores brasileiros. 2010. 159 f. Tese (Doutorado em Ciência da Informação) – Universidade Federal de Minas Gerais. Disponível em: https://repositorio.ufmg.br/bitstream/1843/ECID-8B3Q6C/1/tese_anapaulaladeira_cd.pdf. Acesso em 25 nov. 2019.

LAMA, P. **Clustering System Based on Text Mining Using the K-Means Algorithm**. 2013. Disponível em: https://www.theseus.fi/bitstream/handle/10024/69505/Lama_Prabin.pdf?sequence=1&isAllowed=y. Acesso em: 10 fev. 2018.

LANCASTER, F. W. **If you want to evaluate your library**. Imprint, Champaign, IL: University of Illinois, Graduate School of Library and Information Science, 1993.

LANCE, E. **Identification of Topics and Their Evolution in Management Science: Replicating and Extending an Expert Analysis Using Semi-Automated Methods**. 2017. Dissertação (Mestrado em Ciências Aplicadas) – Departamento deTechnology Innovation Management Carleton University Ottawa, Ontario, 2017. Disponível em: https://curve.carleton.ca/system/files/etd/b2119eac-d9df-4303-8e83-d843f7075863/etd_pdf/16808ca4cd8d07adc036516574f28b01/lance-identificationoftopicsandtheirevolutionin.pdf. Acesso em: 10 jan. 2020.

LAW, M. H. C.; TOPCHY, A. P.; JAIN, A. K. **Multiobjective data clustering**. Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004., v. 2, p. 32–41, 2004. 6

LEE, D. D.; SEUNG, H. S. **Algorithms for non-negative matrix factorization**. *In*: NIPS[S.l.: s.n.], 2000. p. 556–562.

LEITÃO, P. J. de O. **Organização da Informação em Subject Gateways**. 2004. Dissertação (Mestrado em Estudos de Informação e Bibliotecas Digitais) - Departamento de Ciências e Tecnologias da Informação - Instituto Superior de Ciências do Trabalho e da Empresa – ISCTE, Instituto Universitário De Lisboa, Portugal, 2004.

LEWIS, D. D. (1992). **An evaluation of phrasal and clustered representations on a text categorization task**. In Proceedings of the 15th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 37–50.

LIMA, G. A. B. de O. Modelos de categorização: apresentando o modelo clássico e o modelo de protótipos. **Perspectivas em Ciência da Informação**, v.15, n.2, p.108-122, maio/ago. 2010. Disponível em: http://www.scielo.br/pdf/pci/v15n2/a08v15n2.pdf. Acesso em: dez. 2018

LIMA, F. A. N. de. **Agrupamento de fornos de redução de alumínio utilizando os algoritmos Affinity Propagation, Mapa auto-organizável de Kohonen (SOM), Fuzzy C–Means e K–Means**. 2017. Dissertação (Mestrado em Engenharia Elétrica) - Instituto de Tecnologia – ITEC, Universidade Federal do Pará, Belém, 2017.

LINDEN, R. Técnicas de Agrupamento. **Revista de Sistemas de Informação da FSMA**, v. 4, p. 18–36, 2009.

LINOFF, G.S.; BERRY, M. J. A. **Mining the Web. Transforming Customer Data into Customer Value**. New York: Wiley, 2001.

LIU, B. **Web Data Mining. Exploring** Hiperlinks, Contents, and Usage Data. Springer: Chicago, 1 ed. 2007.

LIU, B. **Web Data Mining. Exploring** Hiperlinks, Contents, and Usage Data. Springer: Chicago, 2 ed. 2011.

LIU, B., ZHANG, L., 2012, A survey of opinion mining and sentiment analysis, Mining Text Data, 415- 463.Disponível em: http://www.cs.unibo.it/~montesi/CBD/Articoli/SurveyOpinionMining.pdf. Acesso em: 27 dez. 2018.

LIU, L.; KANG, J.; YU, J.; WANG, Z. A comparative study on unsupervised feature selection methods for text clustering. **Proceeding of Natural Language Processing and Knowledge Engineering'05**, 2005.

LOBO, V. **Árvores de decisão**. Nota de aula, 2010. Disponível em: http://www.novaims.unl.pt/docentes/vlobo/isegi_dm2/DM2_4_arvores.pdf. Acesso em 06 mar. 2019.

