UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Atílio Gabriello

Uso das Redes Neurais Convolucionais na Identificação de Fake News

Uberlândia, Brasil

2024

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Atílio Gabriello

Uso das Redes Neurais Convolucionais na Identificação de Fake News

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, como parte dos requisitos exigidos para a obtenção título de Bacharel em Ciência da Computação.

Orientador: Fernanda Maria da Cunha Santos

Universidade Federal de Uberlândia – UFU

Faculdade de Computação

Bacharelado em Ciência da Computação

Uberlândia, Brasil

2024

Atílio Gabriello

Uso das Redes Neurais Convolucionais na Identificação de Fake News

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, como parte dos requisitos exigidos para a obtenção título de Bacharel em Ciência da Computação.

Trabalho aprovado. Uberlândia, Brasil, 18 de abril de 2024:

Fernanda Maria da Cunha Santos
Orientador

Ivan da Silva Sendin

Marcelo Zanchetta do Nascimento

Uberlândia, Brasil
2024

Agradecimentos

Agradeço primeiramente a minha família, ao meu pai que infelizmente não conseguirá ver esse resultado mas que me acompanha todos os dias em meu pensamento, sempre me fazendo lembrar da pessoa incrível e amada que ele era e sempre me mostrando como posso ser uma pessoa melhor. A minha irmã Júlia que é um exemplo de coragem e determinação sempre me mostrando sua força. A minha mãe Adriana que é a pessoa que mais me apoiou em todos os meus sonhos e objetivos, por mais loucos que eles parecessem e que é uma inspiração para mim e tantos outros por conta de sua resiliência.

Agradeço a UFU e todos os professores que me ensinaram muito e participaram da minha formação, um agradecimento especial para a minha orientadora Fernanda que me acompanhou durante este trabalho, me ajudando muito com sua experiência, paciência e constante suporte durante todo o desenvolvimento.

Deixo aqui também um agradecimento especial para meus grandes amigos Alessandra, Junin, Gabriela e Renato, todos foram essenciais para mim e minha família durante os dias mais difíceis oferecendo um incrível suporte e que hoje fazem parte da minha vida e fico muito feliz por tê-los por perto.

Por fim agradeço a Deus que sempre me colocou no caminho correto mesmo quando eu me afastava.

Resumo

Uma *fake news* é um tipo de notícia que é caracterizada por espalhar uma informação incorreta ou fictícia e que, frequentemente, passa despercebida, sendo responsável por gerar a desinformação em diversas áreas, desde assuntos correlatos à saúde até a política. A detecção dessas notícias é importante para garantir que elas não sejam tratadas como verdade e não tragam prejuízos e males à sociedade. Sabendo disso, objetiva-se neste trabalho a criação de um modelo computacional, constituído por redes neurais convolucionais, para realizar a classificação de notícias digitais e, consequentemente, identificar as *fake news*, tornando uma ferramenta para auxiliar o combate das mesmas. Para realização desta tarefa, duas bases de dados com notícias verdadeiras e falsas foram selecionadas e preparadas por meio de técnicas de processamento de linguagem natural, para testar a aplicabilidade do modelo proposto na classificação de *fake news*. As redes neurais convolucionais conseguiram apresentar resultados satisfatórios com acurácia acima dos 89% para ambas as bases de dados utilizadas, bem como as limitações ao realizar a tarefa de classificação textual.

Palavras-chave: Processamento de Linguagem Natural, Redes Neurais Convolucionais, Fake News.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

AM	Aprendizado de Máquina
CNN	Rede Neural Convolucional
IA	Inteligência Artificial
LSTM	*Long Short Term Memory*
PLN	Processamento de Linguagem Natural
RNN	Rede Neural Recorrente

Sumário

1 Introdução

Ao analisar o tipo de notícias divulgadas na década atual, notoriamente destacam-se as *fake news*, devido a rápida disseminação e maior acessibilidade de informações em tempo real proporcionados pelos avanços das tecnologias.

As *fake news* já foram vilãs em vários episódios: na área da saúde, onde vacinas foram divulgadas como um malefício (G1, 2023b), até na área da politica, onde foi usada como ferramenta de campanha eleitoral (G1, 2023a). Assim, o tema torna-se de extrema valia já que *fake news* é algo que atinge de 4 a cada 10 brasileiros diariamente (BRASIL, 2022), sendo responsável por causar diversos malefícios. Uma ferramenta computacional que seja capaz de identificar as *fake news*, poderia ajudar os usuários da Internet, reduzindo os danos que possam causar e até auxiliar na redução da disseminação das mesmas.

Por ser um tema relevante é possível encontrar diversos artigos que englobam o tema e que utilizam diferentes técnicas para lidar com a identificação das *fake news*. Umas que merecem destaques são as técnicas de Processamento de Linguagem Natural (PLN), que realizam o pré-processamento textual, e os algoritmos da Inteligência Artificial (IA), como as redes neurais, e de Aprendizado de Máquina (AM) para identificar e classificar *Fake News*.

