

## Capítulo

# 2

## Processamento de Linguagem Natural para Identificação de Notícias Falsas em Redes Sociais: Ferramentas, Tendências e Desafios

Nicollas R. de Oliveira (UFF), Pedro Silveira Pisa (Solvimm), Bernardo Costa (Solvimm), Martin Andreoni Lopez (Samsung Research), Igor Monteiro Moraes (UFF), Diogo M. F. Mattos (UFF)

### *Abstract*

*The epidemic spread of fake news is a side effect of the expansion of social networks to circulate news, in contrast to traditional mass media such as newspapers, magazines, radio, and television. Human inefficiency to distinguish between true and false facts exposes fake news as a threat to logical truth, democracy, journalism, and credibility in government institutions. In this chapter, we present methods for preprocessing data in natural language, vectoring, dimensionality reduction, machine learning, and quality assessment of information retrieval. We also present a practical demonstration of identifying fake news will, from the collection and textual processing of social media news to the application of detection algorithms.*

### *Resumo*

*A disseminação epidêmica de notícias falsas (fake news) é um efeito colateral da expansão do uso de redes sociais como meio de circulação de notícias, em contraste às mídias tradicionais de comunicação massiva, como jornal, revista, rádio e televisão. A ineficiência humana para distinção entre fatos verídicos e falsos expõe as notícias falsas como uma ameaça à verdade lógica, à democracia, ao jornalismo e à credibilidade nas instituições de governo. Este capítulo apresenta métodos de pré-processamento de dados em linguagem natural, vetorização, redução de dimensionalidade, aprendizado de máquina e avaliação da qualidade de recuperação de informação. Ao final do capítulo é apresentada uma demonstração prática do processo de identificação de notícias falsas, desde a coleta e processamento textual de notícias de mídias sociais até a aplicação algoritmos de classificação.*

---

Este capítulo foi realizado com recursos do CNPq, CAPES, RNP, FAPERJ, FAPESP (2018/23062-5) e Prefeitura de Niterói/FEC/UFF (Edital PDPA 2020).

## 2.1. Introdução

A veracidade da informação é parte da essencial da sua integridade. O combate às notícias falsas torna indissociáveis os problemas de integridade e veracidade da informação em rede social e do consumo de dados na camada de aplicação. A divulgação de conteúdo falso implica desperdícios de recursos da rede e de processamento e, também, consiste em grave ameaça à integridade das informações e à credibilidade do serviço prestado [de Oliveira et al., 2020]. Dessa forma, o compartilhamento de informações inverídicas diz respeito à qualidade da confiança (*Quality of Trust* - QoT) aplicada à distribuição de notícias [Liu et al., 2010], referindo-se a quanto um usuário confia em um conteúdo de uma determinada fonte.

Em diferentes países, observam-se baixos níveis de confiança nas mídias de massa, *e.g.* apenas 40% nos Estados Unidos<sup>2</sup>, enquanto há altas porcentagens de compartilhamento de *links* nunca lidos (*blindshares*), *e.g.* 59% no Reino Unido [Gabiolkov et al., 2016]. Em 2016, durante as eleições presidenciais dos Estados Unidos, a sociedade americana testemunhou uma epidemia alarmante de notícias falsas, cujos efeitos foram sentidos multilateralmente. Efeito semelhante foi sentido nas eleições de 2018 no Brasil. Devido ao seu potencial de disseminação, aceitação e destruição [Vosoughi et al., 2018], as notícias falsas são atualmente uma das grandes ameaças ao conceito de verdade lógica, deteriorando a democracia, o jornalismo, a justiça e até a economia [Zhou e Zafarani, 2018, Wang, 2017]. Esta última, em especial, teve de lidar com flutuações de 130 bilhões na bolsa de valores, como consequência de uma declaração falsa afirmando que Barack Obama havia se ferido em uma explosão<sup>3</sup>. Nesse sentido, há um crescente esforço conjunto da comunidade acadêmica para desenvolver abordagens capazes de analisar, detectar e intervir na atuação desses conteúdos enganosos. Comprovações científicas já revelaram a vulnerabilidade dos humanos em distinguir verdade e falsidade, sendo reduzida a quase uma probabilidade aleatória, em média 54% de acerto [Zhou e Zafarani, 2018, Wang, 2017, Rubin, 2010, Rubin et al., 2016].

O objetivo deste capítulo é apresentar os principais algoritmos e técnicas que auxiliam na caracterização linguística e detecção de notícias falsas em redes sociais para a garantia da integridade da informação. Este capítulo caracteriza o fenômeno [Rubin et al., 2015a, Rubin et al., 2015b], investiga a propagação em mídias sociais e apresenta as ferramentas e os algoritmos para a detecção de notícias falsas.

O fator chave que impulsiona a criação de notícias falsas é que estas são criadas e publicadas *online*, de maneira mais rápida e barata quando comparadas a veículos tradicionais de mídia como jornais e televisão. Assim, o capítulo evidencia que embora a identificação de notícias falsas possa ser realizada manualmente por profissionais em jornalismo, o foco deste capítulo está na identificação automática através de aparato computacional. A identificação automática pode seguir abordagens distintas, como a prova automática de afirmações lógicas através de fatos já conhecidos, através da análise de propagação das notícias nas redes sociais, através da análise do perfil dos usuários que

---

<sup>2</sup>Disponível em <https://news.gallup.com/poll/185927/americans-trust-media-remains-historical-low.aspx>.

<sup>3</sup>Disponível em <https://www.forbes.com/sites/kenrapoza/2017/02/26/can-fake-news-impact-the-stock-market/#559102f12fac>.

compartilham as notícias ou através do processamento de linguagem natural para a extração de conhecimento em uma abordagem estilístico-computacional [Zhou e Zafarani, 2018]. O escopo do capítulo se limita à abordagem estilístico-computacional baseada em processamento de linguagem natural e justifica-se no fato de que o consumo de dados em redes sociais, por parte de usuários, está restrito à informação que chega ao usuário final. O usuário não tem acesso aos modelos de disseminação de conteúdo ou a modelos de reputação dos usuários que compartilharam os conteúdos consumidos. No capítulo são apresentadas ainda as métricas de qualidade na extração das informações, assim como plataformas computacionais, disponíveis no mercado, que já executam o processamento de linguagem natural como serviço na nuvem.

O restante do capítulo está organizado da seguinte forma. A Seção 2.2 define o fenômeno da propagação de notícias falsas. Os métodos tradicionais de identificação de notícias falsas são discutidos na Seção 2.3. A criação de uma base de dados para identificação correta de notícias falsas é apresentada na Seção 2.4. A Seção 2.5 descreve o processamento de dados em linguagem natural. A Seção 2.6 explica os processos para a transformação de textos em matrizes operáveis computacionalmente, enquanto a Seção 2.7 apresenta as principais ferramentas de aprendizado de máquina usadas sobre dados em linguagem natural. A Seção 2.8 elenca soluções comerciais em nuvens computacionais para tratamento de dados em linguagem natural e a Seção 2.9 descreve iniciativas de pesquisas para a identificação de notícias falsas. Os desafios e oportunidades são discutidos na Seção 2.10. A Seção 2.11 apresenta uma atividade prática para a identificação de notícias falsas. A Seção 2.12 realiza as considerações finais do trabalho.

## 2.2. A Definição de Notícias Falsas

O termo notícias falsas (*fake news*) originalmente faz referência a informações falsas e muitas vezes sensacionais divulgadas sob o disfarce de reportagem. Contudo, o uso desse termo evoluiu e, atualmente, é considerado sinônimo de propagação de informações falsas em mídias sociais [Sharma et al., 2019]. Ressalta-se que, segundo o *Google Trends*, o termo “fake news” alcançou grande popularidade entre os anos de 2017 e 2018, tendo o pico de popularidade em outubro de 2018, quando houve a eleição presidencial no Brasil<sup>4</sup>.

Notícias falsas são definidas como notícias que são intencionalmente e comprovadamente falsas [Zhou e Zafarani, 2018], ou, como quaisquer informações apresentadas como notícias que são factualmente incorretas e projetadas para enganar o consumidor, fazendo-o acreditar que são verdade [Golbeck et al., 2018]. Sharma *et al.* argumentam que essas definições, no entanto, são restritas pelo tipo de informação ou pela intenção de engano e, portanto, não capturaram o escopo amplo do uso atual. Assim, Sharma *et al.* definem o termo como uma notícia ou mensagem publicada e propagada pela mídia, contendo informações falsas, independentemente dos meios e motivos por trás dela [Sharma et al., 2019]. Apesar da inexistência de um consenso claro sobre o conceito de notícias falsas, a definição formal mais aceita interpreta como notícias intencionalmente e verificavelmente falsas. Com relação essa definição, destacam-se dois aspectos: a intenção e a autenticidade. O primeiro aspecto diz respeito à intenção desonesta usada com o intuito

---

<sup>4</sup>Disponível em <https://trends.google.com.br/trends/explore?date=all&geo=BR&q=fake%20news>.

de enganar o leitor. Já o segundo se relaciona com a possibilidade de essas informações falsas terem sua veracidade verificadas.

As notícias falsas podem ser diferenciadas pelos meios empregados para falsificar informações. O conteúdo das notícias pode ser completamente falso, totalmente fabricado para ludibriar o consumidor ou pode ser um conteúdo ardiloso que se utiliza de informações enganosas para abordar um determinado problema. Há também a possibilidade de serem usados conteúdos impostores que simulam fontes genuínas, mas, quando na verdade, as fontes são falsas. Outras características fraudulentas dos conteúdos de notícias falsas são também o uso de conteúdos manipulados, como manchetes e imagens que não estão de acordo com o conteúdo veiculado, ou também a contextualização da notícia com elementos falsos, assim como com conteúdo legítimo, porém em um contexto falso.

As notícias falsas também apresentam motivações ou intenções diversas. São identificadas como motivações para a criação e divulgação de notícias falsas as intenções: de prejudicar ou desacreditar pessoas ou instituições; intenções de lucro para gerar ganhos financeiros aumentando a veiculação e a visualização de publicações *online*; intenções de influenciar e manipular a opinião pública; assim como intenções de promover a discórdia ou, simplesmente, por diversão.

Diversos conceitos concorrem e se sobrepõem ao conceito de notícias falsas. Uma síntese desses múltiplos conceitos, não considerados notícias falsas, pode ser elencada como [Rubin et al., 2015a, Shu et al., 2017, Zhou e Zafarani, 2018, Chen et al., 2015]:

1. **sátiras e paródias**, que pelo conteúdo humorístico embutido, usando sarcasmos e ironias, é factível de ter seu caráter enganoso identificado;
2. **rumores e boatos**, que não se originaram de eventos de notícias, porém são aceitos publicamente;
3. **teorias de conspiração**, por não serem facilmente verificáveis como verdadeiras ou falsas;
4. **spams**, comumente associados a *e-mails* não desejados, os *spams* constituem qualquer campanha publicitária que chega aos leitores por mídia sociais sem que sejam desejadas;
5. **trotes e embustes (*hoaxes*)** que são motivados apenas por diversão ou para enganar indivíduos direcionados;
6. **caça-cliques (*clickbait*)** que pelo fato de empregarem imagens em miniaturas, ou manchetes sensacionalistas, no processo convencimento de usuários a acessarem e compartilharem conteúdos duvidosos, mais se assemelham a um tipo de propaganda falsa;
7. **desinformação (*misinformation*)** que é criada involuntariamente, sem uma origem ou intenção específica de desorientar o leitor;
8. **contra-informação (*disinformation*)** que são informações criadas com intenção específica de confundir o leitor.

As características de cada um desses tipos de conteúdo fraudulento são comparadas às notícias falsas na Tabela 2.1

**Tabela 2.1. Termos e conceitos relacionados à notícias falsas.**

	<b>Autenticidade</b>	<b>Intenção</b>	<b>Forma de Notícia</b>
Sátira e Paródias	Falsa	Não Ruim	Não
Rumores e Boatos	Desconhecida	Desconhecida	Desconhecido
Teorias de Conspiração	Desconhecida	Desconhecida	Não
Spam	Possivelmente Verdadeira	Ruim / Publicitária	Não
Trotes e Embustes	Falsa	Não Ruim	Não
Caça-cliques	Possivelmente Verdadeira	Publicitária	Não
Desinformação	Falsa	Desconhecida	Desconhecido
Contra-informação	Falsa	Ruim	Desconhecido

### 2.2.1. As Características das Notícias Falsas

O crescimento das comunicações mediadas por mídia social é um dos principais fatores que fomentam a mudança de características nas notícias falsas atuais [Sharma et al., 2019]. A incapacidade de um indivíduo de discernir com precisão as notícias falsas das verdadeiras leva ao compartilhamento contínuo e à crença em informações falsas nas redes sociais [Zhou e Zafarani, 2018, Wang, 2017, Rubin, 2010, Rubin et al., 2016]. É difícil para um indivíduo diferenciar entre o que é verdadeiro e o que é falso enquanto é sobrecarregado com informações enganosas que são recebidas por repetidas vezes. Ademais, os indivíduos tendem a confiar em notícias falsas porque há atualmente uma descrença do público em relação às mídias tradicionais e, porque, muitas vezes tais notícias são compartilhadas por amigos ou confirmam um conhecimento prévio. Isso torna a identificação de notícias falsas mais crítica em comparação a outros tipos de informações, já que geralmente são apresentadas com elementos que lhe conferem autenticidade e objetividade, sendo relativamente mais fácil de obter a confiança do público.

A mídia social e o compartilhamento colaborativo de informações em plataformas *online* fomentam também a propagação de notícias falsas, efeito chamado de câmara de eco (*echo chamber effect*) [Shu et al., 2020]. O realismo ingênuo, em que os indivíduos tendem a acreditar mais facilmente nas informações que estão alinhadas a seus pontos de vista, o viés de confirmação, no qual os indivíduos procuram e preferem receber informações que confirmam seus pontos de vista existentes, e teoria da influência normativa, em que os indivíduos escolhem compartilhar e consumir opções socialmente seguras como uma preferência para aceitação e afirmação em um grupo social, são fatores importantes na percepção e compartilhamento de notícias falsas que fomentam o efeito da câmara de eco [Shu et al., 2020]. Esses conceitos implicam a necessidade de os indivíduos buscarem, consumirem e compartilharem informações que estejam alinhadas com suas visões e ideologias. Como consequência, os indivíduos tendem a formar conexões com indivíduos ideologicamente semelhantes e, de forma complementar, os algoritmos de recomendação de redes sociais tendem a personalizar recomendações de conteúdos que atendam às preferências de um indivíduo ou grupo. Esses comportamentos levam à formação de câmaras de eco e bolhas de filtro, em que os indivíduos ficam menos expostos a pontos

de vista conflitantes e ficam isolados em sua própria bolha de informação [Fuller et al., 2009, Sharma et al., 2019]. O confinamento das notícias falsas em câmaras de eco ou bolhas de informação tendem a aumentar as suas sobrevivência e divulgação, pois incorrem no fenômeno da credibilidade social, que sugere que a percepção das pessoas sobre a credibilidade de uma informação aumenta se os outros também a percebem como verdadeira, já que há a tendência de que indivíduos considerem como verdadeira uma informação a que são submetidos repetidas vezes [Rubin et al., 2016].

Os padrões de difusão de notícias falsas nas redes sociais têm sido frequentemente estudados para identificar as características das notícias falsas que auxiliam na discriminação entre notícias falsas e verdadeiras. O problema de identificação de notícias falsas pode ser definido de diversas formas. A classificação pode ser vista como a execução de uma classificação binária entre falsa ou verdadeira, boato ou não boato, farsa ou não farsa. Outra forma de definir o problema é como a execução de uma classificação de várias classes, verdadeira, quase verdadeira, parcialmente verdadeira, principalmente falsa ou falsa, ou ainda como rumor não verificado, rumor verdadeiro, rumor falso ou não é rumor [Sharma e Sharma, 2019]. A principal diferença entre a definição dos problemas de classificação é em função dos diferentes esquemas de anotação ou contextos de aplicativos em conjuntos de dados diferentes. Normalmente, os conjuntos de dados são coletados de declarações anotadas em sites *web* de verificação de fatos, como o “Fato ou Fake”<sup>5</sup> ou a “Agência Lupa”<sup>6</sup>. Esses sites refletem o esquema de rotulagem usado pela organização de verificação de fatos específica.

Sharma *et al.* identificam três características relevantes para a identificação de notícias falsas: as fontes ou promotores da notícia; o conteúdo da informação; e a resposta do usuário ao receber a notícia em redes sociais [Sharma et al., 2019]. A fonte ou os promotores da notícia têm grande influência na classificação da veracidade da notícia. Contudo, Sharma *et al.* ressaltam que as listas de fontes possíveis de notícias falsas não são exaustivas e que os domínios usados para a divulgação de uma notícia podem ser falsificados [Sharma et al., 2019]. Outro ponto a ser ressaltado é que *bots*, contas falsas ou comprometidas controladas por humanos ou programas para apresentar e promover informações nas redes sociais, são responsáveis por acelerar a velocidade de propagação de informações verdadeiras e falsas de forma quase igual, a fim de alavancar a credibilidade e a reputação das contas de *bots* [Davis et al., 2016]. A segunda característica importante é o conteúdo da informação propagada. O conteúdo da informação é uma das principais características a ser analisada para classificar a notícia como verdadeira ou falsa. Oliveira *et al.* identificam que notícias falsas e notícias reais veiculadas no Brasil têm um comportamento estatisticamente diferente no somatório da frequência relativa das palavras usadas no conteúdo. Notícias falsas tendem a usar menos palavras relevantes do que notícias reais [de Oliveira et al., 2020]. Outras características textuais incluem o uso de palavras sociais, auto-referências, declarações de negação, reclamações e itens generalizantes, além de que há uma tendência de que notícias falsas apresentem menor complexidade cognitiva, menos palavras exclusivas, mais palavras de emoção negativa e mais palavras de ação [Sharma et al., 2019]. Por fim, as respostas do usuário nas redes sociais fornecem informações auxiliares para a detecção de notícias falsas. A resposta

---

<sup>5</sup>Disponível em <https://g1.globo.com/fato-ou-fake/>.

<sup>6</sup>Disponível em <https://piaui.folha.uol.com.br/lupa/>.

dos usuários é importante para a identificação, pois, somada aos padrões de propagação, são mais difíceis de serem manipuladas do que o conteúdo da informação e, por vezes, as respostas dos usuários contêm informações óbvias sobre a veracidade [Zhou e Zafarani, 2018]. O engajamento dos usuários, nas formas de curtidas, compartilhamentos, respostas ou comentários, contêm informações capturadas na estrutura de árvores de propagação que indicam o caminho do fluxo de informações, informações temporais em carimbos de data e hora, informações textuais em comentários do usuário e informações de perfil do usuário envolvido no engajamento [Sharma et al., 2019].

A caracterização da fonte ou da propagação, do conteúdo e da reposta do usuário permitem definir diferentes técnicas de identificação de notícias falsas, tais como, identificação baseada em retroalimentação pelo padrão de propagação, identificação no processamento de linguagem natural aplicado ao conteúdo de mensagens e aplicação de mecanismos de aprendizado de máquina e, por fim, identificação baseada em intervenção dos usuários. Este capítulo foca nas soluções baseadas na análise do conteúdo das notícias.

### **2.2.2. O Processo de Disseminação de Notícias Falsas**

Diversas entidades, indivíduos e organizações interagem na divulgação, moderação e consumo de notícias falsas nas redes sociais. Devido à pluralidade de atores envolvidos, o problema de identificação e mitigação da disseminação de notícias falsas torna-se ainda mais complexo. A divulgação das notícias falsas é fortemente baseada em mídias sociais em detrimento de mídias tradicionais de jornalismo, devido à grande escala, ao alcance das mídias sociais e à capacidade de compartilhar colaborativamente conteúdo. Os sítios *web* de mídias sociais têm se tornado a forma mais popular de disseminação, devido à crescente facilidade de acesso e popularização da comunicação mediada por computador e do acesso à Internet [Mattos et al., 2019]. Paralelamente, enquanto nas mídias tradicionais de jornalismo a responsabilidade pela criação do conteúdo cabe ao jornalista e à organização redatora, a moderação nas redes sociais varia bastante. Cada mídia social está sujeita a diferentes regras de moderação e regulamentação de conteúdo. A informação é consumida principalmente pelo público em geral ou pela sociedade, que constituem número crescente de usuários de mídia social. O crescimento no consumo de informação por meio de mídia social aumenta o risco de notícias falsas causarem danos generalizados [Sharma et al., 2019].