LÖNNBERG, M.; YREGÅRD, L**. Large scale news article clustering**. June, 2013. Disponível em http://publications.lib.chalmers.se/records/fulltext/179841/179841.pdf. Acesso em 08 mar. 2018.

LOPES, M. C. S. **Mineração De Dados Textuais Utilizando Técnicas de Clustering para o Idioma Português**. 2004. 258 f. Tese (Doutorado em Ciências em Engenharia Civil) - COPPE/UFRJ. Rio de Janeiro.

LUFT, C. P. **Moderna Gramática Brasileira**.3. ed. São Paulo: Globo, 2008.

LUHN, H. P. The automatic creation of literature abstracts, **IBM Journal os Research and Development**, vol. 2, no. 2, pp. 159–165, 1958.Disponível em: http://courses.ischool.berkeley.edu/i256/f06/papers/luhn58.pdf. Acesso em: 19 mar. 2019.

MADEIRA, R. de O. C. **Aplicação de técnicas de mineração de texto na detecção de discrepâncias em documentos fiscais**. 2015. 66 f. Dissertação (Mestrado em Modelagem Matemática da Informação). Escola de Matemática Aplicada - Fundação Getúlio Cargas, Rio de Janeiro. Disponível em: https://bibliotecadigital.fgv.br/dspace/bitstream/handle/10438/14593/TEXTO%20DISSERTA%C3%87%C3%83O%20VFINAL1.pdf. Acesso em: 08 ago. 2018.

MAHESHWARI, P.; AGRAWAL, J. **Centroid Based Text Clustering**. 2010. Disponível em: http://www.ijest.info/docs/IJEST10-02-09-36.pdf. Acesso em: 23 mar. 2018.

MAIA, L. C.; SOUZA, R. R. Uso de sintagmas nominais na classificação automática de documentos eletrônicos. **Perspectivas em Ciência da Informação**, v. 15, p. 154-172, 2010. ISSN 1413-9936. Disponível em: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1413-99362010000100009. Acesso em 13 nov. 2019.

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

MARKOV, Z.; LAROSE, D. T. **Data Mining the Web: Uncovering Patterns in Web Content, Structure, and Usage**. New Britain: Wiley, 2007.

MATOS, D. **Conceitos Fundamentais de Machine Learning.** Disponível em: http://www.cienciaedados.com/conceitos-fundamentais-de-machine-learning/. Acesso em 08 mar. 2018

MARTHA, A.S. **Recuperação de informação em campos de texto livre de prontuários eletrônicos do paciente baseada em semelhança semântica e ortográfica**. 2005. 93 f. Dissertação (Mestrado em Ciências). Universidade Federal de São Paulo, São Paulo.

MATSUBARA, E. T. **O Algoritmo de Aprendizado Semi-Supervisionado CO-TRAINING e sua Aplicação na Rotulação de Documentos**. 2004. 105 f. Dissertação (Mestrado em Ciências de Computação e Matemática Computacional). Instituto de Ciências Matemáticas e de Computação - ICMC-USP. São Carlos.

MCLACHLAN, G.; BASFORD, K. **Mixture Models: Inference and Applications to Clustering**. New York: Marcel Dekker, 1988.

MIMNO, D. **Topic Modeling Workshop.** University of Maryland - College Park, 2012. Video (30 min). Disponível em: http://journalofdigitalhumanities.org/2-1/the-details-by-david-mimno/. Acesso em: 02 jan. 2020.

MITCHELL, T. M. **Machine Learning**. McGraw-Hill, 1997.

MAÇADA, A. C. G.; CANARY, V. P. (2014). A Tomada de Decisão no Contexto do Big Data: estudo de caso único. **XXXVIII Encontro da ANPAD - EnANPAD**, XXXVIII, 1–17.

MENDES, J. C. **Agrupamento de Dados e suas aplicações**. 2017. 52 f. Monografia (Graduação em Ciência da Computação) – Universidade Federal do Maranhão, São Luís.

Disponível em: https://monografias.ufma.br/jspui/bitstream/123456789/3570/1/JAKELSON-MENDES.pdf. Acesso em 21 nov. 2019.

MONARD, M. C.; BARANAUSKAS, J. A. **Sistemas Inteligentes: Fundamentos e Aplicações, capítulo Conceitos sobre Aprendizado de Máquina**, pp. 89–114. Manole, 2003.