A grande maioria dos trabalhos correlatos foram aplicados à textos da língua inglesa. Kaliyar et al. (2020) propõem um modelo único para realizar a classificação de notícias e o modelo obteve uma acurácia de 99.41%. Esse modelo implementa uma Rede Neural Convolucional (CNN) com modificações para que a mesma seja mais profunda do que o usual, por isso é chamada de *Deeper CNN*.

Em relação à base de dados da língua portuguesa, Silva et al. (2020) se propõe a trazer o estudo de detecção de *fake news* para o português e utiliza a base de dados Monteiro et al. (2018) e algoritmos de AM para classificação, como por exemplo *random forest*. Já, o trabalho de Guarise e Rezende (2019) mostra a eficiência da rede neural Long Short Term Memory (LSTM) para classificar as *fake news* na língua portuguesa. As redes neurais LSTM são uma arquitetura de rede neural recorrente (RNN) muito utilizada em trabalhos relacionados à PLN para realizar a classificação de textos.

Uma outra arquitetura de rede neural que se destaca é das Redes Neurais Convolucionais (CNN), a qual atua como método supervisionado. No trabalho desenvolvido pelos autores Yu et al. (2017), implementaram um modelo baseado em CNN que extrai características essenciais do texto para que a camada totalmente conectada faça a classificação. Essa arquitetura de rede já foi empregada em trabalhos vinculados à PLN como tradução, análise de sentimentos e outros.

1.1 Objetivo Geral

O objetivo desse estudo é definir um modelo computacional para identificar *fake news* em textos digitais da língua portuguesa. Esse modelo será constituído por técnicas de processamento de linguagem natural, na etapa de pré-processamento textual, e por redes neurais convolucionais para a etapa de classificação. O modelo computacional será atestado em duas bases de dados: a Fake.br (MONTEIRO et al., 2018) e uma segunda base criada com a união da Fake.br com a base de dados FakeRecogna.

1.1.1 Objetivos Específicos

Os objetivos específico podem ser listados em:

- Definição das Bases de Dados: também conhecido por corpus. Nesta etapa, será descrito as características das duas bases de dados da língua Portuguesa;
- Pré-processamento das Bases de Dados: o conjunto de dados precisa ser pré-processado para que possa ser usado em um modelo de redes neurais. Isso inclui a limpeza dos dados, como remoção de pontuações e caracteres especiais, e a transformação dos dados em um formato apropriado para o modelo, como vetores de palavras.
- Definição do modelo computacional: escolha da melhor topologia da rede neural CNN;
- Análise dos dados: comparação dos resultados obtidos, com aqueles que já foram descritos em outros trabalhos com a mesma base de dados Fake.br;

As ferramentas computacional que serão utilizadas para o desenvolvimento do trabalho são:

- linguagem de programação Python;
- Bibliotecas Python com enfase em aprendizado de máquina;
- Bibliotecas Python com enfase em processamento de linguagem natural;
- Bibliotecas Python que permitem criação de gráficos para análise dos resultados;
- Base de dados Fake.br (MONTEIRO et al., 2018);
- Base de dados FakeRecogna (GARCIA; AFONSO; PAPA, 2022);

1.2 Organização do Texto

Para melhor descrever o projeto o texto apresenta a organização em capítulos que englobam todas as informações que envolvem o mesmo. O primeiro capítulo apresentou o motivo e a importância deste projeto e também explicita quais são os objetivos dele.

O capitulo 2 irá explicar a fundamentação teórica dos principais conceitos envolvidos neste trabalho. Além disso, este capítulo descreverá outros trabalhos que também tratam de conceitos correlacionados ao tema proposto e que contribuíram para auxiliar no engajamento das ideias.

O capitulo 3 é onde todas as etapas de desenvolvimento do projeto são detalhadas e também onde estão presentes explicações de decisões de todos os aspectos essenciais para a criação de um sistema computacional para a criação de *fake news*. Além disso é no capítulo 3 onde serão apresentados as métricas e testes realizados para validar o modelo criado.

Por fim o capitulo 4 contém a conclusão, onde estão as considerações finais do projeto bem como as sugestões para trabalhos futuros que podem ser realizados para aprofundar o estudo de classificação de *fake news*.

2 Fundamentação Teórica

2.1 Redes Neurais Convolucionais

Redes neurais convolucionais (CNNs) são um tipo de arquitetura de rede neural artificial, principalmente, utilizada na classificação de padrões que podem ser vistos em duas dimensões, como por exemplo imagens. Geralmente as CNNs apresentam uma estrutura que é dividia em três camadas principais constituídas por diferentes funções matemáticas para garantir que a classificação final seja a mais precisa possível. As três camadas são: camada convolucional, camada de agrupamento e camada inteiramente conectada. A primeira utiliza filtros que recebem os dados de entrada e esses filtros são capazes de aprender características sobre os dados. A camada de agrupamento busca reduzir o tamanho dos dados resultantes da camada de convolução, focando em suas características e tratando cada uma independentemente. A última camada apresenta os pesos atribuídos a cada característica do dado de entrada, além de que essa camada se conecta com o restante da rede para que esses pesos possam ser atualizados conforme a rede é treinada (KIM, 2014). A figura 1 é um exemplo simples de como as camadas se relacionam em uma tarefa de classificação de texto.