Sharma *et al.* destacam três atores distintos na propagação das notícias falsas: o adversário, o verificador de fatos e o usuário susceptível [Sharma et al., 2019]. Os adversários são indivíduos ou organizações mal-intencionados que muitas vezes se passam por usuários comuns de redes sociais usando *bots* [Davis et al., 2016] ou contas reais. Os adversários podem tanto agir como fonte ou como promotores de notícias falsas. Essas contas também agem em grupo propagando conjuntos de notícias falsas. O verificador de fatos consiste em um conjunto de várias organizações de verificação de fatos, como “Fato ou Fake” e a “Agência Lupa”, que buscam expor ou confirmar notícias que gerem dúvidas sobre a sua veracidade. Muitas das vezes, as verificações se baseiam no jornalismo de checagem de fatos que depende da verificação humana. Contudo, há soluções tecnológicas automatizadas que visam a detecção de notícias falsas para empresas e consumidores. Essas soluções atribuem pontuações de credibilidade a conteúdo da *web* usando inteligên-

cia artificial. Por fim, o usuário susceptível consiste no usuário de rede social que recebe o conteúdo duvidoso, porém não é capaz de distinguir entre uma notícia falsa ou verdadeira e, assim, acaba propagando a notícia falsa em sua rede social, mesmo que não tenha a intenção de contribuir para a proliferação de conteúdo fraudulento.

### 2.3. Os Métodos Tradicionais de Detecção de Notícias Falsas

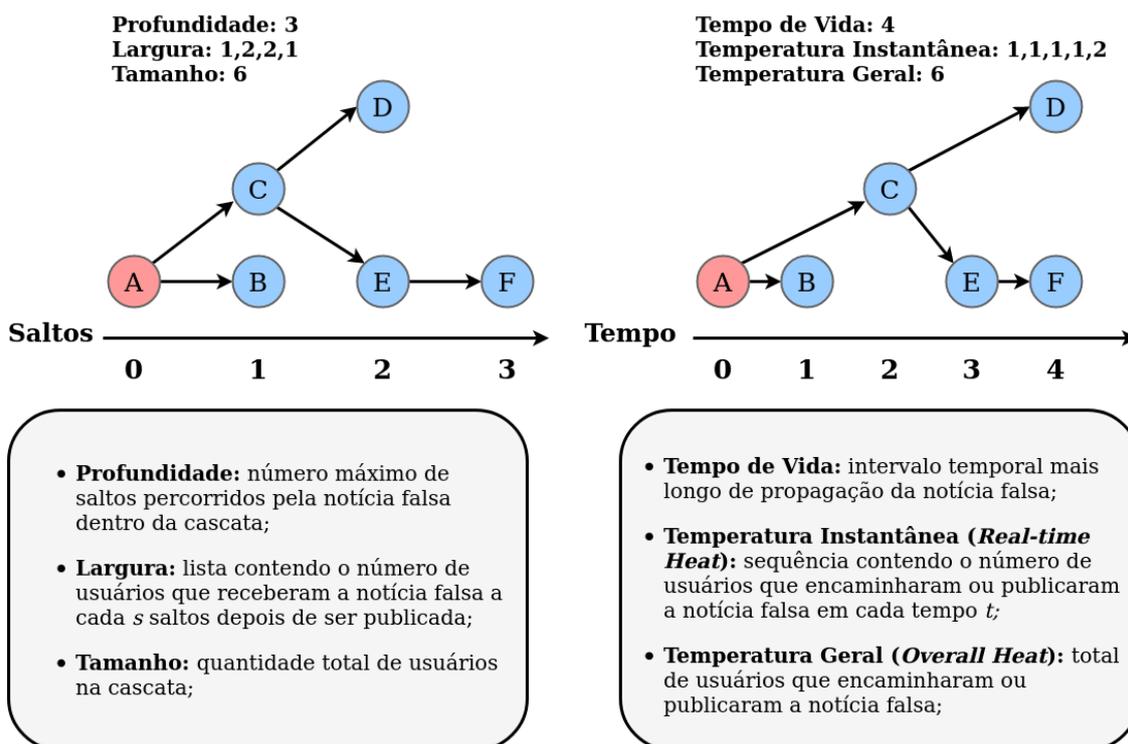
A identificação de notícias falsas pode ser realizada por meios manuais, através de profissionais em jornalismo, sendo a abordagem mais utilizada normalmente. Contudo, tal abordagem não é compatível com o volume atual de criação e disseminação de conteúdo nas redes sociais. Para contrapor esse problema de escalabilidade, métodos automáticos geralmente integram técnicas de Recuperação de Informação, Processamento de Linguagem Natural (PLN) e Aprendizado de Máquina no processo de verificação da veracidade de notícias veiculadas na Internet.

A respeito de métodos automáticos de detecção de notícias falsas, distinções podem ser observadas ao discretizar as formas de detecção por foco de atuação. Na literatura são vislumbradas três teorias analíticas preponderantes e potencialmente úteis na contenção de notícias falsas. A primeira teoria é fundamentada em uma **análise baseada na propagação**, cujo foco está no mapeamento qualitativo ou quantitativo do espalhamento das notícias falsas em rede sociais, a partir de padrões empíricos ou modelagem matemática, respectivamente. A base de ambos os mapeamentos é a cascata de notícias falsas, uma estrutura em árvore que representa todo o processo de disseminação de notícias falsas, podendo ser pautada tanto em uma perspectiva por saltos ou por tempo. A Figura 2.1 retrata as perspectivas de representação da propagação citadas.

Um desses padrões de propagação foi mapeado por Kwon *et al.*, cujo estudo revelou uma tendência das notícias não-confirmadas exibirem múltiplos e periódicos picos de discussão ao longo do dia no *Twitter*, enquanto que notícias confirmadas apresentavam apenas um pico proeminente [Kwon et al., 2013]. Adicionalmente, os estudos de Zhou *et al.* e Vosoughi *et al.* alertaram sobre a capacidade das notícias falsas, sobretudo do âmbito político, se espalharem de maneira mais rápida, mais abrangente e com mais abrangência do que notícias verdadeiras. Tal conclusão foi embasada no comportamento da representação em cascata das notícias falsas, marcado por uma maior largura máxima, profundidade e tamanho, alcançados em menos tempo que a representação em cascata de notícias legítimas [Zhou et al., 2015, Vosoughi et al., 2018].

Embora útil, a descoberta de padrões empíricos de propagação característicos de cada tipo de notícia é uma estratégia com resultados temporários, pois a alta dinamicidade e variabilidade de comportamento das notícias falsas. É conveniente a aplicação em conjunto com uma modelagem matemática. Em geral, essa modelagem recorre a uma análise regressiva usando modelos clássicos como o epidêmico e o econômico.

A construção matemática da difusão de notícias falsas através de uma modelagem epidêmica visa principalmente a predição do número de disseminadores (temperatura geral). Essa estratégia de modelagem inicia com uma etapa que associa cada usuário a um dentre três estados: (i) disseminadores; (ii) potenciais disseminadores; e (iii) disseminadores arrependidos, aqueles que após encaminharem ou publicarem uma notícia falsa a apagam. Nesta etapa, há também a definição inicial das taxas de transição entre es-



**Figura 2.1. Ilustração das cascatas de notícias falsas, tanto numa perspectiva baseada em saltos (à esquerda) quanto baseada em tempo (à direita). O nó-raiz A, em ambas as perspectivas, representa o primeiro usuário a publicar ou criar a notícia falsa e os demais nós representam usuários atuantes no encaminhamento ou compartilhamento do conteúdo falso.**

ses estados. A próxima etapa consiste na construção do modelo, a qual pode considerar fenômenos como o efeito *backfire*<sup>7</sup> e o reflexo de Semmelweis<sup>8</sup>, que revelam a rejeição de indivíduos às ideias contrárias às suas. A terceira etapa consiste na determinação das taxas reais de transição entre estados [Zhou e Zafarani, 2018].

A modelagem econômica introduz uma abordagem racional sobre interações de notícias falsas, que tenta capturar e prever o comportamento dos indivíduos ao serem expostos a uma notícia falsa. Neste tipo de modelagem, o ciclo de geração e consumo de notícias é visto como um jogo de estratégia entre dois jogadores, os publicadores e os consumidores. A cada jogador, a decisão de encaminhar ou deletar uma notícia falsa implica pares de vantagens específicas e excludentes entre si. Aos publicadores, cabe a escolha entre obter uma vantagem de curto prazo ( $g_p$ ), que maximiza o lucro relacionado ao número de consumidores alcançados, ou uma vantagem de longo prazo ( $b_p$ ), que privilegia sua reputação, tornando-os uma fonte autêntica de notícias. Já para os consumidores, as consequências dessa decisão dual é dividida entre uma vantagem de informação ( $g_c$ ), que permite a obtenção de informação verdadeira e não enviesada, ou uma vantagem psicológica ( $b_c$ ), ligada à teoria de viés confirmatório que reflete sua preferência por receber notícias que satisfazem opiniões prévias e necessidades sociais. Dessa forma,

<sup>7</sup>Relacionado ao fato de indivíduos rejeitarem mais fortemente evidências opostas às suas crenças.

<sup>8</sup>Remete a tendência dos indivíduos rejeitarem novas evidências por estas contradizerem suas normas e crenças estabelecidas.

quando  $g_p > b_p$  e  $g_c > b_c$  constrói-se uma cadeia propícia para o espalhamento de notícias falsas [Shu et al., 2017].

Paralelamente, existe a **análise baseada no usuário**, que considera o papel deste na disseminação das notícias, conseqüentemente distinguindo um usuário malicioso daqueles sem má intenção, os ingênuos. Sejam motivados por benefícios monetários ou não monetários, a atuação de usuários maliciosos nas redes sociais se dá através de contas que escondem a real identidade do gerenciador. Ao analisar o nível de participação de humanos no processo de gerenciamento dessas contas, pode-se dividi-las em três categorias: *social bots*, *cyborgs* e *trolls*. Todas essas contas maliciosas altamente ativas e partidárias têm um único propósito de tornarem-se fontes poderosas de proliferação de notícias falsas. Em um nível baixo de dependência humana, os *social bots* são contas controladas por um algoritmo de computador, cujo objetivo é produzir conteúdo automaticamente e interagir com humanos ou outros *bots*. Já em nível intermediário, os *cyborgs* são contas que alternam entre atividades automatizadas e humanas. Normalmente, este tipo de conta maliciosa é registrada por um usuário humano, fornecendo assim uma camuflagem para definir programas automatizados para realizar atividades nas redes sociais. No nível mais elevado de dependência, os *trolls* são contas totalmente mantidas por usuários humanos reais que visam perturbar comunidades *online* e provocar uma resposta emocional dos consumidores [Shu et al., 2017].

Outros trabalhos, como Barreto *et al.*, propõem uma metodologia capaz de distinguir usuários legítimos e *spammers* considerando a 2-vizinhança no *Twitter*. A proposta é subdividida em três etapas, cuja primeira é a pré-seleção manual de possíveis usuários. Como critério de pré-seleção de um usuário malicioso utiliza-se o fato do usuário enviar mensagens contendo pelo menos um tópico popular. A segunda etapa inclui a coleta dos dados da rede no entorno dos usuários pré-selecionados. Como última etapa, é feita uma análise desses dados avaliando métricas como distribuição de grau, centralidade de grau, coeficiente de agrupamento e *PageRank*. Ao final, os autores relatam um comportamento diferenciado da distribuição de grau dos *spammers*, contrariando a lei de potência esperada para os usuários legítimos [Barreto et al., 2014].

Mesmo não intencionalmente, usuários comuns são igualmente susceptíveis a se tornarem propagadores de notícias falsas. Além da baixa capacidade de detecção de notícias falsas, usuários normais são influenciados por fatores psicológicos e sociais. Na psicologia, esses fatores são identificados como vulnerabilidades individuais cujo um dos exemplos conhecidos é o realismo ingênuo. Esta vulnerabilidade formula uma tendência dos usuários em acreditar que suas percepções da realidade são os únicos pontos de vista, enquanto as demais são consideradas desinformadas, irracionais ou tendenciosas. Considerando o campo social, a disseminação das notícias falsas está intimamente conectada à dinâmica social dos indivíduos, estando correlacionada à três teorias: (i) a Teoria da Prospecção, que descreve a tomada de decisão como um processo pelo qual os indivíduos fazem escolhas com base nos ganhos e perdas relativas em comparação com seu estado atual; (ii) a Teoria da Identidade Social, que associa o autoconceito dos indivíduos é derivado a partir da percepção de pertencimento a um grupo social relevante; (iii) a Teoria da Influência Normativa, na qual enfatiza que aceitação e afirmação social são essenciais para a identidade e autoestima de um indivíduo, fazendo com que os usuários escolham ser “socialmente seguros” [Shu et al., 2017].

Embora a existência de notícias falsas preceda o surgimento das mídias sociais, seu advento alterou e ampliou a dinâmica de propagação desse tipo de informação, inclusive adicionando novos atores. Outro fator atual que facilita a disseminação desse tipo de notícia é o fenômeno de bolha social ou câmara de eco (*echo chamber*) em que usuários tendem a se relacionar virtualmente com seus *like-minders*, ou seja, pessoas que pensam como eles. Nessas bolhas sociais estão presentes duas ideias principais, sendo a primeira conhecida como credibilidade social. Tal ideia é explicada pelo fato de as pessoas serem mais propensas a considerar uma fonte como credível se os outros também a considerarem, especialmente quando não há como se comprovar. A segunda ideia remete a uma heurística de frequência, segundo a qual consumidores naturalmente preferem notícias que são ouvidas mais constantemente, mesmo sendo falsas [Shu et al., 2017].

Uma terceira teoria analítica remete à **análise baseada no estilo** da escrita, cujo foco de atuação principal está no conteúdo da notícia, ou seja, no texto propriamente dito. Essa análise parte da premissa de que notícias falsas detêm perfis de escrita únicos, diferentes dos seus pares legítimos. Cabe então aos métodos de detecção alinhados com essa teoria aplicar técnicas para extração de características linguísticas.

Dentre os estudos relacionados a essa abordagem estilística, destaca-se o apresentado por Rashkin *et al.*, que trabalha sob a hipótese de que as notícias falsas tendem a conter uma narrativa mais interessante a fim de atrair leitores [Rashkin et al., 2017]. Assim, utilizando um *corpus* composto por artigos de notícias de diversas intenções, fontes e graus discretos de veracidade, o método empregado prevê a extração de características léxicas latentes. A análise dessas características permitiu formular perfis distintos de notícias dependendo da sua fonte de veiculação. Assim, constata-se que notícias oriundas de fontes confiáveis normalmente apresentam alguma forma de embasamento concreto, como comparações numéricas e expressões relativas a dinheiro. Em um sentido oposto, notícias de fontes menos confiáveis detinham uma incidência maior de pronomes de primeira e segunda pessoa, superlativos, advérbios de modo e palavras que expressam hesitação (*hedging words*). A análise baseada no estilo é a explorada a seguir.

## 2.4. A Construção da Base de Dados

A caracterização do problema de identificação de notícias como um problema de classificação implica a construção de uma base de dados adequada. A construção de uma base de dados com qualidade e disponibilidade é o pilar de qualquer mecanismo automático de detecção de notícias falsas. Sua importância está atrelada à necessidade de armazenar a máxima quantidade de exemplos contrastantes, notícias falsas e verdadeiras, para então serem absorvidos por algoritmos de aprendizagem de máquina [Oshikawa et al., 2018]. A Tabela 2.4 traz uma compilação de bases de dados de notícias falsas disponíveis, tanto na língua inglesa quando em na língua portuguesa.

Nesse contexto, uma eventual coleta errônea de dados tem o potencial de causar inúmeras consequências negativas, que variam desde a particularização da análise até a obtenção de resultados dissonantes. Logo, é prudente adotar algumas diretrizes sugeridas por Rubin *et al.* para a formação de um *corpus* de notícias falsas [Rubin et al., 2015a]. Rubin *et al.* defendem que qualquer construção de uma base de dados, *corpus*, de notícias falsas deve se ater a nove condições importantes, elencadas a seguir. (i) Considerar

**Tabela 2.2. Base de dados de notícias falsas disponíveis**

	Conteúdo	Quant.	Rotulagem	Anotador
<b>Buzzface</b> [Santia e Williams, 2018]	Postagens e comentários de redes sociais (Facebook)	2263	Granular em quatro níveis (predominantemente verdade, predominantemente falso, mistura de verdadeiro e falso e nenhum conteúdo factual)	Previamente checado por agências de notícias (Buzzfeed)
<b>FAKENEWSNET</b> [Shu et al., 2020]	Artigos inteiros	23921	Binária (verdadeiro ou falso)	Previamente checado por agências de notícias (PolitiFact e GossipCop)
<b>Fake.Br Corpus</b> [Monteiro et al., 2018]	Artigos inteiros	7200	Binária (verdadeiro ou falso)	Considera a credibilidade da fonte
<b>LIAR</b> [Wang, 2017]	Declarações curtas (políticas)	12,8k	Granular em seis níveis (verdade, predominantemente verdade, meia-verdade, quase verdade, falso, <i>pants-fire</i> )	Previamente checado por agências de notícias (PolitiFact)
<b>Emergent</b> [Ferreira e Vlachos, 2016]	Declarações e títulos relacionados	300	Binária (verdadeiro ou falso)	Equipe jornalística
<b>FEVER</b> [Thorne et al., 2018]	Declarações curtas (Wikipedia)	185k	Granular em três níveis (suportada, refutada e sem informação suficiente)	Anotadores humanos treinados
<b>CREDBANK</b> [Mitra e Gilbert, 2015]	Postagens de redes sociais (Twitter)	60M	Vetor com 30 dimensões contendo pontuações variáveis em cinco níveis de veracidade	Crowd-sourcing
<b>BuzzfeedNews</b>	Postagens de redes sociais (Facebook)	2282	Granular em quatro níveis	Equipe jornalística
<b>BuzzFeed-Webis</b> [Potthast et al., 2017]	Postagens de redes sociais (Facebook)	1687	Granular em quatro níveis	Previamente checado por agências de notícias (Buzzfeed)
<b>PHEME</b> [Zubiaga et al., 2016]	Postagens de redes sociais (Twitter)	330	Binária (verdadeiro ou falso)	Jornalistas e crowd-sourcing

tanto as instâncias falsas como as verdadeiras permite que eventuais métodos preditivos aplicados à base considerem padrões característicos de cada tipo de notícia. (ii) A informação deve estar preferencialmente em formato textual, em vez de ser apresentada como mídia em formato de áudio ou vídeo. Informações nesses formatos devem ser transcritas, tornando-se manipuláveis por ferramentas de processamento de linguagem natural. (iii) A homogeneidade das notícias quanto ao tamanho e (iv) quanto a maneira da escrita, são outras duas condições a serem consideradas, evitando sempre que possível instâncias muito díspares. Igualmente, existe uma preocupação com (v) a forma de distribuição das notícias, visto que há suspeitas de que ao saber como e em qual contexto estas foram fornecidas, *e.g.* humorístico, sensacionalista, pode-se influenciar os leitores. Além disso,

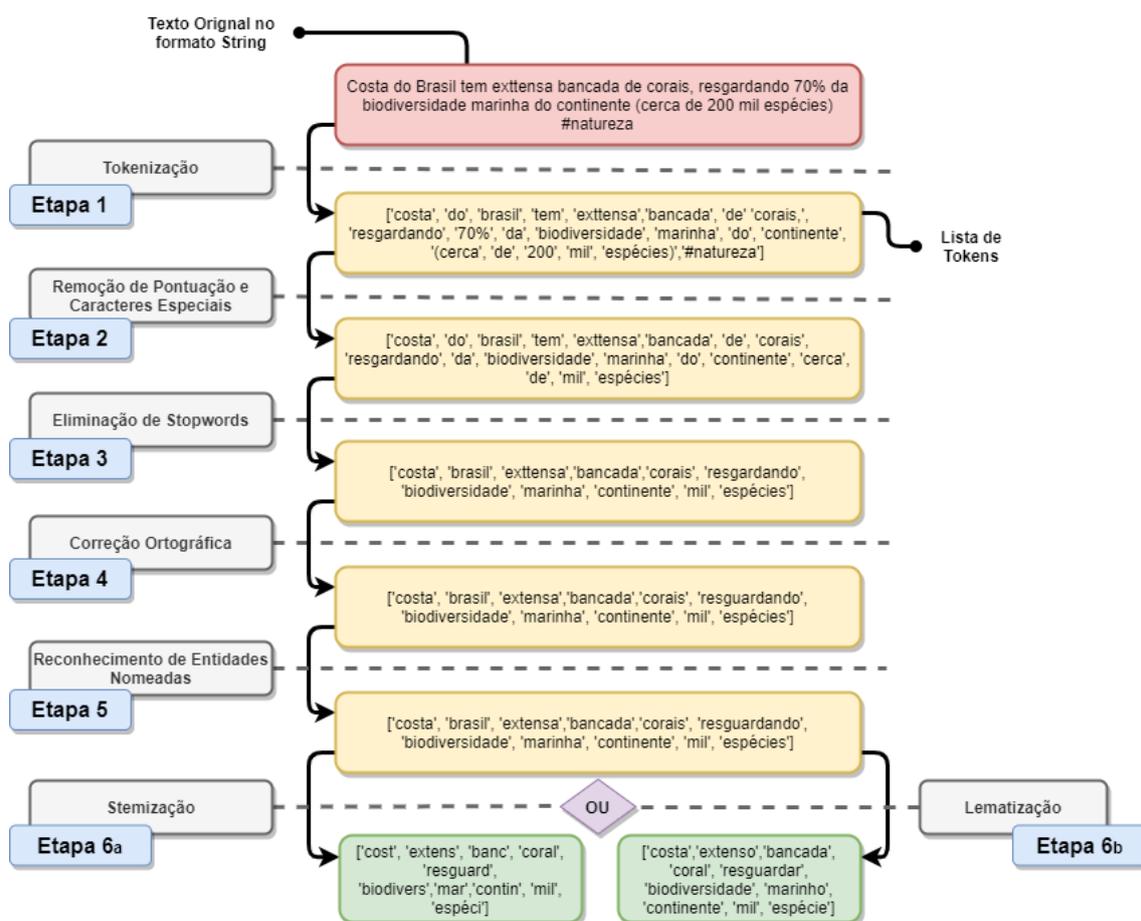
(vi) a aquisição de notícias de um mesmo intervalo temporal é um fator primordial, pois os assuntos podem variar drasticamente em um curto intervalo de tempo. Adicionalmente, (vii) é aconselhável atender alguns aspectos pragmáticos, tais como custos com direito autoral, disponibilidade, facilidade de obtenção e privacidade dos escritores. Não se deve negligenciar o (viii) idioma e a (ix) cultura a que pertencem os dados coletados, pois a tradução pode implicar ambiguidades ou más interpretações, afetando negativamente a eficiência de processos de detecção [Rubin et al., 2015a, Rubin, 2014].