MONTEIRO, L. d. O., I. R. GOMES, et al. (2006). Etapas do Processo de Mineração de Textos – uma abordagem aplicada a textos em Português do Brasil. **Anais do XXVI Congresso da SBC** - I Workshop de Computação e Aplicações, Campo Grande, Centro Universitário do Pará (CESUPA)

MONTEIRO, S. D. et al. Sistemas de recuperação da informação e o conceito de relevância nos mecanismos de busca: semântica e significação**. Encontros Bibli: revista eletrônica de biblioteconomia e ciência da informação**, v. 22, n. 50, p. 161, 2017.

MOOERS, Calvin N. **Zatocoding applied to mechanical organization of knowledge**. American Documentation.2(1), p.20-32, 1951.

MOONEY, R. J.; BUNESCU, R. **Mining knowledge from text using information extraction**. ACM SIGKDD Explorations Newsletter, v. 7, n. 1, p. 3–10, 2005.

MORAIS, E. A. M.; AMBRÓSIO, A. P. L. **Mineração de Textos**, 2007. 30p. Relatório Técnico - Instituto de Informática, Universidade Federal de Goiás. Goiânia, 2007. Disponível em: http://www.inf.ufg.br/sites/default/files/uploads/relatorios-tecnicos/RT-INF_005-07.pdf. Acesso em: 03 jun. 2019.

MORSHED, A.; SINGH, R. **Evaluation and Ranking of Ontology Construction Tools**. 2005. Technical Report DIT-05-013. March 2005. Disponível em: <http://eprints.biblio.unitn.it/archive/00000747/01/013.pdf>. Acesso em 10 jun. 2017.

MOTTA, Dilza Fonseca da. **Método relacional como nova abordagem para construção de tesauros**. Rio de janeiro: SENAI, 1987. 89 p. (Coleção Albano Franco, n. 12).

MÜLLER, A. C. e GUIDO, S**. Introduction to Machine Learning with Python**. O'Reilly Media, 2017.

NASSIF, L. F. DA C. **Técnicas de agrupamento de textos aplicadas à computação forense**. 2011. 91 f. Dissertação (Mestrado em Engenharia Elétrica) – Faculdade de Tecnologia, Universidade Federal de Brasília, Brasília. Disponível em: http://repositorio.unb.br/handle/10482/10718?mode=full. Acesso em: 08 jan. 2018.

NETO, J. M. de O.; TONIN, S. Duarte; PRIETCH, S. S. Processamento de Linguagem Natural e suas Aplicações Computacionais. Universidade Federal de Mato Grosso - Campus Universitário de Rondonópolis. Disponível em: https://www.inpa.gov.br/erin2010/Artigo/Artigo9.pdf. Acesso em: 01 mar. 2019.

NOGUEIRA, B. M. **Avaliação de métodos não supervisionados de seleção de atributos para Mineração de Textos**. 2009. 104 f. Dissertação (Mestrado em Ciência da Computação e Matemática Computacional). Instituto de Ciências Matemáticas e de Computação – ICMC-USP – São Paulo. Disponível em: http://www.teses.usp.br/teses/disponiveis/55/55134/tde-06052009-154832/pt-br.php. Acesso em: ago. 2018.

OLIVEIRA, A. F. de. **Favorecendo o Desempenho do k-Means via Métodos de Inicialização de Centroides**. 2018. 107 f. Dissertação (Mestrado em Ciência da

Computação). Centro Universitário Campo Limpo Paulista – São Paulo. Disponível em: http://www.cc.faccamp.br/Dissertacoes/AndersonFranciscoOliveira.pdf. Acesso em: 22 nov. 2019.

ORENGO, V. M. E HUYCK, C. (2001). A Stemming Algorithm for Portuguese Language. Em: **Symposium on String Processing and Information Retrieval**, 8. Chile. Disponível em: http://www.inf.ufrgs.br/~viviane/rslp/. Acesso em: 08 ago. 2018.

OLIVEIRA, E.; BRANQUINHO FILHO, D. **Automatic classification of journalistic documents on the Internet.** Disponível em:http://www.scielo.br/pdf/tinf/v29n3/0103-3786-tinf-29-03-00245.pdf. Acesso em 20 mar. 2018.