O conceito de CNNs surgiu em 1980 com Fukushima (2007) que apresentou um modelo precursor das CNNs que já apresentava o conceito de reconhecimento de padrões através do aprendizado, entre outros. No entanto, foi em 1998 que Lecun et al. (1998) apresentou o primeiro modelo bem sucedido de uma CNN, os autores mostram com detalhes como uma arquitetura neural do tipo CNN pode ser usada para o reconhecimento

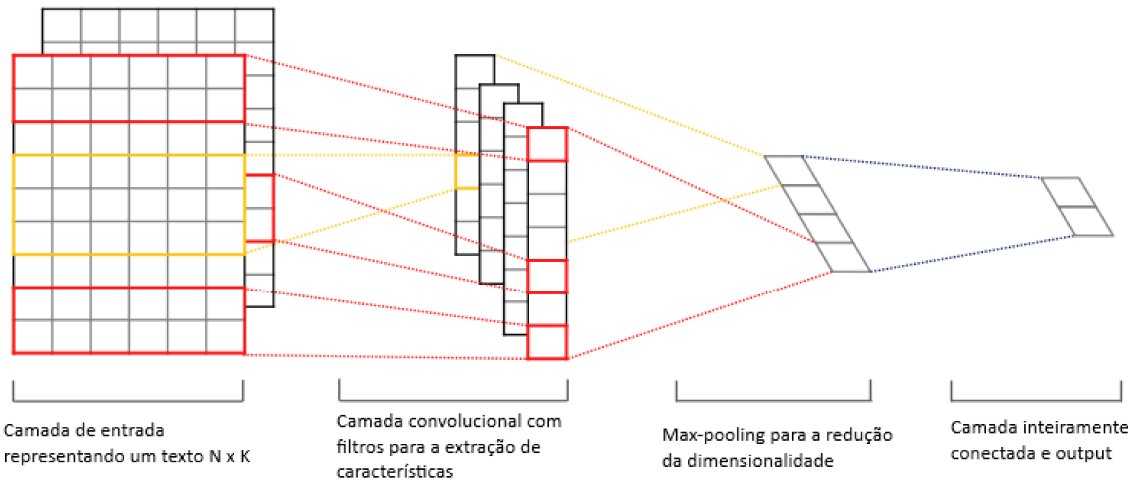

Figura 1 – Exemplo de arquitetura de redes neurais convolucionais (CNN) para classificação de texto. Adaptado de Kim (2014)

de escrita feita à mão de dígitos em documentos. Apenas por volta de 2010, com o artigo Russakovsky et al. (2015), os autores mostraram como a CNN era capaz de dominar a área de computação visual contribuindo para a notoriedade que se tem nos dias atuais.

O uso de CNNs para o reconhecimento de imagens e identificação de objetos foi amplamente empregado, como mostram os artigos já citados. Porém, avanços recentes permitiram que outros tipos de dados fossem classificados. Kim (2014) trouxe o uso do modelo de rede neural para a classificação de textos, mostrando a possibilidade de utilizar a alta escalabilidade das redes CNN aplicadas à grandes textos.

Alguns exemplos relevantes nas CNNs são:

- **GoogLeNet (Inception)**: desenvolvidas por pesquisadores do *Google* Szegedy et al. (2014), esse modelo introduz um bloco que é chamado *Inception* que propõe apresentar múltiplos filtros de tamanhos diferentes dentro da mesma camada convolucional, isso possibilita a extração de diferentes características de uma imagem para apresentar maiores precisões na tarefa de classificação.
- **FNDNet**: arquitetura proposta por Kaliyar et al. (2020) para a detecção de *fake news*. Esta arquitetura se diferencia ao apresentar camadas de convolução paralelas que geram resultados diferentes sobre a mesma entrada, esses resultados depois são concatenados e juntos passam pelas próximas camadas da rede, essa abordagem tem o intuito de extrair diferentes características sobre o mesmo texto para produzir melhores resultados.

2.2 Processamento de Linguagem Natural

Processamento de linguagem natural (PLN) é uma subárea da inteligência artificial que tem como objetivo capacitar as máquinas para que elas possam lidar com a linguagem humana, isto é, capacitar a máquina para poder compreender textos ou falas de forma semelhante a que um ser humano usaria numa comunicação. Para isso são necessários algoritmos para transformar a linguagem humana para uma forma que a máquina possa entender e que também possa responder.

O primeiro relato do uso de processamento de linguagem natural aconteceu em 1954, o artigo de Hutchins (2004) detalha como a *IBM* utilizou um computador para fazer a tradução de frases em russo para o inglês apresentando a capacidade computacional para ler e compreender textos. Depois disso diferentes abordagens foram usadas nas décadas seguintes para avançar a capacidade computacional para lidar com esse tipo de problema.

Mais recentemente pode-se observar avanços, pesquisadores do *Google* apresentaram o **BERT** no trabalho de Devlin et al. (2018), o modelo é capaz de realizar diferentes tarefas como classificação e responder perguntas, por exemplo.