## 2.5. Processamento de Linguagem Natural

O Processamento de Linguagem Natural (PLN), também conhecido como linguística computacional, consolida-se como um campo de pesquisa que envolve modelos e processos computacionais para a solução de problemas práticos de compreensão e manipulação de linguagens humanas. Independentemente de sua forma de manifestação, textual ou fala, a linguagem natural é entendida como qualquer forma de comunicação diária entre humanos. Tal definição exclui linguagens de programação e notações matemáticas, consideradas linguagens artificiais. As linguagens naturais estão em constante mudança, dificultando o estabelecimento de regras explícitas para computadores [Clark et al., 2012, Otter et al., 2020, Bird et al., 2009].

Em uma decomposição refinada, o PLN pode ser dividido em cinco estágios primários de análise, que, quando realizados, permitem que o significado pretendido pelo autor seja extraído computacionalmente de um documento textual. Embora seja mais condizente com um estágio de pré-processamento, o primeiro estágio é a segmentação por *tokenização*. A *tokenização* é uma técnica obrigatória dado que os documentos textuais em linguagem natural geralmente são compostos de frases longas, complicadas e mal formadas. A etapa seguinte é a análise léxica, que visa relacionar as variantes morfológicas aos seus *lemas*, ou seja, a forma primitiva das palavras do dicionário. A análise sintática foca no 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. Linguisticamente, a análise semântica tenta destilar o significado de palavras, expressões fixas, sentenças inteiras, sendo assim frequentemente aplicada na resolução de ambiguidades. Por fim, a análise pragmática busca compreender uma determinada frase, observando referências pronominais e a coerência textual da estrutura das frases adjacentes. Embora o PLN possa introduzir outros estágios de análise, como reconhecimento de emoção, esses cinco estágios básicos são suficientes para extrair a informação semântica contextualizada de um documento de linguagem natural [Indurkha e Damerau, 2010].

Limitando o processamento até o estágio de análise morfológica, é possível compor uma sequência básica de técnicas de PLN para garantir a identificação, e posterior remoção, de qualquer ruído textual que possa comprometer a extração e interpretação inteligente das informações contidas em cada sentença. Nesta sequência, ilustrada na Figura 2.2, são aplicadas técnicas de limpeza e conformação dos dados incluindo *tokenização*, remoção de pontuação e caracteres especiais, eliminação de *stopwords*, correção ortográfica, reconhecimento de entidades nomeadas e *stemização* ou *lematização*. Guiada pela ordem acima mencionada, cada *sentença* do texto original é primeiramente submetida a um procedimento de discretização visto na Etapa 1, conhecido como *tokenização*. Usando neste caso o caractere de espaço como critério delimitador, a *tokenização* trans-



**Figura 2.2. Aplicação do processamento de linguagem natural em um texto bruto. A tokenização segmenta o texto contíguo em um conjunto de *tokens*. Elementos de pouca relevância semântica são removidos, assim como pontuações, caracteres especiais e *stopwords*. Entidades nomeadas são identificadas e removidas. *Stemização* ou *lematização* reduzem a diversidade de *tokens*.**

forma cada sentença contígua em uma lista de *tokens*, permitindo o manuseio individualizado dos *tokens*. Basicamente cada *token* é visto como uma instância de uma sequência de caracteres. Posteriormente na Etapa 2, recursos ortográficos como pontuação, *e.g.* pontos final, de exclamação e de interrogação, e caracteres especiais, *e.g.* números, cifrão e asterisco, são removidos de cada *token*.

Na Etapa 3, eliminam-se as *stopwords*, ou palavras mais frequentes, como conectivos, artigos e pronomes. Essa tarefa em especial tem como base o princípio de que quanto maior a frequência de uma palavra no *corpus*, menos informação relevante a palavra possui. Em seguida na Etapa 4, ocorre a correção ortográfica através da comparação do *token* com seu correspondente mais próximo no dicionário. Executa-se tal procedimento calculando a distância Levenshtein, *i.e.*, o número mínimo de operações necessárias para transformar um nome no banco de dados em outro contido em um dicionário de nomes. O reconhecimento de entidades nomeadas, Etapa 5, identifica principalmente nomes próprios, com subsequente remoção dessas palavras. Na *stemização*, as palavras flexionadas ou derivadas são reduzidas ao seu radical, eliminando possíveis variantes ou

**Tabela 2.3. Características usadas em cada abordagem de detecção de notícias falsas baseadas em processamento de linguagem natural.**

		[Zhou et al., 2004]	[Fuller et al., 2009]	[Afroz et al., 2012]	[Hauch et al., 2015]	[Monteiro et al., 2018]	[Rashkin et al., 2017]	[Rubin et al., 2016]
<b>Tipo</b>	<b>Atributos</b>							
Quantidade	Contagem de caracteres ou <i>tokens</i>	x		x		x		
	Contagem de palavras		x	x	x	x		
	Contagem de sentenças	x	x	x	x	x		
	Contagem de verbos	x	x		x		x	
	Contagem de frases nominais <sup>1</sup>	x						
	Contagem de substantivos					x		
	Contagem de <i>stopwords</i>					x		x
	Contagem de adjetivos					x		x
	Contagem de modificadores <sup>2</sup>	x	x	x		x	x	x
Informalidade	Taxa de erros tipográficos	x			x	x		
Complexidade	Média de caracteres por palavra	x	x	x	x			
	Média de palavras por sentença	x	x	x	x	x		
	Média de orações por sentença	x						
	Média de pontuações por sentença	x	x	x		x		
Incerteza	% de verbos modais	x	x	x	x	x	x	
	% de termos que indicam certeza <sup>3</sup>	x	x	x	x		x	
	% termos que indicam generalização		x		x		x	
	% termos que indicam tendência		x	x	x			
	% de números e quantificadores <sup>4</sup>			x	x		x	
	# de pontos de interrogação			x				
Não Imediação	% de voz passiva	x	x		x	x		
	Pronomes na 1ª pessoa do singular	x	x	x	x	x	x	x
	Pronomes na 1ª pessoa do plural	x	x	x	x	x	x	x
	Pronomes na 2ª ou 3ª pessoa do plural	x	x	x	x	x		x
Diversidade	Diversidade Léxica: % palavras únicas	x	x	x	x			
	Redundância: % de <i>function words</i> <sup>5</sup>	x	x	x	x			
	% de <i>content words</i> <sup>6</sup>	x	x		x			
	Entidades nomeadas aleatórias <sup>7</sup>							x
Sentimento	% de palavras positivas	x	x	x	x	x		
	% de palavras negativas	x	x	x	x	x		x
	# de pontos de exclamação			x				
	Teor humorístico/sarcástico							x

<sup>1</sup>frases cujos núcleos são substantivos. <sup>2</sup>adjetivos e advérbios. <sup>3</sup>e.g. “nunca”, “sempre”. <sup>4</sup>advérbios de intensidade. <sup>5</sup>palavras com pouco significado atrelado, usadas para expressar relações gramaticais entre palavras ou especificar a atitude ou o humor do falante, e.g., preposições, pronomes, verbos auxiliares, conjunções e artigos. <sup>6</sup>palavras que contém um conteúdo semântico, e.g., substantivos, verbos, adjetivos e a maioria dos advérbios. <sup>7</sup>presença de nomes próprios, nunca antes citados no texto, na última sentença.

plurais. Por fim, com o objetivo de reduzir o processamento desnecessário causado por eventuais redundâncias entre as palavras, seja por flexões ou derivações, é comum a adoção da Etapa 6a ou 6b, sendo respetivamente a *lematização* e a *stemização*. Na tarefa de *lematização*, procura-se eliminar as possíveis variantes ou plurais de uma mesma palavra, reduzindo-as ao mesmo lemas, conhecidos como forma de dicionário. Em contrapartida,

na *stemização* esta redução é feita transformando cada palavra no seu radical [de Oliveira et al., 2020, Navigli, 2009, Manning e Schutze, 1999].

Expandindo o processamento textual a outros estágios linguísticos, há técnicas de PLN que desempenham a tarefa de análise sintática em diferentes graus de complexidade. Em um nível básico, a POS (*part-of-speech*) *tagging* caracteriza-se como uma técnica que retorna apenas a camada mais inferior da árvore de análise, ou seja, a marcação gramatical. Assim, cada a palavra de sentença é atribuído um meta-dado, identificando sua classe gramatical e conjugação. Em um nível intermediário, a técnica *chunking*, também chamada de análise superficial, é uma técnica que analisa frases inteiras, primeiro identificando as partes constituintes das frases (substantivos, verbos, adjetivos) e, em seguida, ligando-as a unidades de ordem superior com significado gramatical discreto. Através dessa técnica, é selecionar estruturas sintáticas específicas como frases nominais, verbais [Manning et al., 2014].

A análise de sentimento, ou mineração de opinião, inspeciona o texto fornecido e identifica a atitude ou emoção dominante no texto através de um grau de polaridade, classificando-o como positivo, negativo ou neutro. Outra propriedade comumente associada à análise de sentimento é a subjetividade, que permite diferenciar frases com alta acidência de opinião, julgamento ou emoção das frases com informações factuais. Normalmente a classificação do sentimento de frases funciona considerando as palavras isoladamente, atribuindo pontos positivos para palavras positivas e pontos negativos para palavras negativas e, em seguida, resumindo esses pontos. A simplicidade dessa lógica resulta em um desprezo pela ordem das palavras implicando perdas semânticas relevantes [Socher et al., 2013]. Modelos online atuais consideram a estrutura da sentença e constroem a representação de sentenças inteiras. Assim, esses modelos calculam o sentimento baseados em como as palavras da sentença compõem o significado de frases longas.

Atualmente, dentre as ferramentas mais poderosas de extração de conhecimento sobre textos, a Stanford CoreNLP <sup>9</sup> e a NLTK <sup>10</sup> são as mais conhecidas. Outras ferramentas como a Consulta Linguística e Contagem de Palavras, *Linguistic Inquiry and Word Count - LIWC* [Pennebaker et al., 2001] destaca-se como um software de análise textual capaz de analisar e quantificar os componentes emocionais, cognitivos e estruturais presentes nos textos. A capacidade do LIWC revelar características latentes de um texto é intimamente dependente do idioma do dicionário de palavras associado ao software. Embora originalmente otimizado para a língua inglesa, atualmente o dicionário LIWC foi traduzido para a língua portuguesa [Balage Filho et al., 2013]. Essas ferramentas são igualmente úteis na extração de características como as vistas na Tabela 2.3.

## 2.6. Representação Vetorial de Textos

Mesmo devidamente padronizada, cada sentença não é passível de ser operada matematicamente, visto que ainda é composta por radicais de palavras e não por valores mensuráveis. Destaca-se que até este momento, as operações realizadas sobre os dados são realizadas em cadeias de caracteres. No entanto, para o cálculo de modelos de aprendizado de máquina são necessários dados que possam ser operados matematicamente.

---

<sup>9</sup>Disponível em <https://stanfordnlp.github.io/CoreNLP/>.

<sup>10</sup>Disponível em <https://www.nltk.org/>.

Para obter uma representação numérica, emprega-se o Modelo de Espaço Vetorial. Esse modelo define que textos, sejam sentenças ou documentos, podem ser interpretados como um espaço vetorial de palavras, em que cada palavra pode ser representada em diferentes padrões, tais como: o binário, Saco-de-Palavras, Frequência do Termo – Inverso da Frequência nos Documentos (*Term Frequency–Inverse Document Frequency*, TF-IDF). Para ilustrar as particularidades de cada padrão de vetorização, considera-se o *corpus*<sup>11</sup> da Tabela 2.4 formado por uma coletânea de quatro documentos, cada um contendo apenas uma única sentença. Devido à unicidade na quantidade de sentenças adotada no *corpus* exemplo, as descrições a seguir mostram as possíveis representações vetoriais em nível de documento e não em nível de sentença, embora isto seja igualmente viável.

**Tabela 2.4. Corpus exemplo**

<b>Documento 1 (D1)</b>	Primeira sentença do corpus
<b>Documento 2 (D2)</b>	A segunda sentença é curta
<b>Documento 3 (D3)</b>	A terceira é curta
<b>Documento 4 (D4)</b>	A quarta sentença é a maior do corpus

### 2.6.1. Modelo de Espaço Vetorial Binário

Consiste no modelo mais intuitivo de vetorização, em que para cada palavra é atribuído um valor 1 ou 0 de acordo com sua presença ou ausência na sentença. Embora simples, é possível constatar pela Tabela 2.4 que este padrão de representação é pobre do ponto de vista semântico, uma vez que não traz qualquer informação sobre a importância de um termo para o conjunto de textos. No entanto, este modelo de representação é bastante útil para técnicas que aplicam filtros sobre os dados em linguagem natural, já que permite a criação de máscaras binárias de comparação. Ademais, esse modelo de representação requer poucos recursos computacionais para a sua implementação.

**Tabela 2.5. Representação vetorial do corpus exemplo da Tabela 2.4 no modelo binário.**

Termos	primeira	quarta	a	corpus	curta	do	maior	segunda	sentença	terceira	é
<b>D1</b>	1	0	0	1	0	1	0	0	1	0	0
<b>D2</b>	0	0	1	0	1	0	0	1	1	0	1
<b>D3</b>	0	0	1	0	1	0	0	0	0	1	1
<b>D4</b>	0	1	1	1	0	1	1	0	1	0	1

### 2.6.2. Modelo de Espaço Vetorial de Saco-de-Palavras

O modelo de Saco-de-Palavras, tradução livre para *Bag-of-Words* (BoW), caracteriza-se como um tipo de modelo vetorial que atribui pesos aos termos, correspondentes ao número de ocorrências observadas do termos no texto. Matematicamente, os vetores dessa representação são expressos conforme a equação

$$V_D = [w_1, w_2, \dots, w_{n-1}, w_n], \quad (1)$$

<sup>11</sup>Linguisticamente, um *corpus* é uma coletânea de documentos sobre determinado tema. Um conjunto de *corpus* é denominado *corpora*.

em que  $V_D$  é o vetor de pesos  $w$  para cada sentença do documento  $D$  até o  $n$ -iésimo termo.

A Tabela 2.6 ressalta a presença de um peso igual a 2 na última linha da coluna referente ao termo “a”. Isto de fato está condizente com a quantidade de vezes que esse termo aparece em D4 na Tabela 2.4, entretanto não reflete a importância semântica para o *corpus* considerado.

**Tabela 2.6. Representação vetorial do *corpus* exemplo da Tabela 2.4 no modelo *Bag-of-Words*.**

Termos	primeira	quarta	a	corpus	curta	do	maior	segunda	sentença	terceira	é
D1	1	0	0	1	0	1	0	0	1	0	0
D2	0	0	1	0	1	0	0	1	1	0	1
D3	0	0	1	0	1	0	0	0	0	1	1
D4	0	1	2	1	0	1	1	0	1	0	1

Este modelo de representação, assim como seu antecessor, sofre do mesmo problema crítico, a presunção de uma igualdade de relevância de todos os termos perante ao *corpus*. Tal suposição pode conferir resultados questionáveis, uma vez que, termos com alta ocorrência em um único documento podem eventualmente ser supervalorizados em uma avaliação baseada na soma total de cada termo no *corpus* [Manning et al., 2010]. Embora esse modelo falhe ao identificar a importância semântica de um termo, o custo computacional para a sua implementação é baixo e permite identificar termos mais prevalentes tanto em um documento quanto em todo *corpora* através de operações simples, soma de colunas, com a matriz de pesos. Destaca-se ainda que o Saco de Palavras é um primeiro passo da implementação de modelos mais complexos.

### 2.6.3. Modelo de Espaço Vetorial Frequência do Termo – Inverso da Frequência nos Documentos

Esse modelo clássico de vetorização é definido pela equação:

$$tfidf_t = tf_{t,d} \times idf_t, \quad (2)$$

em que, para um termo  $t$ , a Frequência do Termo – Inverso da Frequência,  $tfidf_t$ , é o produto de duas medidas estatísticas, a **frequência do termo** (TF),  $tf_{t,d}$ , e o **inverso da frequência nos documentos** (IDF),  $idf_t$ . Embora o cálculo de frequência do termo ( $tf$ ) siga a mesma lógica apresentada na Seção 2.6.2, o diferencial está na sua ponderação por  $idf_t$ , uma parcela que remete a quanto esse termo é citado nos demais documentos. Em sua fórmula, expressa na equação:

$$idf_t = \log \frac{N}{df_t}, \quad (3)$$

define-se  $N$  como a contabilização do número de ocorrências do termo  $t$  no conjunto de documentos e  $df_t$  considera a frequência do termo  $t$  no documento em questão.

**Tabela 2.7. Representação vetorial do *corpus* exemplo da Tabela 2.4 no modelo TF-IDF.**

Termos	primeira	quarta	a	corpus	curta	do	maior	segunda	sentença	terceira	é
<b>D1</b>	0.614	0	0	0.484	0	0.484	0	0	0.392	0	0
<b>D2</b>	0	0	0.378	0	0.467	0	0	0.592	0.378	0	0.378
<b>D3</b>	0	0	0.408	0	0.505	0	0	0	0	0.640	0.408
<b>D4</b>	0	0.419	0.535	0.330	0	0.330	0.419	0	0.267	0	0.267

Essa modificação permite mensurar o grau de relevância semântica de um termo de um documento, em relação a toda coletânea. Como esperado, verifica-se que a Tabela 2.7 possui a mesma quantidade de linhas e colunas do modelo Saco-de-Palavras. Uma variante do TF-IDF original, é conhecido como TF-ISF (*Term Frequency – Inverse Sentence Frequency*), sendo largamente empregada na sumarização de textos em nível de sentença e não em nível de documento como o TF-IDF.