PADILHA, V. A.; CARVALHO, A. C. P. L. F. **Mineração de Dados em Python**. Apostila. Instituto de Ciências Matemáticas e de Computação. Universidade de São Paulo, 2017. Disponível em: https://edisciplinas.usp.br/course/view.php?id=52960. Acesso em 05 jun. 2019.

PAICE, C. D. Method for Evoluation of Stemming Algorithms Based on Error Counting. **Journal of the American Society for Information Science**. 47 (8):632-649, 1996. Disponível em: https://onlinelibrary-wiley.ez27.periodicos.capes.gov.br/doi/epdf/10.1002/%28SICI%291097-4571%28199608%2947%3A8%3C632%3A%3AAID-ASI8%3E3.0.CO%3B2-U. Acesso em 16 dez. 2018

PAK, A., PAROUBEK, P. **Twitter as a Corpus for Sentiment Analysis and Opinion Mining**. Université de Paris-Sud, Laboratoire LIMSI-CNRS, 2010.

PAL S. K., Varum Talwar, Pabitra Mitra, "**Web Mining in Soft Computing Framework: Relevant**, State of the Art and Future Directions", 2002. Disponível em:http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.24.6219&rep=rep1&type=pdf. Acesso em 08 mar. 2018.

PINHEIRO, H. M. **Aplicação de Ensembles de Classificadores na Detecção de Patologias na Coluna Vertebral.** Instituto de Informática – Universidade Federal de Goiás (UFG), 2014. Disponível em: http://inf.ufg.br/~gustavo/courses/grad/pra/final_projetcts/2014.2/Aplica%C3%A7%C3%A3o%20de%20Ensembles%20de%20Classificadores%20na%20Detec%C3%A7%C3%A3o%20de%20Patologias%20na%20Coluna%20Vertebral.pdf Acesso em: 06 mar. 2019.

PONTES N.; ANCHIETA, R. **Text Clustering**. Disponível em: http://slideplayer.com.br/slide/8957690/. Acesso em 22 fev. 2018

PORTER, M. F. (2005). **The stemming algorithm**. Disponível em http://snowball.tartarus.org/algorithms/portuguese/stemmer.html. Acesso em: 08 ago. 2018.

PRATES, W. R. **O que é árvore de decisão** (decision tree)? Exemplos em R, 2018. Disponível em: https://www.wrprates.com/o-que-e-arvore-de-decisao-decision-tree-linguagem-r/. Acesso em 03 mar. 2019.

RADER, E.; WASH, R. 2015. Identifying patterns in informal sources of security information. **Journal of Cybersecurity**, v. 1, n. 1, p. 121–144, 2015. Disponível em: https://doi.org/10.1093/cybsec/tyv008. Acesso em 10 jan. 2020.

REGINA, G.; FACHIN, B. **Recuperação Inteligente da Informação e Ontologias**: um levantamento na área da Ciência da Informação. v. 23, n. 1, p. 259–283, 2009.

REIS, W. da Silva. **Sistema de Diálogo em Linguagem Natural para Serviços de Atendimento ao Cliente**. Trabalho de Conclusão de Curso (Sistema de Informação) - Universidade Federal de Mato Grosso do Sul, Corumbá, 2017. Disponível em: https://cpan.ufms.br/files/2017/10/Westerley-Reis-min.pdf. Acesso em 01 mar. 2019.

REZENDE, S. O. (org). **Sistemas Inteligentes**: fundamentos e aplicações. São Paulo: Manole, 2003.

REZENDE, S. O.; MARCACINI, R. M.; MOURA, M. F. O uso da Mineração de Textos para Extração e Organização Não Supervisionada de Conhecimento. **Revista de Sistemas de Informação da FSMA**, n. 7, p. 7-21, 2007. Disponível em: http://www.fsma.edu.br/si/edicao7/FSMA_SI_2011_1_Principal_3.pdf. Acesso em: 18 mar. 2019.

RODRIGUES, H. J. F. **Ferramenta para Text Mining em Textos Completos**. 2016. 50 f. Dissertação (Mestrado Integrado em Engenharia e Computação). Faculdade de Engenharia – Universidade do Porto. Disponível em: https://sigarra.up.pt/feup/pt/pub_geral.show_file?pi_gdoc_id=827575. Acesso em: 10 ago. 2018.

ROKACH, L.; MAIMON, O. **Data Mining and Knowledge Discovery Handbook**.US: Springer, 2005.