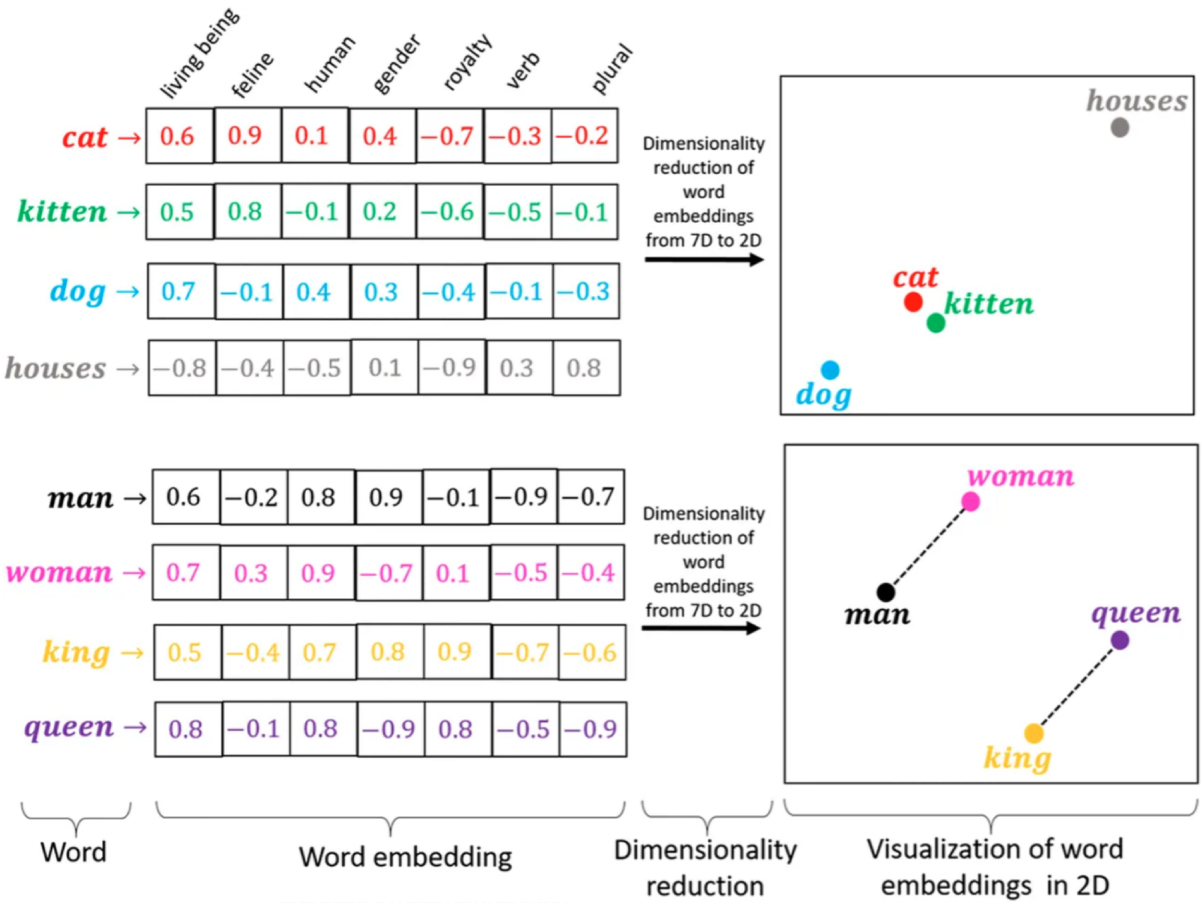

Figura 2 – Exemplo de vetores gerados pelo *word embedding*. Retirado de (GAUTAM, 2020).

Um exemplo muito relevante e atual em relação à PLN é o **GPT-3**, conhecido popularmente por *Chat GPT*. Ele utiliza de diversas técnicas de processamento de linguagem natural para poder receber, processar e responder as perguntas direcionadas ao (ChatGPT, 2022). O site ficou famoso por permitir que qualquer pessoa possa fazer perguntas sobre diversos temas e obter respostas de forma rápida e na maioria das vezes de forma assertiva.

2.2.1 *Word Embedding*

Uma das técnicas apresentadas para realizar o processamento de linguagem natural é o *word embedding*, esta técnica busca representar palavras como vetores de tamanho fixo onde em cada posição do vetor apresenta o quanto aquela palavra se relaciona com outra. A figura 2 demonstra como é o resultado do *word embedding*.

Para aproximar palavras semelhantes e criar o vetor de cada uma delas é necessário utilizar algoritmos que usam um grande vocabulário, a partir daí utiliza textos durante o treinamento e ao final gera associações entre as palavras com um grau de relação. Os principais algoritmos utilizados são o *Word2Vec* e o *Glove* embora existam outros.