A representação pelo modelo TF-IDF, em relação às demais, é a que carrega maior correlação entre a semântica do termo e o seu peso no espaço vetorial. Essa representação é bastante útil em problemas que visam extrair conhecimento das bases de dados de acordo com a semântica dos documentos [de Oliveira et al., 2020]. No entanto, essa representação é sensível ao uso de sinônimos de palavras comuns. Como sinônimos pouco usuais têm baixa frequência de utilização, mesmo que se refiram a significados comuns amplamente representado por outras palavras, o termo sinônimo passa a ter alto peso na representação TF-IDF, embora possa não ser tão significativo para a representação do dado. Tal anomalia é frequentemente abordada em trabalhos que se baseiam em dicionários de sinônimos, tesauro (*thesaurus*), para normalizar o vocabulário do texto [Jarmasz e Szpakowicz, 2003].

Um ponto importante a ser esclarecido é que, independente da representação aplicada, a dimensão do vetor está vinculada à quantidade restante de palavras distintas contidas em todo o banco de dados, já que várias delas foram removidas durante as etapas descritas na Seção 2.5. As palavras mantidas na sentença são as que carregam significado e, portanto, são as mais importantes para o entendimento da ideia central do texto. Ao se considerar a modelagem de problemas de aprendizado de máquina baseados no processamento de linguagem natural, as palavras remanescentes são as características do conjunto de dados sobre o qual deseja-se fazer o aprendizado.

#### 2.6.4. Modelo de Espaço Vetorial de *Feature Hashing*

Diferentemente das representações anteriores, a representação por *Feature Hashing* delimita o tamanho do espaço vetorial com base em posições em uma tabela *hash*. Essa representação usa uma função *hash* para geração dos vetores, a qual mapeia dados de tamanho variável em índices de uma tabela de tamanho fixo, denominada tabela *hash*, ou tabela de dispersão. No contexto da vetorização, os índices resultantes correspondem aos termos analisados. Cada documento pode ser representado a partir dos  $N$  índices da tabela, de forma que, para um agrupamento de  $M$  documentos, a sua representação matemática é verificada por meio de uma matriz  $M \times N$ , que identifica a coleção de documentos (*corpora*). A determinação de  $N$  é arbitrária, podendo ser menor ou igual

a quantidade total de termos (*tokens*). Entretanto, o valor ótimo de posições deve ser avaliado pois, sendo inferior à quantidade de termos observados nos documentos, a representação pode apresentar inconsistência, uma vez em que há a colisão de termos em índices comuns que podem armazenar informações não correlatas.

Para a representação dos *corpus* exemplo conforme o modelo de *feature hashing* foram selecionados 5 índices, arbitrariamente, considerando um vocabulário de 11 palavras distintas. Assim, os vetores são verificados na Tabela 2.8.

**Tabela 2.8. Representação vetorial do *corpus* exemplo da Tabela 2.4 usando o modelo de *feature hashing*. Diferentemente dos demais modelos, são observadas apenas 5 colunas para representação dos documentos, o que corresponde ao número de índices da tabela de dispersão.**

<i>Hashes</i>	Índice 1	Índice 2	Índice 3	Índice 4	Índice 5
<b>D1</b>	1	1	1	0	0
<b>D2</b>	0	1	1	1	1
<b>D3</b>	0	1	1	0	1
<b>D4</b>	1	3	1	1	1

Esse modelo de espaço vetorial fornece uma representação compacta dos dados, ao custo de uma menor granularidade semântica, já que cada índice da tabela *hash* pode conter dados não correlacionados semanticamente.

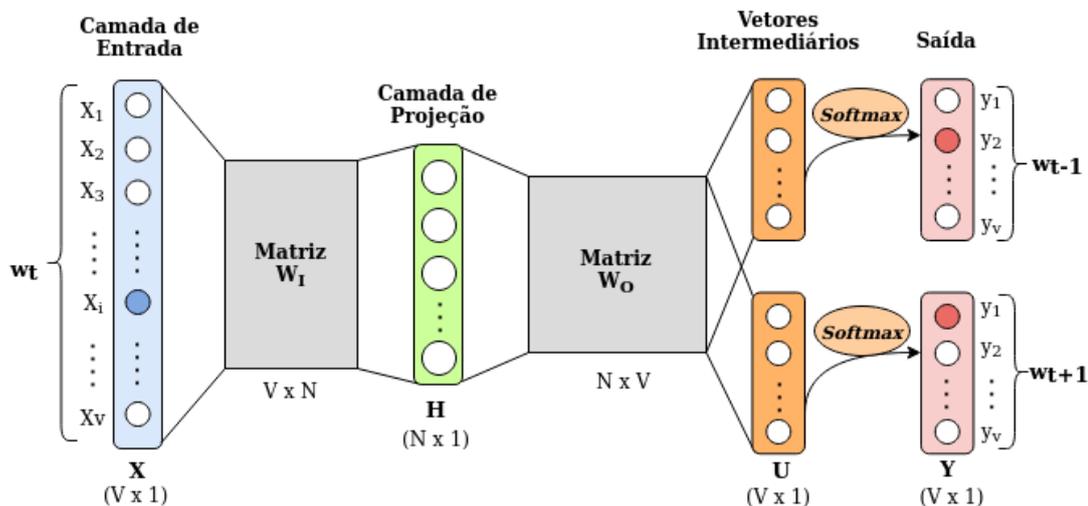
### 2.6.5. Incorporações de Palavras (*Word Embeddings*)

A escolha de tratar as palavras como unidades atômicas, isto é, sem uma conexão semântica entre si, traz simplicidade e robustez ao modelo de espaço vetorial. Apesar de possibilitar uma avaliação da similaridade entre frase ou documento, esses modelos inviabilizam uma medição por palavra, tornando palavras com sentidos próximos como “mar” e “oceano” invisíveis à modelagem vetorial. Uma consequência imediata dessa carência semântica é a dificuldade de lidar com sinônimos. Outra desvantagem é a alta dimensionalidade, um reflexo do caráter esparso dos vetores gerados [Camacho-Collados e Pilehvar, 2018, Mikolov et al., 2013].

Como alternativa, as incorporações de palavras (*words embeddings*) surgem como uma forma de representação distribuída de palavras, idealizada segundo a hipótese distribucional. Nesta hipótese, cada palavra é caracterizada pela sua vizinhança, expressando, portanto uma tendência de palavras com significados semelhantes que aparecerem em contextos similares [Firth, 1957]. Tais representações de palavras podem ser obtidas aplicando modelos preditivos baseados em redes neurais que, quando treinados com grandes volumes de dados textuais, incorporam a semântica das palavras em vetores de baixa dimensão, densos e de tamanho fixo. A principal vantagem da representação vetorial individualizada para cada palavra consiste na preservação das relações semânticas e sintáticas entre palavras, permitindo assim que sinônimos ou palavras minimamente relacionadas sejam mapeadas em vetores semelhantes [Li et al., 2015].

A popularização das técnicas de incorporações de palavras ocorreu através da *Word2Vec* [Mikolov et al., 2013], uma ferramenta que computa a representação vetorial de palavras através de dois modelos possíveis, o Saco de Palavras Contínuo (*Continuous*

*Bag-of-Words*, CBOW) e a *Skip-gram*. Ambos os modelos dividem os textos em dois grupos, palavra-alvo e contexto. Em especial, o contexto é interpretado como um conjunto limitado das palavras que circundam a palavra-alvo. O tamanho dessa limitação, conhecida como janela, define o número de palavras a serem consideradas a esquerda e a direita da palavra-alvo.



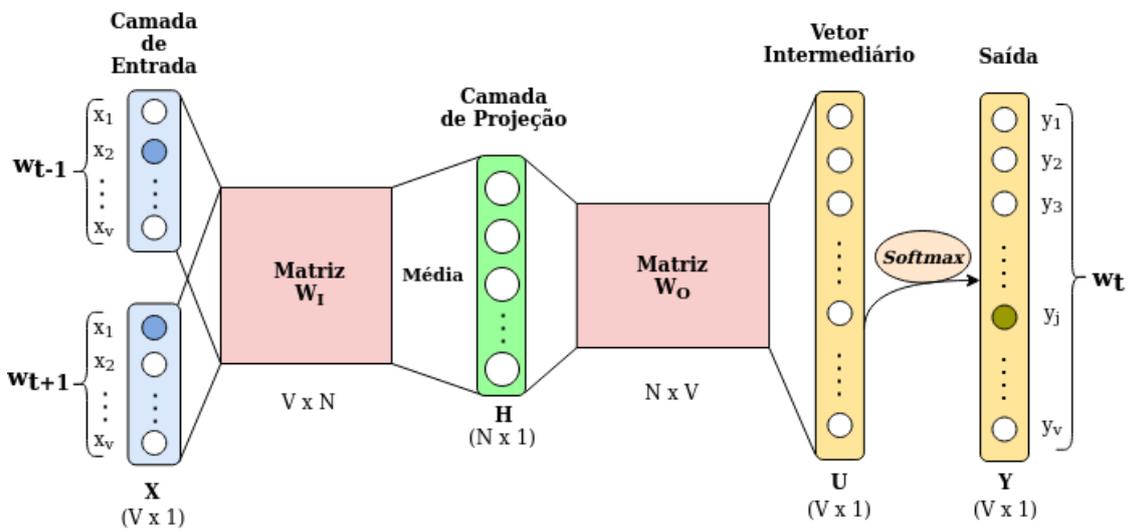
**Figura 2.3.** Ilustração da arquitetura do modelo Skip-gram considerando como entrada a palavra-alvo  $w_t$  codificada no seu vetor one-hot  $X$ . Este vetor representa a palavra-alvo como uma sequência de  $V$  zeros, exceto por um único valor um na posição  $x_i$ . Na saída do modelo são obtidos  $C$  vetores de distribuição de probabilidade, um para cada palavra do contexto. Com o modelo devidamente treinado, espera-se que as maiores probabilidades de cada vetor  $Y$ , encontradas nas posições  $y_2$  e  $y_1$ , expressem as palavras de contexto  $w_{t-1}$  e  $w_{t+1}$ .

A particularidade do modelo Skip-gram está na sua capacidade usar uma palavra-alvo  $w_t$  na predição do contexto de palavras  $W_t = [w_{t-j}, \dots, w_{t+j}]$  que a circunda. Como ilustrado na Figura 2.3, a arquitetura do modelo Skip-gram é composta pelas camadas de entrada e saída, intercaladas por uma camada de projeção. O tamanho da camada de entrada, assim como da camada de saída, está atrelado ao número de palavras  $V$  existentes no vocabulário usado no treinamento. Já o tamanho da camada de projeção é determinado com base em um parâmetro  $N$  arbitrário, que expressa a dimensão do futuro vetor de palavras gerado  $H$  (*word embeddings*). Esta dimensão indica a quantidade de características usadas na representação numérica de cada palavra, sendo portanto inferior à dimensão do vetor original de cada palavra inserido na camada de entrada. A conexão da camada de entrada para a camada de projeção é feita através de uma matriz de pesos  $W_I$  de tamanho  $V \times N$ . Analogamente, a conexão da camada de projeção para a camada de saída é desempenhada pela matriz  $W_O$  de tamanho  $N \times V$ . Como usualmente feito antes do treinamento de redes neurais, ambas as matrizes de peso  $W_I$  e  $W_O$  são inicializadas com valores aleatórios pequenos. A inserção de uma palavra-alvo na camada de entrada da rede neural inicia com a codificação desta palavra em seu vetor *one-hot*, uma matriz coluna  $N \times 1$  usada para distinguir cada palavra em um vocabulário. Este vetor consiste em 0s em todas as posições, com exceção de um único 1 em uma posição usada exclusivamente para identificar a palavra.

No processo de treinamento, a cada iteração são empregados dois algoritmos de aprendizado: de propagação direta (*forward propagation*) e de retropropagação (*back-propagation*). Aplicando primeiramente o algoritmo de propagação direta, o vetor *one-hot* da palavra-alvo de entrada é multiplicado pela matriz de pesos  $W_I$  para formar o vetor  $H$  da camada oculta. Em seguida, o vetor  $H$  é então multiplicado por  $W_O$  gerando assim  $C$  vetores intermediários idênticos, cada um representando uma palavra de contexto. As saídas do modelo são adquiridas aplicando a cada vetor intermediário a função *softmax*:

$$p(w_{t+j}|w_t) = \frac{\exp(v'_{w_o} \top v_{w_t})}{\sum_{i=1}^V \exp(v'_{w_i} \top v_{w_t})}, \quad (4)$$

em que dada a palavra-alvo  $w_t$ ,  $v_{w_t}$  é sua linha correspondente na matriz de peso  $W_I$  e  $v_w$  é sua coluna correspondente na matriz  $W_O$ . Esta função normaliza o vetor intermediário  $U$  composto por  $V$  números flutuantes, transformando-o no vetor de distribuição de probabilidade  $Y$ . Uma vez descoberto o vetor normalizado de probabilidades de cada palavra de contexto, o algoritmo de retropropagação os compara com o vetor *one-hot* da palavra correspondente para assim atualizar as matrizes de peso  $W_I$  e  $W_O$ . Essa atualização ocorre especificamente nos valores da coluna correspondente de  $W_O$  e da linha correspondente de  $W_I$ .



**Figura 2.4.** Ilustração da arquitetura do modelo CBOW considerando como entrada as palavras de contexto  $w_{t+1}$  e  $w_{t-1}$  codificadas em seus vetores *one-hot*. Na saída do modelo é obtido um vetor de distribuição de probabilidade. Com o modelo devidamente treinado, espera-se que a maior probabilidade do vetor  $Y$ , encontrada na posição  $y_i$ , expresse a palavra-alvo  $w_t$ .

Ao inverter a atuação da palavra-alvo e as palavras de contexto na rede neural, a arquitetura do modelo CBOW torna possível a predição de uma palavra-alvo a partir do contexto de palavras próximas. Como consequência dessa inversão, o modelo admite múltiplas entradas, uma para cada palavra de contexto. Essa multiplicidade de vetores de entrada deriva a necessidade do cálculo da média de seus vetores de palavras correspondentes, estes construídos pela multiplicação dos múltiplos vetor *one-hot* de entrada e

pela matriz  $W$ . Uma segunda consequência é a presença uma única função *softmax*, ao contrário das  $C$  existentes na arquitetura do modelo Skip-gram [Hu et al., 2016]. O modelo CBOW converge de maneira mais rápida em relação ao Skip-gram. Contudo, este apresenta melhores resultados para palavras pouco frequentes em relação àquele.

## 2.7. Aprendizado Sobre Dados de Redes Sociais em Linguagem Natural

Aprendizado de máquina é inerentemente um campo multidisciplinar, focado na construção de programas de computador que melhoram automaticamente com a experiência [Boutaba et al., 2018]. O aprendizado de máquina está relacionado à extração de conhecimento a partir de dados brutos. Os algoritmos de aprendizado de máquina têm como objetivo descobrir como realizar tarefas importantes generalizando as suas operações a partir de exemplos de dados [Domingos, 2012]. Embora existam diferentes definições para o aprendizado de máquina, todas convergem para a ideia de usar algoritmos para obter dados, aprender com eles e então determinar ou prever algum fenômeno. Existem diferentes algoritmos de aprendizado de máquina, cada qual indicado para um tipo de saída desejada. O **aprendizado supervisionado**, também intitulado aprendizagem com exemplos, pressupõe a existência de entradas e saídas marcadas, compondo um conjunto de treinamento, para assim aprender uma regra geral que mapeia as entradas em saídas. Em contraste, o **aprendizado não-supervisionado**, independe de qualquer marcação sobre os dados, forçando o algoritmo a identificar padrões entre as entradas, de modo que as entradas que têm algo em comum sejam agrupadas na mesma categoria. O **aprendizado por reforço** aprende à medida que interage com um ambiente dinâmico e, dessa maneira, qualquer ação que tenha algum impacto no ambiente fornece uma retroalimentação que orienta o algoritmo [Ayodele, 2010].

Esta seção discute as técnicas de redução de dimensionalidade a partir das representações vetoriais extraídas dos dados em linguagem natural, apresenta as métricas de similaridade possíveis de serem usadas sobre os dados, os algoritmos supervisionados e não-supervisionados e, por fim, métricas de avaliação dos algoritmos.

### 2.7.1. Redução Dimensional

Ao utilizar base de dados extensas, ainda mais sendo composta por textos de domínios de conhecimento heterogêneos, é inevitável lidar com vetores de características extremamente longos. Além da elevação da complexidade computacional, o uso de representações vetoriais demasiadamente grandes pode não ser a opção mais adequada. Essa hipótese é confirmada no problema conhecido como “maldição da dimensionalidade”, o qual expressa a existência um número ótimo de características que podem ser selecionados em relação ao tamanho da amostra para maximizar o desempenho do aprendizado [Zhai et al., 2014]. Nesse cenário, torna-se conveniente a aplicação de algum procedimento para redução da base de dados, seja pela seleção de características originais ou através de técnicas de redução da dimensionalidade. Esta última alternativa tem o objetivo de encontrar representações vetoriais menos complexas, criando novas características sintéticas a partir das originais.

Redução de dimensionalidade é o processo de derivar um conjunto de graus de liberdade menor que reproduza a maior variabilidade de um conjunto de dados [Zhai et al.,

2014, Andreoni Lopez et al., 2019]. Idealmente, a representação reduzida deve ter uma dimensionalidade que corresponda à dimensionalidade intrínseca dos dados. A dimensionalidade intrínseca dos dados é o número mínimo de parâmetros para contabilizar as propriedades observadas nos dados. Matematicamente, na redução da dimensionalidade, dada a variável aleatória  $p$ -dimensional  $x = (x_1, x_2, \dots, x_p)$ , calcula-se uma representação dimensional inferior a ela,  $s = (s_1, s_2, \dots, s_k)$  com  $k \leq p$ .

Diferentes abordagens são propostas para reduzir a dimensionalidade, classificadas em lineares ou não lineares. A redução linear da dimensionalidade é uma projeção linear dos dados originais, na qual os dados  $p$ -dimensionais são reduzidos em dados  $k$ -dimensionais usando  $k$  combinações lineares de  $p$  características originais. Dois exemplos importantes de algoritmos de redução de dimensão linear são a análise de componentes principais (*Principal Component Analysis* - PCA) e a análise de componente independente (*Independent Component Analysis* - ICA). O objetivo da PCA é encontrar uma transformação linear ortogonal que maximize a variância das características. O primeiro vetor base do PCA, a componente principal, descreve a direção de maior variabilidade dos dados. O segundo vetor é a segunda melhor descrição e deve ser ortogonal ao primeiro e assim por diante em ordem de importância. De forma semelhante, o objetivo da ICA é encontrar uma transformação linear, na qual os vetores da base sejam estatisticamente independentes e não gaussianos, ou seja, a informação mútua entre duas características no novo espaço vetorial é igual a zero. Ao contrário da PCA, os vetores de base na ICA não são ortogonais nem classificados em ordem. Todos os vetores são igualmente importantes. A PCA é geralmente aplicada para reduzir a representação dos dados. Por outro lado, a ICA normalmente é usada para obter a extração de características, identificando e selecionando as características que melhor se adaptam à aplicação. Métodos não-lineares aplicam transformadas nos dados, levando-os a um novo espaço vetorial, no qual aplicam métodos lineares.

Direcionada especialmente para representações vetoriais derivadas de textos, a **Indexação Semântica Latente**<sup>12</sup> (*Latent Semantic Indexing*, LSI) é uma técnica de redução dimensional baseada na Decomposição em Valores Singulares (*Singular Value Decomposition*, SVD). Sua adaptabilidade a dados de origem textual está atrelada a natureza esparsa dos dados. A LSI propõe construir um espaço “semântico” em que termos e documentos intimamente associados são colocados próximos uns dos outros.

Supondo  $A$  como a matriz original  $n \times m$ , em que termos e documentos são representados em linhas e colunas respectivamente, a aplicação da LSI inicia-se pela adoção de um nível de aproximação  $k$ . Com isso,  $A$  pode ser decomposta da seguinte forma:

$$A \approx A_k = U_k D_k V_k^T, \quad (5)$$

em que  $A_k$  é uma aproximação de  $A$ , composta pelo produto da matriz de termo-conceito  $U_k$ , a matriz de valores singulares  $D_k$  e a matriz de conceito-documento  $V_k$ . Assim, esta matriz  $A_k$  expressa a melhor representação da estrutura semântica do *corpus* original, omitindo todos, exceto os  $k$  maiores valores singulares na decomposição. Por tal razão, a LSI

<sup>12</sup>Também referenciada como Análise Semântica Latente (*Latent Semantic Analysis*, LSA) para propósitos para além da área de recuperação da informação.