SAIAS, J. M. G. **Uma Metodologia para a construção automática de Ontologias e a sua aplicação em Sistemas de Recuperação de Informação**. 2003. 149 f. Dissertação (Mestrado Engenharia Informática). Universidade de Évora. Portugal. Disponível em: http://host.di.uevora.pt/~jsaias/data/dissertacao-mei-js.pdf. Acesso em: 08 mar. 2018.

SALTON, G. **Automatic Information Organization and Retrieval**. New York: McGraw-Hill. 1968

SALTON, G; MCGILL, M. J. **Introduction to Modern Information Retrieval**. John Wiley and Sons, New York, 1983.

SALTON, G.; WONG, A.; YANG, C. S. A vector space model for automatic indexing. **Communications of the ACM**, 18(11):613-620, 1975.

SAMPAIO, L. O que é o Big Data e qual significado dos 5 V's. **MDP Agência**, 2020. Disponível em: https://mdpagencia.com.br/big-data/. Acesso em: 28 mar. 2020.

SANTANA, R. **Mineração de Textos**: 7 Técnicas e Aplicações para você extrair valor dos dados e alavancar suas análises. Disponível em: http://minerandodados.com.br/index.php/2017/06/22/mineracao-de-textos-7-tecnicas/. Acesso em: dez. 2018.

SANTOS, C. M. DOS. **Classificação de Documentos com Processamento de Linguagem Natural.** 2015. 217 f. Dissertação (Mestrado em Informática e Sistemas) - Departamento de Engenharia Informática e de Sistemas Instituto Superior de Engenharia de Coimbra. Coimbra. Disponível em: http://files.isec.pt/DOCUMENTOS/SERVICOS/BIBLIO/Teses/Tese_Mest_Cedric-Michael-Santos.pdf. Acesso em: 08 mar. 2018.

SANTOS, R. E. S.et al. Técnicas de processamento de linguagem 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 https://revistas.unifacs.br/index.php/rsc/article/view/3030/2498

SARKAR, D. **Text Analytics with Python: A Practical Real-World Approach to Gaining Actionable Insights from your Data**. Apress. 2016.

SARACEVIC, T. Information Science. **JASIS** – Journal of the American Society for Information Science, v. 50, n. 12, p. 1051-1063, 1999.

SEBASTIANI, F. Machine learning in automated text categorisation. **ACM Computing Surveys**, v. 34, n.1), p. 1-47, 2002. Disponível em: http://nmis.isti.cnr.it/sebastiani/Publications/ACMCS02.pdf. Acesso em: 23 fev. 2018.

SHI, G.; KONG, Y. Advances in theories and applications of text mining. Information Science and Engineering (ICISE), 2009 1st International Conference on. **IEEE**, (2009)

SILVA, A. T. **Assistente inteligente para extração de elementos orientados a objeto de discurso**. 2008. Tese (Doutorado em Ciências em Engenharia de Sistemas e Computação) – Departamento de Engenharia da Universidade Federal do Rio de Janeiro - COPPE/UFRJ, Rio de Janeiro.

SILVA, B. C. D. da et al. Introdução ao processamento das línguas naturais e algumas aplicações. ago. 2007.**Série de Relatórios do Núcleo Interinstitucional de Linguística Computacional,** NILC-TR-07-10. Disponível em: Série de Relatórios do Núcleo Interinstitucional de Linguística Computacional. Acesso em: 01 mar. 2019.

SILVA, E.R. C. C. **Técnicas de data e Text Mining para anotação de um arquivo digital.** 2010. Dissertação (Mestrado em Engenharia Eletrônica e Telecomunicações) - Departamento de Electrónica Telecomunicações e Informática. Universidade de Aveiro. Portugal.

SILVA, J.R.C **Processamento de Linguagem Natural** (PLN). Notas de aula Disponível em: https://www.inbot.com.br/artigos/educacional/Processamento-de-Linguagem-Natural-PLN-Jacson-Rodrigues-UFES.pdf. Acesso em: 05 mar. 2018

SIVARAJAH, U., KAMAL, M. M., IRANI, Z., & WEERAKKODY, V. (2017). Critical analysis of Big Data challenges and analytical methods. **Journal of Business Research**, 70, 263–286. https://doi. org/10.1016/j.jbusres.2016.08.001

SILVEIRA, D. D. B. **Modelos de tópicos baseados em autocodificadores variacionais utilizando as distribuições gumbel-softmax e mistura de normais-logísticas.** 2018. Dissertação (Mestrado em Informática) **-** Instituto de Computação. Universidade Federal do Amazonas. Manaus, 2018. Disponível em: https://tede.ufam.edu.br/bitstream/tede/7439/5/Disserta%C3%A7%C3%A3o_DenysSilveiraPPGI. Acesso em: 10 jan. 2020.