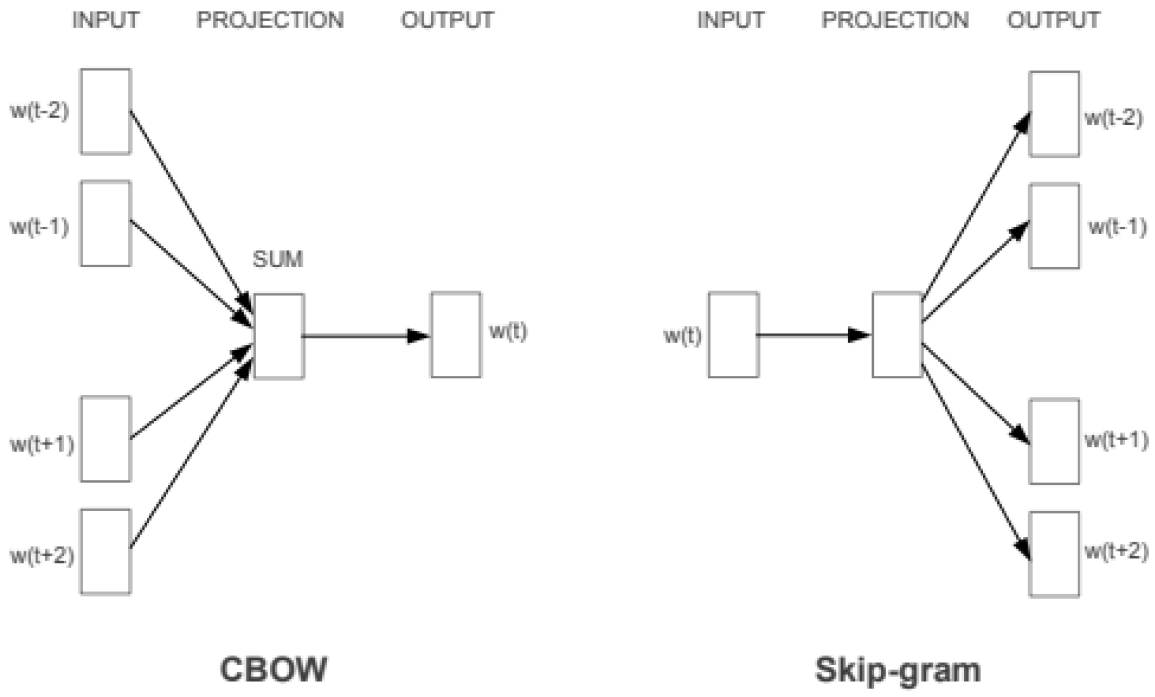

Figura 3 – Diferença entre **CBOW** e o **SKIP-GRAM** no *Word2Vec*. Retirado de (MIKOLOV et al., 2013).

2.2.2 *Word2Vec*

Como dito anteriormente o *Word2Vec* é um dos algoritmos utilizados para realizar a tarefa de *word embedding*, ele foi apresentado por Mikolov et al. (2013) e apresenta duas formas de implementação **CBOW** e **SKIP-GRAM**. Ambas as implementações utilizam um vetor, chamado de 'janela', de tamanho fixo que percorre o texto movimentando as palavras, ou seja, cada elemento desse vetor é uma palavra do texto. A diferença entre as duas implementações esta relacionada com o resultado da janela, o **CBOW** recebe um contexto (uma série de palavras) e como resultado gera uma palavra que melhor apresente esse contexto. Já o **SKIP-GRAM** faz o contrário, recebe uma palavra e analisa o contexto em torno da mesma dentro de uma determinada distância, com isso resulta em uma série de palavras que melhor representam a palavra analisada. A figura 3 ilustra a diferença entre o **CBOW** e o **SKIP-GRAM**.

2.3 *Fake News*

Fake news como o nome diz trata-se de notícias que tem como características apresentar uma falsidade relacionada a ela, isso pode ser visto desde coisas pequenas como um dado falso ou não exato, até informações maiores, como por exemplo quando a notícia inteira apresenta relatos falsos. Independente de como seja, as *fake news* tem sido um

problema sério em nossos meios de comunicação, pois passam a propagar a desinformação de uma forma rápida e diária como mostra em Brasil (2022).

O termo *fake news* ganhou notoriedade principalmente em 2016 com as eleições estadunidenses (PAíS, 2018). Após isso, as *fake news* nunca saíram da vida das pessoas e até hoje são usadas como ferramenta de campanha eleitoral por candidatos como já citado em (G1, 2023a).

Embora as *fake news* sejam difíceis de combater, existem páginas dedicadas a expor notícias falsas como é o caso do (GLOBO, 2018). Além de iniciativas como a anterior, recentes avanços levaram ao desenvolvimento de ferramentas capazes de facilitar na identificação desse tipo de notícia para que ela se propague menos. O avanço do uso da inteligencia artificial para detecção dessas notícias crescem e já apresentam resultados positivos como mostra o artigo Kaliyar et al. (2020). Neste estudo, obteve acurácia de 98.36%, contudo, ainda é preciso avanços para garantir que todas as notícias possam ser reconhecidas e com uma maior velocidade para evitar que as mesmas prejudiquem muitas pessoas.