é também conhecida como SVD truncada [Papadimitriou et al., 2000, Deerwester et al., 1990]. A respeito da escolha de  $k$ , esta é feita através de testes empíricos, avaliando a taxa de variância dos valores singulares. O valor de  $k$  deve ser pequeno o suficiente para permitir uma recuperação rápida da informação e grande o suficiente para capturar adequadamente a estrutura do *corpus*. Para dados textuais, a redução da dimensionalidade é preferível de ser realizada pela técnica LSI em comparação à PCA ou ICA, pois, devido à natureza esparsa dos dados, as técnicas PCA e ICA apresentam resultados menos significativos ou falhos, enquanto a LSI é adequada a dados esparsos.

As técnicas de redução de dimensionalidade carecem de expressividade, pois as características geradas são combinações de outras características originais. Portanto, o significado da nova característica sintética é perdido. Quando há a necessidade de interpretação do modelo, por exemplo, ao criar filtros baseados em textos em linguagem natural, é necessário utilizar outros métodos. As técnicas de **seleção de características** produzem um subconjunto das características originais, que são as melhores representantes dos dados. Assim, não há perda de sentido. Existem três tipos de técnicas de seleção de características [Andreoni Lopez et al., 2019]: *wrapper*, filtro e incorporadas.

Os métodos *wrapper*, também chamados de laço fechado, utilizam diferentes classificadores, como máquina vetor de suporte (SVM), árvore de decisão, entre outros, para medir a qualidade de um subconjunto de características sem incorporar conhecimentos sobre a estrutura específica da função de classificação. Assim, o método avalia subconjuntos com base na precisão do classificador. Esses métodos consideram a seleção de característica como um problema de busca, criando um problema *NP*-difícil. Uma pesquisa exaustiva no conjunto de dados completo deve ser feita para avaliar a relevância do recurso. Os métodos *wrapper* tendem a ser mais precisos do que os métodos de filtro, mas apresentam um custo computacional mais alto [Andreoni Lopez et al., 2019]. Um método *wrapper* popular por sua simplicidade é o *Sequential Forward Selection* (SFS). O algoritmo começa com um conjunto vazio  $S$  e o conjunto completo de todas as características  $X$ . O algoritmo SFS faz uma pesquisa e gradualmente adiciona características, selecionados  $S$  por uma função de avaliação, minimizando o erro quadrático médio (MSE). A cada iteração, o algoritmo seleciona uma característica a ser incluída em  $S$  entre as características disponíveis restantes em  $X$ . A principal desvantagem do SFS é que adicionar uma nova característica ao conjunto  $S$  evita que o método remova qualquer característica que tenha o menor erro após adicionar outras. Os métodos de filtro são computacionalmente mais leves do que os métodos de *wrapper* e evitam o sobreajuste. Os métodos de filtro também chamados de métodos de laço aberto, usam heurísticas para avaliar a relevância da característica no conjunto de dados [Chandrashekar e Sahin, 2014]. O algoritmo filtra a característica que preenche o critério heurístico. Um dos algoritmos de filtragem mais populares é o Relief. O algoritmo Relief associa cada característica a uma pontuação, que é calculada como a diferença entre a distância do exemplo mais próximo da mesma classe e o exemplo mais próximo da outra classe. A principal desvantagem desse método é a obrigatoriedade de rotular os registros de dados com antecedência. Relief é limitado a problemas com apenas duas classes, mas ReliefF [Robnik-Šikonja e Kononenko, 2003] é uma melhoria do método Relief que lida com classes múltiplas usando a técnica dos  $k$  vizinhos mais próximos. Os métodos incorporados têm um comportamento semelhante aos métodos *wrapper*, usando a precisão de um classificador para avaliar a relevância da

característica. No entanto, os métodos incorporados fazem a seleção de características durante o processo de aprendizagem e usam suas propriedades para orientar a avaliação da característica. Essa modificação reduz o tempo computacional em relação aos métodos *wrapper*. O *Support Vector Machine Recursive Feature Elimination* (SVM-RFE) classifica as características de acordo com um problema de classificação baseado no treinamento de uma máquina vetor de suporte (SVM) com um kernel linear. O elemento com a menor classificação é removido, de acordo com o critério  $w$ , em forma de eliminação reversa sequencial. O critério  $w$  é o valor do hiperplano de decisão no SVM.

### 2.7.2. Métricas de Similaridade e Dissimilaridade

Medidas de similaridade e dissimilaridade desempenham um papel crítico na quantificação da semelhança ou distância semântica, respectivamente, entre textos. Independente dos elementos textuais comparados, caracteres, termos, *strings* ou *corpus*, tais medidas estão constantemente presentes na resolução de problemas de análise de padrões, sejam para fins de sumarização, classificação ou agrupamento de textos. Supondo um par de vetores  $A$  e  $B$  não nulos, compostos pela mesma quantidade  $n$  de termos, tal que  $A = [x_1, x_2, \dots, x_n]$  e  $B = [y_1, y_2, \dots, y_n]$ , é possível medir a relação semântica entre eles de diversas formas, tais como Distância Euclidiana, Distância de Manhattan e Similaridade do Cosseno.

A métrica de dissimilaridade conhecido como **Distância de Minkowski** é dada pela equação:

$$Dis(A, B) = \left( \sum_{i=1}^n |x_i - y_i|^p \right)^{\frac{1}{p}}. \quad (6)$$

Tal métrica é uma generalização de outras duas igualmente conhecidas, a **Distância de Manhattan** e a **Distância Euclidiana**, para  $p$  igual a 1 ou 2 respectivamente. Obviamente, espera-se que quanto mais próximo de zero for o valor de  $Dis$ , mais similar  $A$  e  $B$  serão.

Dentre as métricas de similaridade entre conjunto de termos, destaca-se a **Similaridade do Cosseno** que emprega o conceito de produto interno. Sendo definida entre  $[-1, 1]$ , valores dessa medida mais próximos ao limite superior simbolizam uma maior proximidade entre os vetores de termos. Matematicamente, a similaridade do cosseno entre  $A$  e  $B$  é demonstrada pela equação:

$$Sim(A, B) = \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}}. \quad (7)$$

### 2.7.3. Algoritmos supervisionados

A distinção dos algoritmos supervisionados pode ser feita definindo aqueles cujo resultado esperado são variáveis de valor real, intitulados algoritmos de regressão, e aqueles cujo resultado são categorias representadas por valores discretos, conhecidos como algoritmos de classificação. Algoritmos de classificação são o foco do trabalho devido à natureza classificatória das aplicações de processamento de linguagem natural abordadas.

### 2.7.3.1. Máquina de Vetor de Suporte

A Máquina de Vetor de Suporte, *Support Vector Machine* (SVM), consiste em um tipo de algoritmo classificador linear, baseado no conceito de um plano de decisão que define os limites de decisão. O processo decisório acontece através da geração de um hiperplano multidimensional ótimo que separa as amostras em classes, maximizando a distância entre as classes ou a margem de separação. Tal hiperplano é traçado por um subconjunto de amostras, denominados vetores suporte. O caráter ótimo da separação é assegurado pela definição de uma função *kernel* que minimiza a função erro. Embora seja essencialmente um classificador binário, a SVM é igualmente adaptável a um problema multiclases, em que divide-se o problema original em subproblemas de classificação binária.

Ao lidar com conjunto de amostras não linear, uma estratégia é adotar o artifício de utilizar uma função de *kernel*, em que uma função encontra um novo espaço dimensional, obrigatoriamente maior que o original, que viabilize a separação usando o hiperplano. Dentre as funções *kernels* mais utilizados estão a Linear, Polinomial, *Radial Basis Function* (RBF) e Sigmoid. A capacidade da SVM ser menos propensa ao sobreajuste (*overfitting*), ou seja, obter uma função de separação de complexidade superior à necessária, está intimamente relacionada ao grau de relevância atribuído a amostras longe do limite de separação. Basicamente, uma vez encontrado o hiperplano, a maioria dos dados que não sejam os vetores de suporte são vistos como redundantes.

O uso de algoritmos supervisionados para a detecção de notícias falsas depende de uma grande base de dados contendo tanto notícias verdadeiras, como falsas. Contudo, isso impõe a limitação de haver uma base rotulada com notícias reais e falsas. As notícias falsas, embora sejam cada vez mais numerosas, são difusas nas redes sociais e tendem a ser voláteis, já que algum período após a disseminação perdem a credibilidade. Uma estratégia para contrapor a limitação no número de notícias falsas para o treinamento dos classificadores é o aprendizado de uma única classe, como o baseado no algoritmo **Máquina de Vetor de Suporte de Classe Única** (*One-class Support Vector Machine*). A SVM de classe única é um algoritmo de aprendizado supervisionado que deriva um hiperplano de decisão para detecção de anomalias. Novos dados são classificados como semelhantes ou diferentes do conjunto de treinamento. Em contraste com as implementações típicas da SVM, a classe única leva em consideração um conjunto de amostras de treinamento de uma única classe. Qualquer nova amostra que não se encaixe na superfície de decisão definida pelo conjunto de treinamento é considerada uma instância de uma nova classe e, portanto, uma notícia falsa [Perdisci et al., 2006, Gaonkar et al., 2019].

### 2.7.3.2. Floresta Aleatória

A Floresta Aleatória (*Random Forest*, RF) é um algoritmo popular de classificação ou regressão, que opera construindo múltiplas árvores de decisão durante o processo de treinamento. Durante o treinamento, a RF possibilita a aplicação do método de *bagging*, que permite treinar repetidamente o algoritmo com o mesmo conjunto de dados, entretanto, selecionando as características aleatoriamente. Ilustrativamente, para um conjunto

de treinamento com  $X = x_1, x_2, \dots, x_n$  amostras de entrada e respectivos  $Y = y_1, y_2, \dots, y_n$  amostras de saída o método *bagging* implica a seleção aleatória e com repetição dessa base de dados  $K$  vezes. Assim, as árvores são treinadas com a mesma informação, de maneira em que o resultado final é formado pelas predições individuais  $m_i$  de cada árvore do conjunto, conforme a equação:

$$\hat{m} = \frac{1}{K} \sum_k^{i=1} m_i. \quad (8)$$

Uma vantagem relevante da RF para ao modelo tradicional de árvores de decisão é o fato de não ser considerado todo o conjunto de dados, mas apenas um subconjunto dele. Isto implica uma maior aleatoriedade no modelo, auxiliando na correção do sobreajuste. No mesmo sentido, ao incrementar o número de árvores de decisão na RF, a taxa de erro do conjunto de testes converge para um limite, significando que RF mais povoadas são menos suscetíveis ao sobreajuste [Verikas et al., 2011].

### 2.7.3.3. k-Vizinhos Mais Próximos

O algoritmo k-Vizinhos Mais Próximos (*k-Nearest Neighbors*, kNN) depende da escolha prévia de um parâmetro  $k$ , que condiciona o número de amostras vizinhas mais próximas usadas no critério de classificação. A partir de uma amostra ainda não classificada, o algoritmo aplica uma métrica de distância, ou similaridade, entre essa amostra e todas as demais já classificadas. Filtrando as  $k$  amostras vizinhas que tiveram as menores distâncias. O algoritmo verifica e contabiliza a quantidade de amostras integrantes em cada classe. Finalmente, a amostra é alocada na classe majoritária dos  $k$  vizinhos mais próximos. Essa dependência sobre o valor do parâmetro inicial faz com que o resultado do algoritmo apresente diversas classificações, se  $k$  for muito alto, ou apresente amostras ruidosas, se  $k$  for muito pequeno. Ao ser obrigado a calcular a distância de cada amostra nova com todas as demais já classificadas, o algoritmo requer um consumo computacional maior, sendo assim não indicado para *corpus* muito grandes [Kadhim, 2019]. Ressalta-se também o alto consumo de memória do algoritmo, já que é necessário carregar todo o conjunto de dados em memória para a comparação com as novas amostras.

### 2.7.4. Algoritmos não-supervisionados

Algoritmos de agrupamento são a forma mais comum de aprendizado não-supervisionado. Apesar de possuírem lógica operacional, caso de uso, escalabilidade e desempenhos distintos, o propósito genérico de usar esses algoritmos é a segregação de termos em grupos (*clusters*) de acordo com suas características semânticas. Esse procedimento de separação em grupos é conhecido como agrupamento.

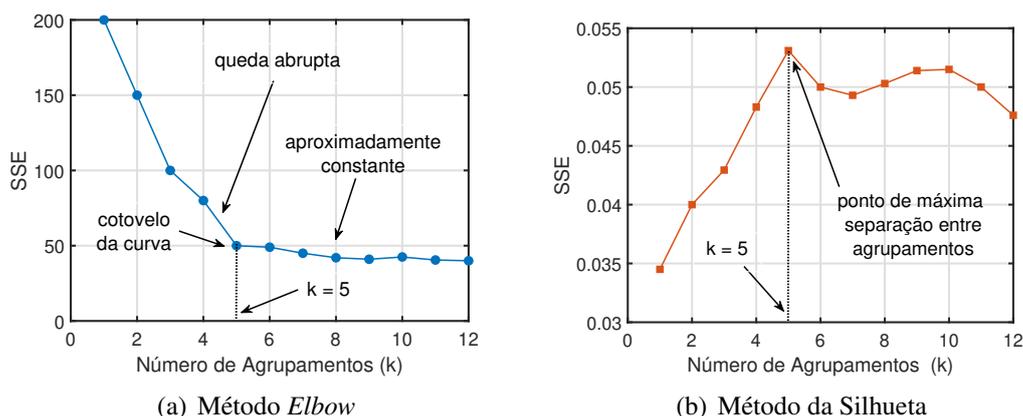
#### 2.7.4.1. Algoritmos Baseados no Particionamento

Essa classificação é dada àqueles algoritmos que são semelhantes no sentido de que cumprem simultaneamente dois critérios no processo de agrupamento de dados. O

primeiro critério expressa a obrigação de ter pelo menos uma amostra em cada agrupamento criado. O segundo refere-se a uma exclusividade de pertencimento, em que cada amostra deve pertencer a somente um agrupamento [Xu e Wunsch, 2005, Fahad et al., 2014].

Um exemplar clássico desse tipo de algoritmo é o *k-means*, uma heurística capaz de particionar dados em  $k$  agrupamentos pela minimização da soma dos quadrados das distâncias em cada agrupamento. Sua lógica de execução, parte da escolha aleatória dos centroides de cada agrupamento seguida do cálculo de distância entre cada amostra e os centroides, segundo uma das métricas de dissimilaridade, ou similaridade, vistas na Seção 2.7.2. Posteriormente cada amostra é alocada no agrupamento cujo centroide está mais próximo. A cada nova amostra alocada a um agrupamento, o centroide é recalculado podendo ocorrer eventuais redistribuições de amostras para outros grupos. O algoritmo finaliza quando cessam essas alterações na alocação das amostras aos agrupamentos.

Outro exemplo é o algoritmo *k-medoids*, indicado para pequenos conjuntos de dados, e que também particiona os dados em  $k$  grupos adotando o critério de minimizar a soma dos quadrados das distâncias em cada grupo. Embora lembre o *k-means*, sua diferença está no fato de escolher efetivamente uma das amostras de entrada como centro dos agrupamentos, não pontos médios como o *k-means*. Essa característica de tomada de decisão se traduz em maior robustez a dados ruidosos e *outliers*, além de uma capacidade de lidar com alta dimensionalidade, útil em representações vetoriais de dados textuais [Xu e Wunsch, 2005, Fahad et al., 2014]. Outra vantagem do *k-medoids* em relação ao *k-means* está no fato de as saídas do *k-medoids* serem mais facilmente interpretadas, dado que os centros dos agrupamentos são amostras reais, ao contrário do *k-means* que fornece um ponto que pode representar uma amostra de dados inviável.



**Figura 2.5. Métodos complementares para determinar o número ótimo de agrupamentos. Ambos os métodos idealmente tendem a convergir para um mesmo  $k$ , verificado neste exemplo como  $k=5$ .**

Contudo ambos os algoritmos, assim como outros, estão sujeitos à uma desvantagem singular: a indeterminação quanto ao número adequado de grupos  $k$ . A fim de contornar essa indeterminação, são usados dois métodos, *Elbow* e da Silhueta, para analisar previamente a conformidade dos dados a quantidades diferentes de grupos e, assim,

obter um resultado adequado aos dados. Em particular, o *Elbow* mede a compactação dos agrupamentos estabelecendo uma relação entre o número de agrupamentos e sua influência na variação total dos dados dentro do grupo. Graficamente, o melhor valor de  $k$  é encontrado identificando o ponto em que o ganho da curva diminui drasticamente, permanecendo aproximadamente constante depois disso. De forma análoga, o método da Silhueta mede a qualidade de um agrupamento. O número ideal de agrupamentos  $k$  é aquele que maximiza a silhueta média em uma faixa de valores possíveis para  $k$  [Ketchen e Shook, 1996, Rousseeuw e Kaufman, 1990]. A Figura 2.5 mostra um exemplo hipotético de uso dos métodos Cotovelo (*Elbow*) e da Silhueta. Nesse exemplo hipotético, é visto que para o valor  $k = 5$  há uma mudança brusca no erro médio quadrático (SSE) interno aos agrupamentos no método Elbow e, para  $k = 5$ , também há um ponto máximo do erro médio quadrático entre os agrupamentos no método da Silhueta, indicando maior separação entre agrupamentos.

Ressalta-se ainda que há variações dos algoritmos *k-means* e *k-medoids* que consideram graus de pertinência de uma amostra a diversos grupos. Nesses casos, chamados *fuzzy k-means* e *fuzzy k-medoids*, o centro dos agrupamentos são calculados considerando a pertinência parcial de cada amostra aos agrupamentos.

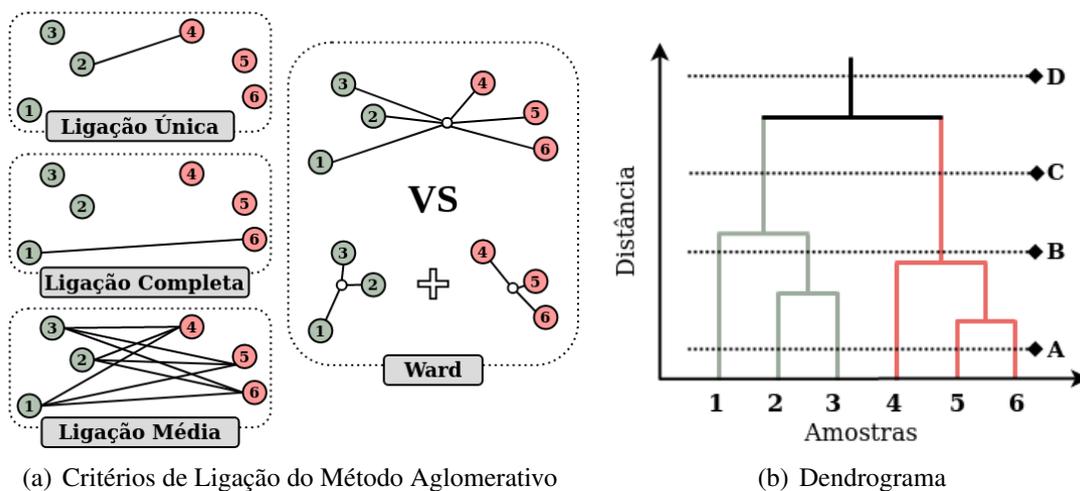
#### 2.7.4.2. Algoritmos Baseados em Densidade

Algoritmos de agrupamento baseados em densidade compartilham uma relação próxima com a abordagem do vizinho mais próximo (*nearest neighbour*). Nesse sentido, um agrupamento, definido como um componente denso conectado, cresce em qualquer direção que a densidade o conduza. Essa lógica de formação dos agrupamentos está diretamente relacionada à principal vantagem desses algoritmos em relação ao grupo dos algoritmos de particionamento, a possibilidade de descobrir agrupamentos com formas arbitrárias, diferente dos agrupamentos tipicamente esféricos retornados pelo algoritmo *k-means*, por exemplo.

Dentre os algoritmos baseados em densidade, o algoritmo de **Clusterização Espacial Baseada em Densidade de Aplicações com Ruído** (*Density Based Spatial Clustering of Application with Noise – DBSCAN*) é o mais popular. Seu intuito é encontrar regiões que satisfaçam uma densidade de pontos mínima estabelecida e que sejam separadas por regiões de menor densidade. Para isso, o algoritmo realiza uma estimativa simples do nível de densidade mínimo, definindo um limite para o número de vizinhos, *minPts*, dentro de um raio  $\epsilon$ . Assim, uma amostra com mais de *minPts* vizinhos dentro desse raio, é considerada um ponto central. Analogamente, uma amostra é considerada como de borda, se dentro de sua vizinhança concentram-se menos amostras que o mínimo definido, porém ainda pertencem à vizinhança de um ponto central qualquer. Por último, amostras que não são alcançáveis por densidade a partir de qualquer ponto central, ou seja, não se configuram nem como pontos centrais nem de borda, são rotulados como *outliers*. Uma desvantagem associada ao seu uso consiste na sua complexidade fortemente polinomial, que requer  $\Omega(n^{\frac{4}{3}})$  tempo para convergir, em que  $n$  é o tamanho do conjunto de dados [Fahad et al., 2014, Gan e Tao, 2015, Schubert et al., 2017].