SOARES, M. V. B.; PRATI, R.; MONARD, C. WCI 02 Improvements on the Porter's Stemming Algorithm for Portuguese. **Latin America Transactions, IEEE (Revista IEEE America Latina)**, v.7, n.4, p. 472 – 477, ago. 2009.

SOLKA, J. L. (2007). **Text Data Mining**: Theory and Methods. Naval Surface Warfare Center Dahlgren Division Statistics Surveys, Vol. 2 (2008), 94-112.

SOUZA, E. F. **Comparação e escolha de agrupamento: uma proposta utilizando a entropia**. 2007. Dissertação (Mestrado em Ciências) – Instituto de Matemática e Estatística.

Universidade de São Paulo. São Paulo. Disponível em: http://www.teses.usp.br/teses/disponiveis/45/45133/tde-13092007-145328/publico/Dissertacao_Estevao.pdf. Acesso em 10 jan. 2018.

SOUZA, J. G. R. de; OLIVEIRA, A. de Paiva; MOREIRA, A. Development of a brazilian portuguese hotel's reviews corpus. *In*: **International Conference on Computational Processing of the Portuguese Language**, p. 353–361. Springer, 2018.

SOUZA, R. R et al. **O projeto Media Cloud Brasil: Uma análise do tratamento de informações em ambientes de big data**. *In*: Regina de Barros Cianconi, Rosa Inês de Novais Cordeiro, Carlos Henrique Marcondes. (Org.). Gestão do conhecimento, da informação e de documentos em contextos informacionais. 1ed. Niterói: UFF, 2014, v. 1, p. 1-11. Disponível em: https://bibliotecadigital.fgv.br/dspace/bitstream/handle/10438/15036/O_Projeto_Media_Cloud_Brasil.pdf. Acesso em 21 mar. 2018.

SOWA, J. F. Architectures for Intelligent Systems. **Special Issue on Artificial Intelligence of the IBM Systems Journal**, vol. 41, no.3, pp. 331-349, 2002. Disponível em http://www.jfsowa.com/pubs/arch.htm. Acesso em 5 jun. 2017.

SRIVASTAVA, A., SAHAMI, M. **Text Mining**: Classification, Clustering, and Applications: Chapman \& Hall/CRC; 2009. 328 p.

STEIN, B., POTTHAST, M. 2007. Putting Successor Variety Stemming to Work. In **Proceedings of the 30th Annual Conference of the Gesellschaft für Klassifikation**. pp. 367-374. Disponível em: https://www.researchgate.net/publication/221649615_Putting_Successor_Variety_Stemming_to_Work. Acesso em: 16 dez. 2018

STEYVERS, M.; GRIFFITHS, T. **Latent Semantic Analysis**: A Road to Meaning. Laurence Erlbaum, 2007.

SUNDARI, G. J.; SUNDAR, D. (2017). A Study of Various Text Mining Techniques. **International Journal of Advanced Networking & Applications** (IJANA), 08(05), 82-85.

TAN, A.-H. Text mining: the state of the art and the challenges. *In:* WORKSHOP ON KNOWLEDGE DISCOVERY FROM ADVANCED DATABASES, 1999. **Proceedings**... Heidelberg, 1999.p.65-70. (Lecture Notes in Computer Science, 1574).

TICOM, A. A. M. **Aplicação das técnicas de mineração de textos e sistemas especialistas na liquidação de processos trabalhistas**. Dissertação (Mestrado em Ciências em Engenharia Civil) – Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2007. Disponível em: http://www.coc.ufrj.br/pt/documents2/mestrado/2007-1/1592-antonio-alexandre-mello-ticom-mestrado/. Acessado em: 05 abr. 2011.

VARGAS, R. R. **Uma Nova Forma de Calcular os Centros dos Clusters em Algoritmos de Agrupamento Tipo Fuzzy C-Means.** Tese (Doutorado em Sistemas de Computação) – Universidade Federal do Rio Grande do Norte, Natal, 2012. Disponível em: http://www.ppgsc.ufrn.br/~rogerio/publications/tese.pdf. Acesso em: 15 jun. 2019.