Um exemplo muito relevante são modelos de checagem de fatos baseados em GPT como o apresentado por Choi e Ferrara (2024), esses são modelos generativos que visam gerar respostas que já tiveram seus fatos checados para garantir que não apresente informações enganosas. Abordagens como esta reduzem a propagação de *fake news* já que os textos gerados tem garantia de veracidade permitindo que as pessoas tenham uma fonte confiável para recorrer, porém esses modelos dependem diretamente dos avanços citados no paragrafo anterior já que esses modelos baseados em GPT necessitam de fontes préviamente validadas para serem treinados.

2.4 Revisão Bibliográfica

Esta seção contém a descrição de trabalhos que foram relevantes para a edificação deste TCC e que fizeram a associação de técnicas de PLNs e algoritmos de Aprendizado de Máquina (AM) para a identificação e classificação de *Fake news*.

O *survey* Hu et al. (2022) descreve um compilado de diferentes métodos de detecção de *fake news* baseados em Aprendizado Profundo segundo os algoritmos de aprendizado supervisionado, não-supervisionado e semi-supervisionado. Para cada uma destas, o artigo descreve quantitativamente os tipos de dados usados pelas pesquisas citadas assim como as técnicas de AM aplicados. Ademais, apresenta uma descrição das bases de dados relevantes, destacando as características de cada uma, como a quantidade de amostras, o período de extração das informações, a origem dos dados e o tema das notícias. Na sequência, descreve os inúmeros algoritmos e métodos de AM empregados nos estudos analisados. Os resultados mostram que embora já existam grandes avanços no campo de

detecção de *fake news*, ainda existe um grande caminho a ser traçado para que esses sistemas de detecção apresentem melhores resultados. Ou seja, que os sistemas de detecção de *fake news* sejam capazes de identificar o surgimento sutil de notícias falsas em contextos maiores, ou quando uma *fake news* é cópia de outras fontes ou se é a própria fonte de disseminação, e, principalmente, que sejam capazes de reconhecer idiomas diferentes da língua inglesa.

Outro artigo de relevância é o Kaliyar et al. (2020) que propõe um modelo baseado em redes neurais convolucionais profundas (*deep convolutional neural network*) que foi nomeado **FNDNet**. Neste artigo, o modelo utiliza uma abordagem onde as características dos textos serão extraídas por diferentes camadas convolucionais que apresentam filtros de tamanho diferentes, isso garante que a rede possua diferentes características para os mesmos textos e consequentemente tenha mais informações para avaliar os mesmos. Inicialmente, foi utilizado uma base de dados pré-processada pelo *Word embedding* utilizando o algoritmo **GLOVE**, isto é, transformou os textos da base de dados do site Kaggle, em estruturas de dados adequadas para serem direcionadas à rede neural. O modelo proposto foi submetida a comparações com resultados obtidos de outros métodos de classificação para a mesma base de dados, a fim de avaliar se os resultados alcançados foram significativos. Dentre os resultados obtidos pelos classificadores, os principais foram: K — Nearest Neighbors (KNN), rede neural convolucional (CNN), *Long Short Term Memory* (LSTM) e a rede FNDNet. A tabela 1 mostra as acurácia dos modelos citados e a partir dela é possível observar como o modelo criado se destacou entre os outros.

Modelo	**Acurácia** %
FDNet	98.36
LSTM	97.25
CNN	91.50
KNN	53.75

Tabela 1 – Resultados dos testes de acurácia realizados em comparação com a **FNDNet**. Dados retirados de (KALIYAR et al., 2020).

Além dos artigos já citados, a monografia Guarise e Rezende (2019) apresenta um modelo LSTM para detecção de *fake news* numa base de dados da língua portuguesa (MONTEIRO et al., 2018). Essa base de dados reúne 7200 notícias em português sendo elas falsas ou verdadeiras. O modelo de rede neural implementado foi o HAN, que é baseado em LSTM, e dividiu a base de dados em 80% para treinamento e o restante para avaliar o modelo. Os resultados obtidos do modelo foram satisfatórios, com o valor da acurácia de 95.35%, o que é um ótimo resultado se comparando com o mesmo modelo utilizado para uma base de dados da língua inglesa que apresentou 95.4%. Esses valores mostram que modelos utilizados para outras línguas podem ser aplicados para a língua portuguesa e ainda assim obter resultados significativos. Além da acurácia outros testes

foram realizados outros cálculos para validar os resultados da rede, as precisões de verdadeiro e de falso foram calculadas gerando 93.89% e 96.80% respectivamente, isso indica que quando a rede classificou um texto como falso apenas 3.20% das vezes ela estava errada. Além dos testes realizados utilizando dados extraídos da base utilizada, a rede foi testada com entradas que foram retiradas de outra fonte, com essas notícias a base não apresentou tanto sucesso mostrando uma acurácia de apenas 73.33%, isso demonstra um problema nas redes indicando a falha em generalizar para notícias de outra origem que não a mesma da qual foi treinada.