### 2.7.4.3. Algoritmos Hierárquicos

Os algoritmos hierárquicos não apenas criam agrupamentos, mas consideram uma lógica multinível e calculam uma representação hierárquica dos dados de entrada. Esta representação é um tipo particular de árvore, em que os nós-folhas expressam dados individuais, e pode ser construída seguindo um método aglomerativo ou divisivo. O método aglomerativo, conhecido também como abordagem *bottom-up*, começa considerando cada amostra como um agrupamento unitário e mescla recursivamente duas ou mais em um novo agrupamento seguindo uma função de ligação escolhida. Tais funções, quando associados à métricas de distância ou de similaridade, definem critérios únicos que elegem os agrupamentos mesclados de cada iteração. A função de ligação única (*single-linkage*), por exemplo, estabelece a união considerando a distância entre as amostras mais próximas de cada agrupamento. De forma oposta, a função de ligação completa (*complete-linkage*) considera a distância das amostras mais distantes entre si cada agrupamento. Paralelamente, a função de ligação média (*average linkage*) calcula a média das distâncias de todas as amostras de um agrupamento em relação a todas as amostras de outro agrupamento. Em especial, o critério de Ward emprega a distância euclidiana na descoberta do par de agrupamentos que minimizam o aumento na variância total interna após a união.



**Figura 2.6. (a) Representação bidimensional de diferentes critérios de ligação da considerando 6 amostras já alocadas em dois agrupamentos. b) Dendrograma resultante da aplicação do algoritmo de agrupamento hierárquico sobre as amostras 1-6. O algoritmo emprega o método aglomerativo usando critério de ligação única.**

Por outro lado, o método divisivo, e.g. abordagem *top-down*, inicia com uma estrutura plana em que todas as amostras pertencem ao mesmo agrupamento, ou seja, nível hierárquico. Portanto, a cada iteração, o algoritmo divide um ramo-pai em dois subconjuntos menores, os ramos-filhos. O processo termina quando um critério de parada é atingido, frequentemente, o número  $k$  de agrupamentos. No final do algoritmo, é criado um dendrograma de agrupamentos, uma hierarquia de árvore binária [Benavent et al., 2019, de Oliveira et al., 2020, Fahad et al., 2014, Govender e Sivakumar, 2020]. Um possível agrupamento hierárquico considerando a disposição espacial entre as amostras 1-6

da Figura 2.6(a) é ilustrada na Figura 2.6(b). Traçando as retas pontilhadas A-D perpendiculares aos ramos verticais do dendrograma é possível identificar diferentes momentos do processo de agrupamento. Em *A*, nota-se a existência de 6 agrupamentos unitários, ou seja, cada um contendo as amostras. Em *B*, constata-se a existência de 3 agrupamentos: o agrupamento unitário da amostra 1, o agrupamento das amostras 2 e 3, além do agrupamento formado pelas amostras 4, 5 e 6. Em *C*, já é possível identificar o mesmo par de agrupamentos retratados na Figura 2.6(a). Por fim, em *D* verificamos a presença de um único agrupamento superpopuloso, contendo todas as amostras iniciais.

### 2.7.5. Métricas de Avaliação

Independente do algoritmo, supervisionado ou não-supervisionado, caso haja o conhecimento prévio sobre dados rotulados com base em uma verdade básica (*ground truth*), torna-se plausível a clara identificação de quantidade de Verdadeiros Positivos (VP), Falsos Positivos (FP), Verdadeiros Negativos (VN) e Falsos Negativos (FN). Tais classificações compõem o cálculo de várias métricas de recuperação de informação, resumidas na Figura 2.7, como:

- **Acurácia** ( $A_c$ ) é definida pela razão do total de amostras classificadas corretamente (VP + VN), pelo número total de amostras (P+N). Para conjunto de dados não-balanceados, uma avaliação de desempenho baseada exclusivamente nesta métrica pode gerar conclusões erradas;
- **Precisão** ( $P_r$ ) é a razão entre, dada uma classe alvo, a quantidade de amostras corretamente classificadas para a classe em questão (VP), pelo conjunto total de predições atribuídas a essa classe, isto é, corretas e incorretas (VP + FP);
- **Sensibilidade** ( $S_s$ ) também conhecida como revocação (*recall*) ou **taxa de verdadeiros positivos** é definida pela razão entre a quantidade de amostras corretamente preditas (VP) para um classe positiva e o total de amostras que pertencem a esta classe, incluindo assim tanto predições corretas quanto as que deveriam ter indicado esta classe (VP + FN). O análogo para a classe negativa é chamado de **especificidade** ou **taxa de verdadeiros negativos**;
- **Medida- $F_1$**  ( $F_1$ -Score) relaciona a precisão e a sensibilidade por uma média harmônica expressa por

$$Medida - F_1 = \frac{2}{\frac{1}{P_r} + \frac{1}{S_s}}; \quad (9)$$

Geralmente, quando maior o valor da medida- $F_1$ , melhor a classificação sendo um reflexo do compromisso mútuo entre a precisão ( $P_r$ ) a sensibilidade ( $S_s$ ):

- **Área abaixo da curva ROC** é medida através da curva Característica de Operação do Receptor (ROC), mostrada na Figura 2.7(a), uma representação da razão entre a taxa de verdadeiros positivos (TPR) e a taxa de falsos positivos (FPR), para vários limiares de corte. Essa curva descreve graficamente o desempenho de um modelo de classificação. Sucintamente, quanto maior a área abaixo da curva (mais próxima ao valor unitário), melhor o desempenho do modelo, independentemente do ponto de corte da probabilidade de pertencimento de à classe de cada amostra.

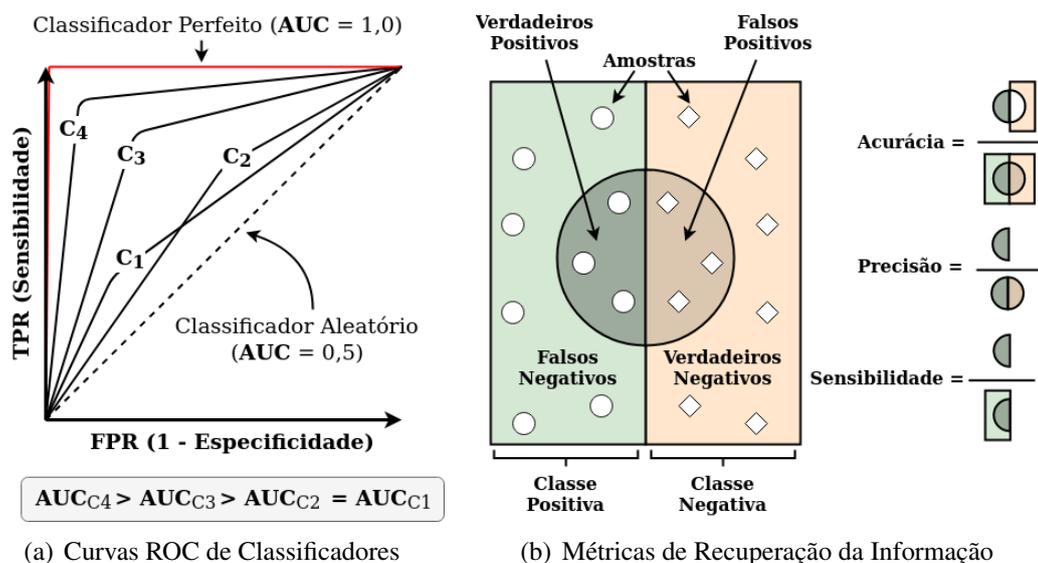


Figura 2.7. (a) Curvas ROC de classificadores e comparação da área abaixo da curva ROC (*Area Under the Curve*, AUC). b) Ilustração das métricas acurácia, precisão e sensibilidade em um problema de classificação binária.

## 2.8. As Soluções para Processamento de Dados em Linguagem Natural em Nuvens Comerciais

Para implementar as diferentes técnicas e algoritmos apresentados em plataformas de computação em nuvem é preciso levar em consideração o tipo de atividade que se espera que o provedor de serviço na nuvem entregue. Os principais provedores de serviço na nuvem públicos são a Amazon Web Service (AWS), a Microsoft Azure e o Google Cloud Platform (GCP). Cada um desses provedores de nuvem oferecem infraestrutura mais simples, como entidades de recursos computacionais análogos a máquinas virtuais, plataformas completas de desenvolvimento, como plataformas que gerenciam a criação de recursos computacionais análogos a máquinas virtuais, até serviços computacionais específicos que não dependem que os usuários gerenciem a infraestrutura de forma alguma, funcionando através de requisições *web* através do protocolo HTTP.

Serviços de inteligência artificial seguem padrão semelhante aos recursos de nuvem em geral, nos quais cada tipo desses recursos tem como público alvo um segmento de profissionais. As plataformas de nuvem dividem a abordagem na qual irão disponibilizar suas ofertas em três camadas: Plataformas de Inteligência Artificial (*Artificial Intelligence Services*), Serviços de Inteligência Artificial (*Artificial Intelligence Platforms*) e Motores de Inteligência Artificial (*Artificial Intelligence Engines*). As próximas subseções detalham cada uma destas três camadas, suas peculiaridades, precificação e público alvo.

### 2.8.1. Plataformas de Inteligência Artificial

As plataformas de inteligência artificial <sup>13</sup>, chamadas de *AI Platforms*, estão relacionados com a utilização de alguma solução que facilita o ciclo de vida de uma aplicação

<sup>13</sup>Disponíveis em <https://aws.amazon.com/pt/machine-learning/ai-services/>.

de inteligência artificial. As *AI Platforms* oferecem o gerenciamento de infraestrutura necessário para treinar e disponibilizar modelos de inteligência artificial e também para fazer a engenharia de características (*feature engineering*) dos dados e lidar com previsões em tempo real utilizando algoritmos implementados em estrutura de computação distribuída, como oferecidas pelo arcabouço Apache Spark em sua biblioteca `MLlib`<sup>14</sup>. Esses serviços exigem conhecimento prévio dos algoritmos de inteligência artificial, visto que eles devem ser implementados e o modelo deve ser treinado com dados fornecidos. As *AI Platforms* apoiam esse processo, oferecendo recursos para gerenciar o treinamento e a implantação dos modelos, nos quais a implantação de um modelo treinada é realizada em menos de 5 minutos<sup>15</sup>.

No caso específico de uma solução de extração de entidades de um texto, o usuário deverá escolher o algoritmo adequado para solucionar o problema em questão, usar a plataforma oferecida pelas *AI Platforms* para auxiliar na ingestão de dados de forma segura, transformação dos dados e engenharia de características, utilizar as ferramentas para controle e gerência do treinamento dos dados, que pode ser realizada com uma estrutura de recursos computacionais totalmente apartada, com CPU e GPU dedicadas para aquele treinamento, e, por fim, utilizar o serviço que recebe o modelo treinado e cria a infraestrutura juntamente com o servidor HTTP responsável por receber as requisições e retornar as previsões.

Os provedores precificam esses serviços com base na quantidade de recurso computacional utilizadas por hora e pela quantidade de armazenamento utilizada, em que alguns deles, como o Amazon Sagemaker na AWS<sup>16</sup>, costumam cobrar uma taxa a mais pelo uso da plataforma em si. Esse modelo de cobrança otimiza o uso de recursos, tendo em vista a possibilidade da utilização de recursos computacionais como GPUs em um modelo de aluguel por hora, não necessitando de investimento inicial.

Serviços de *AI Platforms* são recomendados para cientistas de dados e engenheiro de aprendizado de máquina que necessitam implantar aplicações específicas implementando o ciclo completo de desenvolvimento dos modelos, desde a extração dos dados até o monitoramento do modelo final implantado.

### 2.8.2. Serviços de Inteligência Artificial

Os serviços de inteligência artificial<sup>17</sup>, chamados de *AI Services*, estão relacionados com a utilização de alguma solução de inteligência artificial implementada previamente pelo provedor de serviço em nuvem. Essas soluções são baseadas em soluções de problemas específicos, como a transformação de texto em voz ou para realizar processamento de linguagem natural extraindo entidades de textos. Esses serviços não exigem conhecimento prévio dos algoritmos de inteligência artificial. Entre os serviços mais comuns estão a automatização de revisões de códigos, a criação de *chatbots*, prevenção de fraudes, traduções em tempo real, transcrição e sintetização de fala, entre outros.

---

<sup>14</sup>Disponível em <https://spark.apache.org/mllib/>.

<sup>15</sup>A implantação do modelo treinado se refere ao processo de construção de uma aplicação que utiliza o modelo treinado, recebendo requisições HTTP com os dados de entrada e retornando o resultado da predição do modelo

<sup>16</sup>Disponível em <https://aws.amazon.com/pt/sagemaker/>.

<sup>17</sup>Disponíveis em <https://aws.amazon.com/pt/machine-learning/ai-services/>.

Essa categoria de serviços não exige que os usuários treinem os modelos. A utilização é baseada em requisições através do protocolo HTTP, nas quais o usuário envia os dados específicos para a predição e o provedor de serviços na nuvem executa a predição e retorna o resultado. No caso específico de uma solução de extração de entidades de um texto, o usuário deve enviar o texto para o provedor, o provedor analisará o texto com seu modelo proprietário e retornará as entidades encontradas. Por questão de privacidade dos dados, os provedores não armazenam os dados fornecidos pelos usuários ao realizar as predições.

Os provedores precificam esses serviços com base na quantidade de dados que são utilizados como entrada para as predições e a quantidade de requisições feitas pelos usuários. Essa relação de cobrança acarreta que o custo de execução pode ficar mais elevado do que se o usuário realizasse as predições nos próprios modelos. Contudo, o custo é compensado pelo fato do usuário não precisar criar, treinar e gerenciar esses modelos.

Os *AI Services* são recomendados quando o usuário não tem experiência implementando seus próprios algoritmos de inteligência artificial ou para resolução de problemas clássicos e específicos que o provedor já forneça solução pronta que satisfaz os requisitos para a resolução do problema. Os *AI Services* levam a resultados imediatos mediante as requisições *web* através do protocolo HTTP.

### **2.8.3. Motores de Inteligência Artificial**

Os motores de inteligência artificial, chamados de *AI Engines*, são a camada dedicada ao uso direto, sem intermédio de uma solução do provedor de serviço em nuvem, de arcabouços de código aberto, como Apache MXNet, Tensorflow e Torch, provendo flexibilidade total aos cientistas de dados e engenheiros de aprendizado de máquina para testar novas implementações de algoritmos, sistemas mais sofisticados que exigem algum recurso de mais baixo nível, como o uso de C e C++. Em geral, é a escolha de pesquisadores que estão implementando um novo modelo otimizado e precisam de total liberdade do sistema operacional, não dependendo de nenhuma implementação prévia do provedor de serviços em nuvem.

Esses serviços funcionam com o provisionamento da infraestrutura requerida pelo usuário para execução do desenvolvimento, treinamento e implantação do modelo. Recursos de GPU, CPU, chegando até mesmo ao nível mais específico de *Field-Programmable Gate Arrays* (FPGAs). Nessa categoria de serviço os provedores de serviço em nuvem alugam recursos computacionais análogos a máquinas virtuais para os usuários.

Os provedores precificam esses serviços analogamente às *AI Platform*, exceto pelo custo extra de licença para a utilização de algumas plataformas em particular. *AI Platforms* são recomendadas para cientistas de dados e engenheiro de aprendizado de máquina que precisam escrever aplicações do zero, com total liberdade de escolha de qualquer tipo de arcabouço e linguagem de programação, com a desvantagem de ter que gerenciar a infraestrutura total requisitada, enquanto *AI Engines* são recomendados para usuários que possam utilizar os arcabouços específicos já implantados na nuvem.

## 2.9. As Iniciativas de Pesquisa

Diversas atividades de pesquisa estão ativas e buscam caracterizar e mitigar os desafios causados pelas notícias falsas. Uma definição inicial sobre as notícias falsas é feita por Lazer *et al.*. No artigo, é abordada a história das notícias falsas começando pela difamação na Primeira Guerra Mundial até o impacto das notícias falsas durante a eleição presidencial dos Estados Unidos em 2016 [Lazer et al., 2018]. Grinberg *et al.* aprofundam no impacto das notícias falsas durante as eleições de 2016, analisando as mensagens da rede social *Twitter* [Grinberg et al., 2019]. Os autores coletaram *tweets* enviados por 16.442 contas ativas durante a temporada eleitoral de 2016, de 1º de agosto até 6 de dezembro de 2016. Os resultados mostram que os grupos de maior idade, entre 60 e 80 anos, com afinidade política de direita ou extrema direita são mais propensos à distribuição e ao compartilhamento de notícias políticas falsas. A recente pandemia da Doença Infecciosa por Corona Vírus de 2019 (COVID-19) também é um evento no qual foram disseminadas grande quantidade de notícias falsas. Estudos recentes mostram a correlação entre o uso de mídia social e a desinformação durante a pandemia [Pennycook et al., 2020, Van Bavel et al., 2020].

A detecção de notícias falsas é estudada sob várias perspectivas como Aprendizado de Máquina, Mineração de Dados e Processamento de Linguagem Natural. O Saco-de-Palavras e as frequências de categorias são utilizadas para o treinamento de classificadores como as Máquinas de Vetores Suportes (*Support Vector Machines* - SVM) e modelos bayesianos ingênuos [Poddar et al., 2019]. Uma vez que o modelo matemático é treinado a partir de exemplos conhecidos das duas categorias, notícia falsa ou não, é possível prever instâncias futuras com base em agrupamento numérico e distâncias. O uso de diferentes métodos de agrupamento e funções de distância entre os pontos de dados é uma das bases do algoritmo do SVM. Por outro lado, o algoritmo bayesiano ingênuo faz classificações com base em evidências acumuladas da correlação entre uma determinada variável, como a sintaxe, e as outras variáveis presentes no modelo.

Shu *et al.* fazem uma revisão da detecção de notícias falsas nas mídias sociais de uma perspectiva de mineração de dados, incluindo caracterização de notícias falsas sobre psicologia e teorias sociais, algoritmos existentes, métricas de avaliação e conjuntos de dados representativos [Shu et al., 2017]. *Fake News Tracker* é uma solução para coleta de dados, visualização interativa e modelagem analítica para detecção de notícias falsas. A solução utiliza técnicas de Processamento de Linguagem Natural [Shu et al., 2019].

Alguns trabalhos apresentam técnicas e desafios sobre a detecção de notícias falsas. Zhou e Zafarani identificam e detalham as teorias fundamentais relacionadas em diferentes disciplinas para a detecção de notícias falsas [Zhou e Zafarani, 2018]. Sharma *et al.* discutem os métodos e técnicas existentes aplicáveis à identificação e à mitigação de notícias falsas, com foco nos avanços significativos em cada método e suas vantagens e limitações [Sharma et al., 2019]. Bondielli e Marcelloni fazem um levantamento da literatura sobre as diferentes abordagens para a detecção automática de notícias falsas e rumores [Bondielli e Marcelloni, 2019]. Os autores destacam várias abordagens adotadas para coletar dados de notícias falsas e rumores.

Oshikawa *et al.* apresentam uma comparação dos métodos usados na detecção de notícias falsas usando Processamento de Linguagem Natural (PLN) [Oshikawa et al.,

2018]. De forma semelhante, Sharma *et al.* analisam a revisão da literatura sobre PLN aplicado em notícias falsas, ressaltando a comparação entre as diferentes técnicas de aprendizado de máquina, aprendizado profundo e outras técnicas [Sharma et al., 2019]. Deepak and Chitturi compararam diferentes tipos de redes neurais na detecção de notícias falsas [Deepak e Chitturi, 2020]. Feng *et al.* propõem uma rede neural convolucional de dois níveis com gerador de resposta do usuário, em que a rede neural captura informações semânticas do texto, representando-as no nível de frase e de palavra, e o gerador de resposta de usuário aprende um modelo da resposta do usuário ao texto da notícia [Qian et al., 2018].