VIEIRA, R.; LIMA, V. L. **Linguística computacional: princípios e aplicações**. v. 3, 2001.Jornada de Atualização em Inteligência Artificial. Disponível em: https://www.inf.pucrs.br/linatural/Recursos/jaia-2001.pdf. Acesso em: 03 mar. 2019

XAVIER, B. M.; SILVA, A. D. da; GOMESM G. R. R. Analise comparativa de algoritmos de redução de radicais e sua importância para a mineração de texto. **Operacional para o Desenvolvimento**, v.5, n.1, p.84-99, 2013. Disponível em: https://www.dataci.es.gov.br/publicacoes/arq/189-2122-1-PB.pdf. Acesso em: 09 dez. 2018

YANG, W.; WU, Q.; CHENG, Z. Research on Distributed Text Clustering Based on Frequent Itemset. **Proceedings of the 36th Chinese Control Conference**, China, 2017. Disponível em: http://ieeexplore.ieee.org.ez27.periodicos.capes.gov.br/document/8028263/. Acesso em 23 mar. 2018.

WAEGEL, D. (2006). **The Development of Text-Mining Tools and Algorithms.** Disponível em: http://webpages.ursinus.edu/akontostathis/WaegelHonorsThesis.pdf. Acesso em: 08 mar. 2018.

WEISS, S.; INDUSKHYA, N.; ZHANG, T. and DAMERU, F. **Text Mining**: Predictive Methods for Analyzing Unstructured Information, Springer, New York, NY, 2005

WITTEN, I. H.; FRANK, E. **Data mining: practical machine learning tools and techniques**. 2. ed. San Francisco, California, USA: Elsevier, 2005.

XIONG, C. et al. **An Improved K-means text clustering algorithm By Optimizing initial cluster centers.** 7th International Conference on Cloud Computing and Big Data, 2016. Disponível em: http://ieeexplore.ieee.org.ez27.periodicos.capes.gov.br/stamp/stamp.jsp?tp=&arnumber=7979917. Acesso em 23 mar. 2018

ZAIANE, O., R. **Resource and Knowledge Discovery from the Internet and Multimedia Repositories**.2009. 304 f. Ph.D. thesis, School of Computing Science, Simon Fraser University, Vancouver, BC, Canada. Disponível em: https://www.collectionscanada.gc.ca/obj/s4/f2/dsk1/tape7/PQDD_0025/NQ51940.pdf. Acesso em 08 mar. 2018.

ZANZINI, A. C. da S. **Descritores Quantitativos de Riqueza e Diversidade de Espécies**. (Apostila) - Curso de Pós-Graduação “Lato Sensu” (Especialização) a Distância: Manejo de Florestas Nativas, UFLA/FAEPE, Lavras, 2005. Disponível em: http://www.acszanzini.net/wp-content/uploads/material/livros/Descritores%20Quantitativos%20de%20Riqueza%20e%20Diversidade%20de%20Esp%C3%A9cies.doc. Acesso em: 15 jun. 2019.

ZHAI, C. X.; MASSUNG, S. **Text Data Management and Analysis**: A Practical Introduction to Information Retrieval and Text Mining. Association for Computing Machinery and Morgan, Claypool, New York, NY, USA, 2016.

ZHONG, S. **Efficient online spherical k-means clustering**. *In:* Proceedings of IEEE Int. Joint Conf. Neural Networks (IJCNN 2005), Vol. 5, 2005, pp. 3180–3185. Vol. 5.

ZIPF, G. K. **Selective Studies and the Principle of Relative Frequency in Language**. Harvard University Press, 1932.

ZUCKERMAN, E. **Understanding Media Coverage: Seven Summer-Long Experments with Media Cloud**. (2015). Disponível em: https://civic.mit.edu/blog/ethanz/understanding-media-coverage-media-cloud-experiments. Acesso em: 23 jan. 2018.

ZURINI, M.; SBORA, C**. Clustering Analysis within Text Classification Techniques. Informática Econômica**. v. 15, n. 4, p. 178-189, 2011. Disponível em: http://revistaie.ase.ro/content/60/14%20-%20Zurini,%20Sbora.pdf. Acesso em: 25 nov. 2019.