3 Sistema Computacional para Detecção de *Fake News*

Neste capítulo serão explicadas as etapas do sistema computacional proposto contendo a implementação da CNN capaz de fazer a detecção de *fake news*. Pode-se enumerar as etapas do sistema por:

1. Preparação das bases de dados.
2. Pré-processamento dos dados.
3. *Word embedding*.
4. Construção da CNN e treinamento.
5. Avaliação dos resultados.

3.1 Preparação das bases de dados

Para melhor avaliar o modelo proposto, foram preparados e selecionados duas diferentes bases de dados para que o modelo fosse treinado:

1. A primeira base é a Fake.br, criado por (MONTEIRO et al., 2018).
2. A segunda base foi gerada através da junção da base de dados anteriormente citada e a FakeRecogna de (GARCIA; AFONSO; PAPA, 2022).

A primeira base de dados foi criada em 2018 e é composta por 7200 notícias rotuladas, sendo 50% fatos verdadeiros e 50% fatos falsos, essas notícias abordam diferentes assuntos desde política até televisão e celebridades. A base de dados já apresenta os textos com pré-processamento, ou seja, palavras vazias foram removidas (palavras sem significado) assim como palavras acentuadas. Ademais, os textos são constituídos da manchete e do corpo da notícia.

A base FakeRecogna é mais recente, foi criada em 2022 e possui 11902 notícias, com 50% verdadeiras e a outra metade falsas, o que totaliza a segunda base de dados experimental com 19102 notícias. A base de dados FakeRecogna aborda diferentes temas, incluindo política, celebridades e ciência. Diferente da primeira base de dados, esta não apresenta pré-processamento e os textos são formados pela manchete, subtitulo e pelo corpo da notícia, além dos metadados como a data quando publicada e o autor que escreveu, por exemplo.

Como dito anteriormente, a segunda base de dados foi gerada pela junção da primeira com a FakeRecogna. Para isso foi necessário alguns ajustes para garantir que ela esteja nos mesmos moldes da primeira, para isso o titulo, subtitulo e o corpo da notícia foram concatenados para que se tornasse um texto único como na base número 1. Ao final a base 2 apresentou 19102 notícias com metade verdadeiras e o restante falsas.

3.2 Pré-processamento dos dados

Para garantir bons resultados nas etapas subsequentes do modelo computacional, foi necessário fazer modificações nos textos das notícias para remover palavras e alguns caracteres que não agregam valor significativo para a etapa de classificação.

Como as bases de dados já haviam sido previamente preparadas, as ações na etapa de pré-processamento foram:

- Remoção de colunas da base de dados que não apresentam utilidades para a classificação, como por exemplo a data da notícia.
- Remoção de pontuações.
- Remoção de números.
- Remoção de acentos e símbolos diversos.
- Conversão de todas as palavras para minusculas.
- Lematização, para garantir que palavras com mesmo radical sejam tratadas com o mesmo valor.

Ao final da etapa acima as notícias passaram a ser um vetor de *tokens* onde cada um deles é uma palavra e cada notícia apresentava um identificador 0 ou 1 para indicar se a mesma é falsa ou verdadeira, respectivamente.

3.3 *Word embedding*

Como detalhado na Seção 2.2, o *Word Embedding* é uma etapa de pré-processamento, responsável pela preparação e transformação dos dados textuais em números para que eles sejam redirecionados à entrada da rede neural. Em outras palavras, a base de dados é um conjunto de palavras que para a rede neural não representa significado algum, sendo o *Word Embedding* o tradutor das palavras presentes nas notícias em números compreensível às redes neurais.

O algoritmo que foi utilizado foi o *Word2Vec* com a implementação do **SKIP-GRAM**, ou seja, dado uma palavra que se encontra no meio da janela as palavras ao lado são utilizadas para fazer associações, isso permite que pesos fossem gerados para cada palavra que está dentro da distância da central, e no fim um vetor de quanto relacionadas cada palavra está. A janela é movimentada por todo o texto até que todas as palavras tenham sido percorridas.

O primeiro passo foi utilizar a própria base de para gerar o vocabulário que foi utilizado para o treinamento e após adicionar o vocabulário o modelo *Word2Vec* foi treinado por 10 gerações utilizando uma janela de tamanho 5. Ao final desse processo foi gerado um modelo de *Word2Vec* que contem as associações feitas para as palavras do vocabulário. Com isso dois dicionários são criados para mapear as palavras aos seus respectivos índices e vice-versa, isso possibilita que cada um dos textos seja convertido para uma versão numérica dele mesmo, essa sequência numérica é usada como entrada para que a CNN possa ser treinada.

A última alteração nos textos é adicionar *padding* para que os textos possuam o mesmo tamanho. O tamanho é calculado analisando todos os textos e de forma que 95% deles tenham seu tamanho preservado, os outros textos são ajustados para garantir que a rede possa recebe-los como entrada.

Ao final dessa etapa, os dados são divididos em três partes, sendo elas, 70% para treinamento, 15% para teste e os 15% restantes foram utilizados para a validação durante o treinamento.

3.4 Construção da CNN

A Figura 4 ilustra as camadas presentes na CNN utilizada para a construção do modelo proposto. O conjunto de dados de treinamento serão usados como dados de entrada para a rede. A primeira camada, denominada camada *embedding*, é utilizada para transformar os dados de entrada num vetor de tamanho fixo para as camadas de convolução.