## 2.10. Desafios e Oportunidades de Pesquisa

Embora as pesquisas na identificação, detecção e mitigação da propagação de notícias falsas estejam em pleno desenvolvimento, alguns dos principais desafios no combate às notícias falsas são listados a seguir [Sharma et al., 2019].

- **Grandes interesses e a pluralidade de atores envolvidos.** Devido ao volume que a propagação de notícias falsas atinge em redes sociais em um período curto, as notícias falsas representam uma ameaça às fontes tradicionais de informações, como a imprensa tradicional. O espalhamento de notícias falsas ocorre como um evento distribuído e, então, envolve múltiplas entidades e plataformas tecnológicas. Assim, há uma crescente dificuldade de estudar e projetar estratégias computacionais, tecnológicas e de negócios de combate às notícias falsas sem que haja o comprometimento da rapidez e do acesso colaborativo a informações de alta qualidade.
- **Intensão maliciosa do adversário.** O conteúdo das notícias falsas é projetado para dificultar a identificação por humanos das notícias falsas, explorando suas habilidades cognitivas, emoções e preconceitos ideológicos. Além disso, é desafiador para métodos computacionais detectar notícias falsas, pois a forma como as notícias falsas são apresentadas é semelhante à de notícias verdadeiras e, por vezes, as notícias falsas usam artifícios para dificultar a identificação da fonte ou falsificam a verdadeira fonte da notícia.
- **Suscetibilidade e falta de conscientização do público.** O usuário de redes sociais está sujeito a uma grande quantidade de informações de origens duvidosas, desde informações com cunho humorístico, como sátiras, até informações com o intuito de enganar o consumidor de informações se passando por notícias verdadeiras. Contudo, o usuário de redes sociais não é capaz de diferenciar uma notícia falsa de uma verdadeira apenas pelo conteúdo. O usuário não dispõe de informações sobre a credibilidade da fonte ou padrões de propagação da notícia na rede. Assim, para aumentar a conscientização pública, vários artigos e campanhas publicitárias são veiculados para fornecerem dicas sobre como diferenciar notícias verdadeiras de falsas. Por exemplo, a Universidade de Portland, nos Estados Unidos, disponibiliza um guia para a identificação de desinformação (notícias falsas)<sup>18</sup>.

---

<sup>18</sup>Disponível em <https://guides.library.pdx.edu/c.php?g=625347&p=4359724>.

- **Dinâmica de propagação.** A propagação de notícias falsas em mídia social complica a detecção e a mitigação, pois as informações falsas podem facilmente alcançar e afetar um grande número de usuários em pouco tempo. A informação é transmitida de maneira rápida e fácil, mesmo quando sua veracidade é duvidosa [Friggeri et al., 2014]. A verificação da veracidade deve ser realizada de forma ágil, mas também deve considerar os padrões de propagação da informação ao longo da rede [Meel e Vishwakarma, 2020].
- **Mudanças constante das características das notícias falsas.** Os desenvolvimentos na identificação automatizada de notícias falsas também impulsionam a adaptação da geração de novos conteúdos de desinformação para evitarem de serem classificados como tal. A detecção de notícias falsas baseada em estilo de escrita, diferenciando notícias falsas e verdadeiras por uma análise baseada no processamento de linguagem natural, é uma das principais alternativas usadas devido aos desafios não resolvidos na automatização da verificação de fatos a partir de bases de conhecimento pré-definidas. Assim, abordagens atuais de identificação de notícias falsas baseadas no conteúdo focam na extração de fatos diretamente do conteúdo da notícia e a posterior verificação dos fatos contra bases de conhecimento [de Oliveira et al., 2020].
- **Ataques ao aprendizado por linguagem natural.** Zhou *et al.* argumentam que o uso de processamento de linguagem natural para a identificação de notícias falsas é vulnerável a ataques ao aprendizado de máquina em si [Zhou et al., 2019]. Zhou *et al.* identificam três ataques: a distorção de fatos, a troca entre sujeito e objeto; e a confusão de causas. A distorção de fato consiste em exagerar ou modificar algumas palavras. Elementos textuais, como personagens e tempo, podem ser distorcidos para levar a uma interpretação falsa. A troca entre sujeito e objeto tem como objetivo confundir o leitor entre quem pratica e quem sofre a ação relatada. O ataque de confusão de causa consiste em criar relações causais inexistentes entre dois eventos independentes ou cortar partes de uma história, deixando apenas as partes que o atacante deseja apresentar para o leitor [Zhou et al., 2019].

As oportunidades de pesquisa na identificação e mitigação de notícias falsas focam na detecção rápida ou em tempo real da fonte, no controle da propagação das informações falsas e na redução do impacto das notícias falsas na sociedade. Conjuntos de dados coletados em tempo real, detecção automática de rumores e localização da fonte original são questões de pesquisa desafiadoras [Meel e Vishwakarma, 2020]. A seguir destacam-se as principais oportunidades de pesquisa e desenvolvimento de soluções para o combate às notícias falsas.

- **Extração de características mais significativas.** Determinar as características mais eficazes para detectar notícias falsas de múltiplas fontes de dados é uma oportunidade de pesquisa em aberto. Fundamentalmente, existem duas fontes de dados principais: o conteúdo das notícias e contexto social [Shu et al., 2017]. Da perspectiva de conteúdo de notícias, técnicas baseadas em processamento de linguagem natural e extração de características podem ser usadas para extrair informações do

texto. Técnicas de incorporação, como incorporação de palavras (*word embedding*) e redes neurais profundas são foco de pesquisas atuais para a extração de características textuais e têm o potencial para aprender melhores representações para os dados. Características visuais extraídas das imagens também são indicadores importantes para notícias falsas. O uso de redes neurais profundas é uma oportunidade de pesquisa na extração de características visuais para a detecção de notícias falsas [Sharma et al., 2019, Meel e Vishwakarma, 2020].

- **Detecção em diferentes plataformas e diferentes domínios.** Devido ao fato dos usuários utilizarem diferentes redes sociais, as notícias falsas e boatos se espalham nas diferentes plataformas, dificultando a localização da origem da notícia ou do boato. O rastreamento da origem da informação falsa entre plataformas distintas de redes sociais é uma oportunidade de pesquisa. Para tanto, devem ser considerados diversos aspectos da informação. Contudo, a maior parte da abordagem existente se concentra apenas em uma das formas de detecção da informação falsa: análise de conteúdo, da propagação, do estilo, entre outras. A análise deve considerar, então, diferentes domínios de atributos, como tópicos, sítios *web*, imagens e URLs [Meel e Vishwakarma, 2020].
- **Identificação de câmaras de eco e ponte entre as câmaras.** A mídia social tende a formar câmaras de eco em comunidades em que usuário têm visões e ideologias semelhantes. Os usuários têm suas visões reforçadas e não estão cientes das crenças opostas. Portanto, pesquisas são necessárias para identificar câmaras de eco conflitantes e ligar as câmaras com posições opostas para que os usuários sejam confrontados com visões distintas. Isso também ajuda na descoberta da verdade, fazendo os usuários pensarem criteriosamente e racionalmente em múltiplas dimensões [Meel e Vishwakarma, 2020].
- **Desenvolvimento de modelos de aprendizado de máquina.** Há a necessidade de pesquisa no desenvolvimento de modelos de aprendizado em tempo real, tais como aprendizado incremental e aprendizado federado, capazes de aprender com artigos verificados manualmente e fornecer detecção em tempo real de novos artigos com informações fraudulentas. Outro ponto importante é o desenvolvimento de modelos não-supervisionados em que os algoritmos aprendem com dados reais e, então, artigos que fogem do comportamento de dados reais são classificados como falsos. Há ainda uma escassez de conjuntos de dados específicos para notícias falsas. A falta de conjuntos de dados de larga escala publicamente disponíveis implica a carência de testes (*benchmarks*) para a comparação de desempenho entre algoritmos diferentes [Meel e Vishwakarma, 2020].
- **Desenvolvimento de estruturas de dados capazes de lidar com a estrutura de rede complexa e dinâmica.** A complexidade e a dinamicidade das estruturas de relacionamento em redes sociais tornam a tarefa de identificação e rastreamento de publicações mais complicadas. Assim, há a necessidade de pesquisa para o desenvolvimento de estruturas de dados complexas que reflitam a dinamicidade das relações em redes sociais para permitir a extração de conhecimento acerca da propagação de informações falsas na rede [Meel e Vishwakarma, 2020].

## 2.11. Atividade Prática

Esta seção consolida a uma pluralidade de conceitos teóricos abordados no capítulo através de uma atividade prática do processo de identificação de notícias falsas em redes sociais, empregando a linguagem `Python`. O processo ocorre na rede social *Twitter* e é representado na Figura 2.11. O processo inclui (i) a coleta de notícias, falsas e verdadeiras, empregando a interface de programação de aplicação (*Application Programming Interface* - API) do *Twitter* para efetuar o *web scraping*<sup>19</sup>; (ii) o processamento textual e vetorização do conteúdo das notícias, empregando PLN juntamente com técnicas de representação vetorial de textos; (iii) a aplicação eficiente de algoritmos de detecção, alcançado pela incorporação de técnicas de redução de dimensionalidade; e (iv) a avaliação da eficiência e qualidade da detecção, tendo como parâmetros as métricas de recuperação de informação.

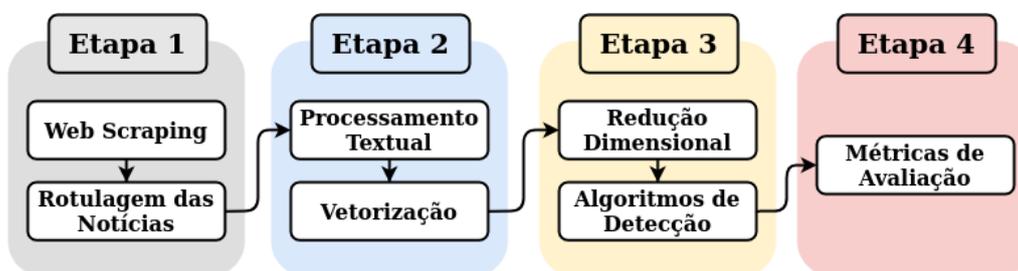


Figura 2.8. Fluxograma do processo de identificação de notícias falsas desenvolvido na atividade prática. A primeira etapa compreende a formação da base de dados formada por notícias falsas e legítimas. Na segunda etapa são aplicadas técnicas de processamento de linguagem natural e vetorização. Na terceira etapa, após uma redução dimensional, a representação vetorial das notícias é submetida a algoritmos de detecção. A quarta etapa concentra-se em avaliar a qualidade da detecção segundo métricas de recuperação de informação.

Como primeira etapa da atividade prática, a composição da base de dados inclui tanto notícias verdadeiras quanto falsas, extraídas através do *web scraping* em contas específicas do *Twitter*. Para obtenção do conteúdo dessas contas, é preciso desenvolver um *script*<sup>20</sup> em `Python` que acessa a API do *Twitter* usando credenciais de desenvolvedor. Em posse dessas credenciais, a biblioteca `tweepy`<sup>21</sup> permite a extração contínua do conteúdo textual dos *tweets* de qualquer perfil aberto na rede social. Contudo, além das limitações temporais igualmente enfrentadas por Barreto *et al.*, como o número máximo de requisições por janela de tempo de 15 minutos [Barreto *et al.*, 2014], há também uma limitação da quantidade de *tweets* históricos passíveis de serem coletados. Dessa maneira, a obtenção de *tweets* é restrita a um período de até, no máximo, dois meses passados a contar pela data de execução do *script*. Devido às restrições de acesso à plataforma, uma solução é diversificar as fontes de busca por notícias verdadeiras, coletando *tweets* de outras fontes jornalísticas. A escolha de perfis de veículos jornalísticos como fonte de conteúdo verdadeiro parte da premissa que estes perfis são menos susceptíveis

<sup>19</sup>Também conhecida como coleta, ou raspagem de dados, é uma forma de mineração capaz de extrair o conteúdo de sítios da *web* para uma posterior análise.

<sup>20</sup>Disponível em <https://github.com/nicollasro/FakeNewsDetection>.

<sup>21</sup>Disponível em <https://www.tweepy.org/>.

a compartilhar conteúdo de procedência duvidosa do que contas de usuários individuais. Analogamente, a coleta de *tweets* comprovadamente falsos pode ser feita extraindo o conteúdo de perfis dedicados checagem de fatos. Nesses perfis é possível encontrar *tweets* falsos previamente verificados por jornalistas.

Os *tweets* extraídos, uma vez armazenados e rotulados entre falso e verdadeiro, são submetidos à sequência básica de processamento de linguagem natural descrita na Seção 2.5. Em Python, várias técnicas de PLN são facilmente implementáveis por funções da biblioteca NLTK<sup>22</sup>. Nos procedimentos seguintes, diversas funções e classes de módulos específicos da biblioteca Scikit-learn<sup>23</sup> serão empregadas. Em especial, na vetorização usa-se o módulo FeatureExtraction, que inclui a classe TfidfVectorizer capaz de converter a coleção de *tweets* já processados textualmente em uma matriz contendo os valores TF-IDF de cada palavra. Diante da alta dimensionalidade da matriz TF-IDF adquirida, torna-se conveniente empregar o módulo Decomposition, que dispõe de diferentes algoritmos de decomposição matricial predominantemente usados na redução dimensional. Devido ao caráter esparso da matriz TF-IDF, a etapa de redução dimensional da atividade prática é desempenhada pelas funções da classe TruncatedSVD, que executam a decomposição em valores singulares truncada, (*Singular Value Decomposition* - SVD), também conhecida como Indexação Semântica Latente (*Latent Semantic Indexing* - LSI). Como alertado na Seção 2.7.1, a configuração do nível de aproximação  $k$ , representado na classe pelo parâmetro `n_components`, precisa ser cuidadosamente escolhido observando o atributo `explained_variance_ratio_`, que expressa o percentual de variância entre as componentes geradas e as originais.

Após obter uma representação vetorial eficientemente reduzida dos *tweets* extraídos, é possível aplicar três exemplos de metodologias diferentes [de Oliveira et al., 2020], capazes de detectar padrões de escrita característicos de notícias falsas. A primeira metodologia, chamada Redução com Treinamento, prevê o treinamento e classificação dos *tweets* coletados usando o algoritmo Máquina de Vetor de Suporte de Classe Única (*One-class Support Vector Machine*), implementado na classe `OneClassSVM` do módulo SVM. A segunda metodologia, chamada Transformação Matricial, introduz uma transformação matricial antes do processo de treinamento com o algoritmo SVM de classe única. Tal transformação é produzida multiplicando a matriz TF-IDF reduzida por uma versão transposta da mesma, porém submetida ao algoritmo de agrupamento *k-means*. A aplicação desse algoritmo no *script* em Python depende do uso da classe `KMeans` do módulo Cluster. A terceira metodologia, denominada Limite Radial, expande o processo de detecção para um cenário estatístico, partindo da hipótese de que notícias verdadeiras e falsas têm distribuição de probabilidades distintas. A última etapa da atividade prática consiste na avaliação da qualidade da detecção de notícias falsas a partir das métricas de recuperação da informação descritas na Seção 2.7.5. Incorporando ao *script* desenvolvido algumas funções do módulo Metrics, tais como `accuracy_score`, `precision_score`, `recall_score` e `roc_curve`, é possível obter respectivamente os valores de acurácia, precisão, sensibilidade e da curva ROC. Ao final da atividade prática é esperado que os resultados se assemelhem aos das Figuras 2.9 e 2.10.

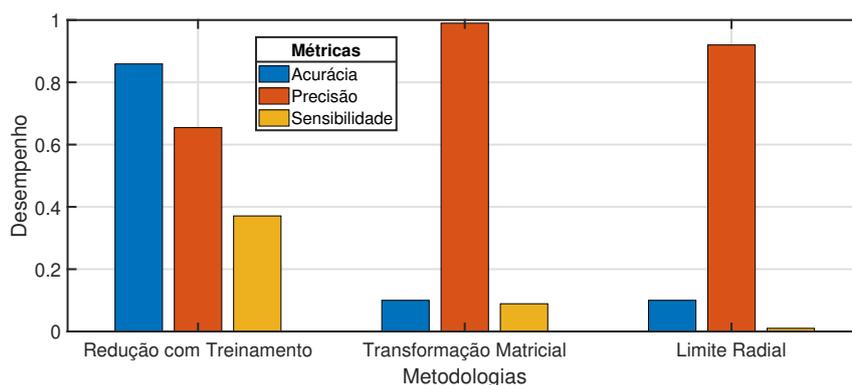
---

<sup>22</sup>Disponível em <https://www.nltk.org/>.

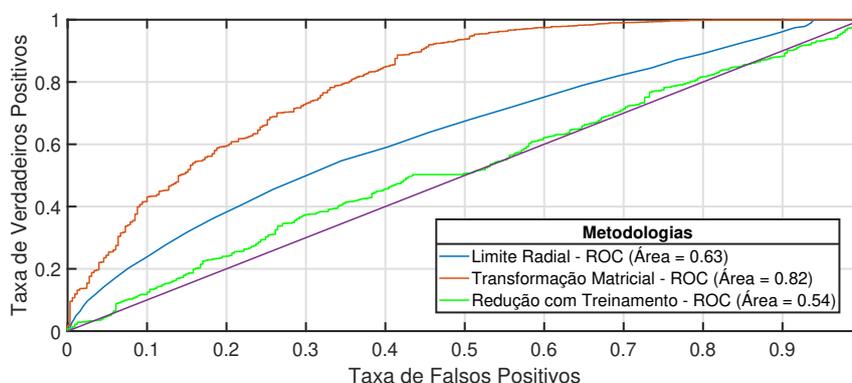
<sup>23</sup>Disponível em <https://scikit-learn.org/stable/>.

Os resultados da primeira metodologia, Redução com Treinamento, demonstram um desempenho mais homogêneo entre as métricas, destacando-se principalmente pela alta acurácia e porcentagem de sensibilidade mais expressiva dentre as três metodologias. Já nos resultados referentes à segunda metodologia, Transformação Matricial, percebe-se uma clara predominância na habilidade de classificar qualitativamente notícias como sendo falsas, expressa pela alta precisão. Em contrapartida, detém baixas porcentagens de acurácia e sensibilidade, essas possivelmente fruto de perdas de características, importantes na diferenciação das notícias, impostas por dois níveis de redução dimensional – LSI e *k-means*. Analogamente à anterior, a terceira metodologia, Limite Radial, apresenta o mesmo caráter preciso na identificação de notícias falsas embora constata-se uma depreciação nos seus níveis de sensibilidade.

A Figura 2.10 apresenta uma comparação entre as curvas ROC de cada metodologia de detecção. O bom resultado obtido pela metodologia de transformação matricial, área abaixo da curva de 0,82, baseia-se no fato de que ao realizar o agrupamento com o *k-means*, a dimensionalidade dos dados é substancialmente reduzida, permitindo que a SVM de classe única defina uma hiper-superfície mais ajustada aos dados.



**Figura 2.9.** Comparação das metodologias revela um comportamento diverso porém ligeiramente complementar nos níveis de recuperação de informação.



**Figura 2.10.** As curvas ROC refletem o desempenho de um sistema classificador binário à medida que o seu limiar de discriminação varia. Dentre as metodologias, a transformação matricial apresenta o melhor desempenho, visto que possui a maior área acima da reta.

## 2.12. Considerações Finais

Nesse minicurso foram apresentados as definições, características e o processo de disseminação de notícias falsas. Em seguida, foram discutidos os métodos tradicionais para a detecção de notícias falsas. Foram comparados os conjuntos de dados de referência mais recentes. Com base em trabalhos da literatura, foi proposto a utilização do Processamento de Linguagem Natural (PLN) na de detecção de notícias falsas. Foram mostrados como o PLN pode ser usado sobre redes sociais e uma comparação com os diferentes métodos de aprendizado de máquina utilizados. Além disso, questões em aberto e desafios também são destacados para explorar as oportunidades de pesquisa em potencial. Esse trabalho é útil para os pesquisadores compreendam os diferentes componentes da comunicação digital *online* de uma perspectiva social e técnica. Divulgação de notícias falsas em várias plataformas multilíngues, estrutura de rede complexa e dinâmica, grandes volumes de dados em tempo real não rotulados e detecção precoce de boatos são alguns problemas desafiadores que ainda não foram resolvidos e necessitam mais pesquisas. Finalmente, a atividade prática desenvolvida mostrar a viabilidade na detecção de notícias falsas. Melhorar a confiabilidade e o futuro do ecossistema de informações *online* é uma responsabilidade conjunta da comunidade científica, formuladores de políticas digitais, administração e da sociedade em geral.

## Referências

- [Afroz et al., 2012] Afroz, S., Brennan, M. e Greenstadt, R. (2012). Detecting hoaxes, frauds, and deception in writing style online. Em *2012 IEEE Symposium on Security and Privacy*, p. 461–475. IEEE.
- [Andreoni Lopez et al., 2019] Andreoni Lopez, M., Mattos, D. M. F., Duarte, O. C. M. B. e Pujolle, G. (2019). A fast unsupervised preprocessing method for network monitoring. *Annals of Telecommunications*, 74(3):139–155.
- [Ayodele, 2010] Ayodele, T. O. (2010). Types of machine learning algorithms. Em *New advances in machine learning*. IntechOpen.
- [Balage Filho et al., 2013] Balage Filho, P., Pardo, T. A. S. e Aluísio, S. (2013). An evaluation of the brazilian portuguese liwc dictionary for sentiment analysis. Em *Proceedings of the 9th Brazilian Symposium in Information and Human Language Technology*.
- [Barreto et al., 2014] Barreto, H. F., Campista, M. E. M. e Costa, L. H. M. (2014). Spammers no twitter: Quando contatos deixam de ser bem-vindos. Em *Workshop de Redes P2P, Dinâmicas, Sociais e Orientadas a Conteúdo (Wp2p+ 2014) - SBRC 2014*, volume 1, p. 23–36.
- [Benavent et al., 2019] Benavent, X., Castellanos, A., de Ves, E., Garcia-Serrano, A. e Cigarran, J. (2019). Fca-based knowledge representation and local generalized linear models to address relevance and diversity in diverse social images. *Future Generation Computer Systems*, 100:250–265.
- [Bird et al., 2009] Bird, S., Klein, E. e Loper, E. (2009). *Natural Language Processing with Python*. O’Reilly Media, Inc., 1st edição.

- [Bondielli e Marcelloni, 2019] Bondielli, A. e Marcelloni, F. (2019). A survey on fake news and rumour detection techniques. *Information Sciences*, 497:38 – 55.
- [Boutaba et al., 2018] Boutaba, R., Salahuddin, M. A., Limam, N., Ayoubi, S., Shahriar, N., Estrada-Solano, F. e Caicedo, O. M. (2018). A comprehensive survey on machine learning for networking: evolution, applications and research opportunities. *Journal of Internet Services and Applications*, 9(1):16.
- [Camacho-Collados e Pilehvar, 2018] Camacho-Collados, J. e Pilehvar, M. T. (2018). From word to sense embeddings: A survey on vector representations of meaning. *Journal of Artificial Intelligence Research*, 63:743–788.
- [Chandrashekar e Sahin, 2014] Chandrashekar, G. e Sahin, F. (2014). A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28.
- [Chen et al., 2015] Chen, Y., Conroy, N. J. e Rubin, V. L. (2015). Misleading online content: Recognizing clickbait as false news. Em *Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection*, p. 15–19. ACM.
- [Clark et al., 2012] Clark, M., Kim, Y., Kruschwitz, U., Song, D., Albakour, D., Dignum, S., Beresi, U. C., Fasli, M. e De Roeck, A. (2012). Automatically structuring domain knowledge from text: An overview of current research. *Information Processing & Management*, 48(3):552–568.
- [Davis et al., 2016] Davis, C. A., Varol, O., Ferrara, E., Flammini, A. e Menczer, F. (2016). Botornot: A system to evaluate social bots. Em *Proceedings of the 25th International Conference Companion on World Wide Web, WWW '16 Companion*, p. 273–274, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.
- [de Oliveira et al., 2020] de Oliveira, N. R., de Medeiros, D. S. V. e Mattos, D. M. F. (2020). Syntactic-relationship approach to construct well-informative knowledge graphs representation. Em *4th Cloud and Internet of Things - CIoT'20 (a ser apresentado)*. IEEE.
- [de Oliveira et al., 2020] de Oliveira, N. R., Medeiros, D. S. V. e Mattos, D. M. F. (2020). A sensitive stylistic approach to identify fake news on social networking. *IEEE Signal Processing Letters*, 27:1250–1254.
- [de Oliveira et al., 2020] de Oliveira, N. R., Reis, L. H., Fernandes, N. C., Bastos, C. A. M., de Medeiros, D. S. V. e Mattos, D. M. F. (2020). Natural language processing characterization of recurring calls in public security services. Em *Proceedings of the 2020 International Conference on Computing, Networking and Communications (ICNC)*, p. 1009–1013. IEEE.
- [Deepak e Chitturi, 2020] Deepak, S. e Chitturi, B. (2020). Deep neural approach to fake-news identification. *Procedia Computer Science*, 167:2236 – 2243. International Conference on Computational Intelligence and Data Science.

- [Deerwester et al., 1990] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K. e Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407.
- [Domingos, 2012] Domingos, P. M. (2012). A few useful things to know about machine learning. *ACM Commun.*, 55(10):78–87.
- [Fahad et al., 2014] Fahad, A., Alshatri, N., Tari, Z., Alamri, A., Khalil, I., Zomaya, A. Y., Foufou, S. e Bouras, A. (2014). A survey of clustering algorithms for big data: Taxonomy and empirical analysis. *IEEE transactions on emerging topics in computing*, 2(3):267–279.
- [Ferreira e Vlachos, 2016] Ferreira, W. e Vlachos, A. (2016). Emergent: a novel dataset for stance classification. Em *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*, p. 1163–1168.
- [Firth, 1957] Firth, J. R. (1957). A synopsis of linguistic theory, 1930-1955. *Studies in Linguistic Analysis*.
- [Friggeri et al., 2014] Friggeri, A., Adamic, L., Eckles, D. e Cheng, J. (2014). Rumor cascades.
- [Fuller et al., 2009] Fuller, C. M., Biros, D. P. e Wilson, R. L. (2009). Decision support for determining veracity via linguistic-based cues. *Decision Support Systems*, 46(3):695 – 703. Wireless in the Healthcare.
- [Gabiolkov et al., 2016] Gabiolkov, M., Ramachandran, A., Chaintreau, A. e Legout, A. (2016). Social Clicks: What and Who Gets Read on Twitter? Em *ACM SIGMETRICS / IFIP Performance 2016*, Antibes Juan-les-Pins, France.
- [Gan e Tao, 2015] Gan, J. e Tao, Y. (2015). Dbscan revisited: Mis-claim, un-fixability, and approximation. Em *Proceedings of the 2015 ACM SIGMOD international conference on management of data*, p. 519–530.
- [Gaonkar et al., 2019] Gaonkar, S., Itagi, S., Chalippatt, R., Gaonkar, A., Aswale, S. e Shetgaonkar, P. (2019). Detection of online fake news : A survey. Em *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)*, p. 1–6.
- [Golbeck et al., 2018] Golbeck, J., Mauriello, M., Auxier, B., Bhanushali, K. H., Bonk, C., Bouzaghane, M. A., Buntain, C., Chanduka, R., Cheakalos, P., Everett, J. B., Falak, W., Gieringer, C., Graney, J., Hoffman, K. M., Huth, L., Ma, Z., Jha, M., Khan, M., Kori, V., Lewis, E., Mirano, G., Mohn IV, W. T., Mussenden, S., Nelson, T. M., Mcwillie, S., Pant, A., Shetye, P., Shrestha, R., Steinheimer, A., Subramanian, A. e Visnansky, G. (2018). Fake news vs satire: A dataset and analysis. *WebSci '18*, p. 17–21, New York, NY, USA. Association for Computing Machinery.

- [Govender e Sivakumar, 2020] Govender, P. e Sivakumar, V. (2020). Application of k-means and hierarchical clustering techniques for analysis of air pollution: A review (1980–2019). *Atmospheric Pollution Research*, 11(1):40–56.
- [Grinberg et al., 2019] Grinberg, N., Joseph, K., Friedland, L., Swire-Thompson, B. e Lazer, D. (2019). Fake news on twitter during the 2016 us presidential election. *Science*, 363(6425):374–378.
- [Hauch et al., 2015] Hauch, V., Blandón-Gitlin, I., Masip, J. e Sporer, S. L. (2015). Are computers effective lie detectors? a meta-analysis of linguistic cues to deception. *Personality and social psychology Review*, 19(4):307–342.
- [Hu et al., 2016] Hu, B., Tang, B., Chen, Q. e Kang, L. (2016). A novel word embedding learning model using the dissociation between nouns and verbs. *Neurocomputing*, 171:1108–1117.
- [Indurkha e Damerau, 2010] Indurkha, N. e Damerau, F. J. (2010). *Handbook of Natural Language Processing*. Chapman & Hall/CRC, 2nd edição.
- [Jarmasz e Szpakowicz, 2003] Jarmasz, M. e Szpakowicz, S. (2003). Not as easy as it seems: Automating the construction of lexical chains using roget’s thesaurus. Em *Advances in Artificial Intelligence*, p. 544–549, Berlin, Heidelberg. Springer Berlin Heidelberg.
- [Kadhim, 2019] Kadhim, A. I. (2019). Survey on supervised machine learning techniques for automatic text classification. *Artificial Intelligence Review*, 52(1):273–292.
- [Ketchen e Shook, 1996] Ketchen, D. J. e Shook, C. L. (1996). The application of cluster analysis in strategic management research: an analysis and critique. *Strategic management journal*, 17(6):441–458.
- [Kwon et al., 2013] Kwon, S., Cha, M., Jung, K., Chen, W. e Wang, Y. (2013). Prominent features of rumor propagation in online social media. Em *2013 IEEE 13th International Conference on Data Mining*, p. 1103–1108. IEEE.
- [Lazer et al., 2018] Lazer, D. M., Baum, M. A., Benkler, Y., Berinsky, A. J., Greenhill, K. M., Menczer, F., Metzger, M. J., Nyhan, B., Pennycook, G., Rothschild, D. et al. (2018). The science of fake news. *Science*, 359(6380):1094–1096.
- [Li et al., 2015] Li, Y., Xu, L., Tian, F., Jiang, L., Zhong, X. e Chen, E. (2015). Word embedding revisited: A new representation learning and explicit matrix factorization perspective. Em *Twenty-Fourth International Joint Conference on Artificial Intelligence*.
- [Liu et al., 2010] Liu, G., Wang, Y. e Orgun, M. A. (2010). Quality of trust for social trust path selection in complex social networks. Em *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*, p. 1575–1576. International Foundation for Autonomous Agents and Multiagent Systems.

- [Manning et al., 2010] Manning, C., Raghavan, P. e Schütze, H. (2010). Introduction to information retrieval. *Natural Language Engineering*, 16(1):100–103.
- [Manning e Schutze, 1999] Manning, C. e Schutze, H. (1999). *Foundations of statistical natural language processing*. MIT press.
- [Manning et al., 2014] Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. e McClosky, D. (2014). The stanford corenlp natural language processing toolkit. Em *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, p. 55–60.
- [Mattos et al., 2019] Mattos, D. M. F., Velloso, P. B. e Duarte, O. C. M. B. (2019). An agile and effective network function virtualization infrastructure for the internet of things. *Journal of Internet Services and Applications*, 10(1):6.
- [Meel e Vishwakarma, 2020] Meel, P. e Vishwakarma, D. K. (2020). Fake news, rumor, information pollution in social media and web: A contemporary survey of state-of-the-arts, challenges and opportunities. *Expert Systems with Applications*, 153:112986.
- [Mikolov et al., 2013] Mikolov, T., Chen, K., Corrado, G. e Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- [Mitra e Gilbert, 2015] Mitra, T. e Gilbert, E. (2015). Credbank: A large-scale social media corpus with associated credibility annotations. Em *ICWSM*, p. 258–267.
- [Monteiro et al., 2018] Monteiro, R. A., Santos, R. L., Pardo, T. A., de Almeida, T. A., Ruiz, E. E. e Vale, O. A. (2018). Contributions to the study of fake news in portuguese: New corpus and automatic detection results. Em *International Conference on Computational Processing of the Portuguese Language*, p. 324–334. Springer.
- [Navigli, 2009] Navigli, R. (2009). Word sense disambiguation: A survey. *ACM computing surveys (CSUR)*, 41(2):1–69.
- [Oshikawa et al., 2018] Oshikawa, R., Qian, J. e Wang, W. Y. (2018). A survey on natural language processing for fake news detection. *arXiv preprint arXiv:1811.00770*.
- [Otter et al., 2020] Otter, D. W., Medina, J. R. e Kalita, J. K. (2020). A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*.
- [Papadimitriou et al., 2000] Papadimitriou, C. H., Raghavan, P., Tamaki, H. e Vempala, S. (2000). Latent semantic indexing: A probabilistic analysis. *Journal of Computer and System Sciences*, 61(2):217–235.
- [Pennebaker et al., 2001] Pennebaker, J. W., Francis, M. E. e Booth, R. J. (2001). Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates*, 71(2001):2001.

- [Pennycook et al., 2020] Pennycook, G., McPhetres, J., Zhang, Y., Lu, J. G. e Rand, D. G. (2020). Fighting covid-19 misinformation on social media: Experimental evidence for a scalable accuracy-nudge intervention. *Psychological science*, 31(7):770–780.
- [Perdisci et al., 2006] Perdisci, R., Gu, G. e Lee, W. (2006). Using an ensemble of one-class svm classifiers to harden payload-based anomaly detection systems. Em *Sixth International Conference on Data Mining (ICDM'06)*, p. 488–498.
- [Poddar et al., 2019] Poddar, K., Umadevi, K. et al. (2019). Comparison of various machine learning models for accurate detection of fake news. Em *2019 Innovations in Power and Advanced Computing Technologies (i-PACT)*, volume 1, p. 1–5. IEEE.
- [Potthast et al., 2017] Potthast, M., Kiesel, J., Reinartz, K., Bevendorff, J. e Stein, B. (2017). A stylometric inquiry into hyperpartisan and fake news. *arXiv preprint arXiv:1702.05638*.
- [Qian et al., 2018] Qian, F., Gong, C., Sharma, K. e Liu, Y. (2018). Neural user response generator: Fake news detection with collective user intelligence. Em *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, p. 3834–3840. International Joint Conferences on Artificial Intelligence Organization.
- [Rashkin et al., 2017] Rashkin, H., Choi, E., Jang, J. Y., Volkova, S. e Choi, Y. (2017). Truth of varying shades: Analyzing language in fake news and political fact-checking. Em *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, p. 2931–2937.
- [Robnik-Šikonja e Kononenko, 2003] Robnik-Šikonja, M. e Kononenko, I. (2003). Theoretical and Empirical Analysis of ReliefF and RReliefF. *Machine Learning*, 53(1/2):23–69.
- [Rousseeuw e Kaufman, 1990] Rousseeuw, P. J. e Kaufman, L. (1990). Finding groups in data. *Hoboken: Wiley Online Library*.
- [Rubin et al., 2016] Rubin, V., Conroy, N., Chen, Y. e Cornwell, S. (2016). Fake news or truth? using satirical cues to detect potentially misleading news. Em *Proceedings of the second workshop on computational approaches to deception detection*, p. 7–17.
- [Rubin, 2010] Rubin, V. L. (2010). On deception and deception detection: Content analysis of computer-mediated stated beliefs. Em *Proceedings of the 73rd ASIS&T Annual Meeting on Navigating Streams in an Information Ecosystem-Volume 47*, p. 32. American Society for Information Science.
- [Rubin, 2014] Rubin, V. L. (2014). Pragmatic and cultural considerations for deception detection in asian languages. Em *ACM Transactions on Asian Language Information Processing*.
- [Rubin et al., 2015a] Rubin, V. L., Chen, Y. e Conroy, N. J. (2015a). Deception detection for news: three types of fakes. Em *Proceedings of the 78th ASIS&T Annual Meeting*:

*Information Science with Impact: Research in and for the Community*, p. 83. American Society for Information Science.

- [Rubin et al., 2015b] Rubin, V. L., Conroy, N. J. e Chen, Y. (2015b). Towards news verification: Deception detection methods for news discourse. Em *Hawaii International Conference on System Sciences*.
- [Santia e Williams, 2018] Santia, G. C. e Williams, J. R. (2018). Buzzface: A news veracity dataset with facebook user commentary and egos. Em *Twelfth International AAAI Conference on Web and Social Media*, p. 531–540.
- [Schubert et al., 2017] Schubert, E., Sander, J., Ester, M., Kriegel, H. P. e Xu, X. (2017). Dbscan revisited, revisited: why and how you should (still) use dbscan. *ACM Transactions on Database Systems (TODS)*, 42(3):1–21.
- [Sharma et al., 2019] Sharma, K., Qian, F., Jiang, H., Ruchansky, N., Zhang, M. e Liu, Y. (2019). Combating fake news: A survey on identification and mitigation techniques. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(3):1–42.
- [Sharma e Sharma, 2019] Sharma, S. e Sharma, D. K. (2019). Fake news detection: A long way to go. Em *2019 4th International Conference on Information Systems and Computer Networks (ISCON)*, p. 816–821. IEEE.
- [Shu et al., 2019] Shu, K., Mahudeswaran, D. e Liu, H. (2019). Fakenewstracker: a tool for fake news collection, detection, and visualization. *Computational and Mathematical Organization Theory*, 25(1):60–71.
- [Shu et al., 2020] Shu, K., Mahudeswaran, D., Wang, S., Lee, D. e Liu, H. (2020). Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media. *Big Data*, 8(3):171–188.
- [Shu et al., 2017] Shu, K., Sliva, A., Wang, S., Tang, J. e Liu, H. (2017). Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 19(1):22–36.
- [Socher et al., 2013] Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y. e Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. Em *Proceedings of the 2013 conference on empirical methods in natural language processing*, p. 1631–1642.
- [Thorne et al., 2018] Thorne, J., Vlachos, A., Christodoulopoulos, C. e Mittal, A. (2018). FEVER: a large-scale dataset for fact extraction and VERification. Em *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, p. 809–819. Association for Computational Linguistics.
- [Van Bavel et al., 2020] Van Bavel, J. J., Baicker, K., Boggio, P. S., Capraro, V., Cichocka, A., Cikara, M., Crockett, M. J., Crum, A. J., Douglas, K. M., Druckman, J. N. et al. (2020). Using social and behavioural science to support covid-19 pandemic response. *Nature Human Behaviour*, p. 1–12.

- [Verikas et al., 2011] Verikas, A., Gelzinis, A. e Bacauskiene, M. (2011). Mining data with random forests: A survey and results of new tests. *Pattern recognition*, 44(2):330–349.
- [Vosoughi et al., 2018] Vosoughi, S., Roy, D. e Aral, S. (2018). The spread of true and false news online. *Science*, 359(6380):1146–1151.
- [Wang, 2017] Wang, W. Y. (2017). “Liar, liar pants on fire”: A new benchmark dataset for fake news detection. Em *Annual Meeting of the Association for Computational Linguistics - ACL 2017*.
- [Xu e Wunsch, 2005] Xu, R. e Wunsch, D. (2005). Survey of clustering algorithms. *IEEE Transactions on neural networks*, 16(3):645–678.
- [Zhai et al., 2014] Zhai, Y., Ong, Y.-S. e Tsang, I. W. (2014). The emerging “big dimensionality”. *IEEE Computational Intelligence Magazine*, 9(3):14–26.
- [Zhou et al., 2004] Zhou, L., Burgoon, J. K., Nunamaker, J. F. e Twitchell, D. (2004). Automating linguistics-based cues for detecting deception in text-based asynchronous computer-mediated communications. *Group decision and negotiation*, 13(1):81–106.
- [Zhou et al., 2015] Zhou, X., Cao, J., Jin, Z., Xie, F., Su, Y., Chu, D., Cao, X. e Zhang, J. (2015). Real-time news certification system on sina weibo. Em *Proceedings of the 24th International Conference on World Wide Web*, p. 983–988.
- [Zhou e Zafarani, 2018] Zhou, X. e Zafarani, R. (2018). Fake news: A survey of research, detection methods, and opportunities. *arXiv preprint arXiv:1812.00315*.
- [Zhou et al., 2019] Zhou, Z., Guan, H., Bhat, M. M. e Hsu, J. (2019). Fake news detection via nlp is vulnerable to adversarial attacks. *arXiv preprint arXiv:1901.09657*.
- [Zubiaga et al., 2016] Zubiaga, A., Liakata, M., Procter, R., Wong Sak Hoi, G. e Tolmie, P. (2016). Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PloS one*, 11(3):e0150989.