A arquitetura do modelo da rede CNN foi inspirada naquela mostrada por Kaliyar et al. (2020) que utiliza a técnica de usar várias camadas convolucionais com diferentes *kernels*, também chamado de filtros, cuja abordagem foi utilizada para que a rede pudesse capturar diferentes caraterísticas para os mesmos dados de entrada.

A camada *flatten* faz a conversão da entrada de múltiplas dimensões, originado da camada anterior, para um vetor de dimensão única que será a entrada das camadas *dense* que apresentam as funções de ativação *relu* e *sigmoid* respectivamente. Entre essas duas camadas foi necessário a utilização da camada de *Batch normalization* para normalizar

a ativação da camada anterior, o que ajuda para evitar o *overfitting* da rede e também fazer com que as convergências sejam mais estáveis e mais rápidas.

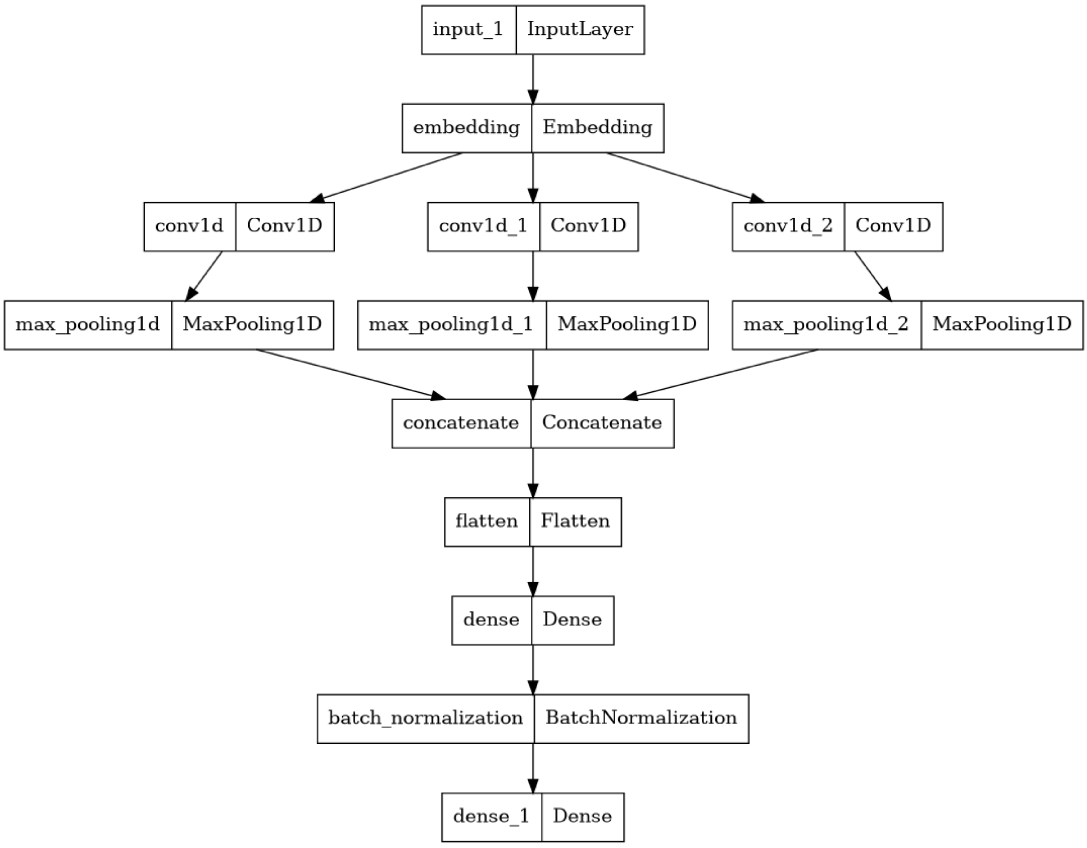

Figura 4 – Arquitetura da CNN proposta pelo modelo do sistema de classificação de fake news.

3.5 Avaliação dos Resultados

A primeira etapa de avaliação do sistema computacional proposto refere-se à fase de aprendizado da rede neural, isto é, as redes neurais foram submetidas ao treinamento com as duas bases de dados por apenas 5 épocas. A Figura 5 mostra como as métricas de acurácia, perda do treinamento e validação se comportaram ao longo do treinamento. As acurácias convergem rapidamente para os 100% enquanto as perdas apresentam decrescimento. É válido ressaltar a importância de ambas as perdas decrescerem, já que o crescimento constante da perda de validação enquanto a perda de treinamento diminui é um indicativo de *overfitting* (LUO et al., 2018). Ou seja, com os gráficos é possível averiguar que o comportamento das métricas sugerem que o treinamento evitou o *overfitting* da rede.

A primeira base de dados da rede apresentou, para o conjunto de teste, acurácia de 89.07% e perda de 39.80%. Já o conjunto de teste da segunda base de dados, a rede apresentou acurácia de 94.00% e perda de 17.45%.

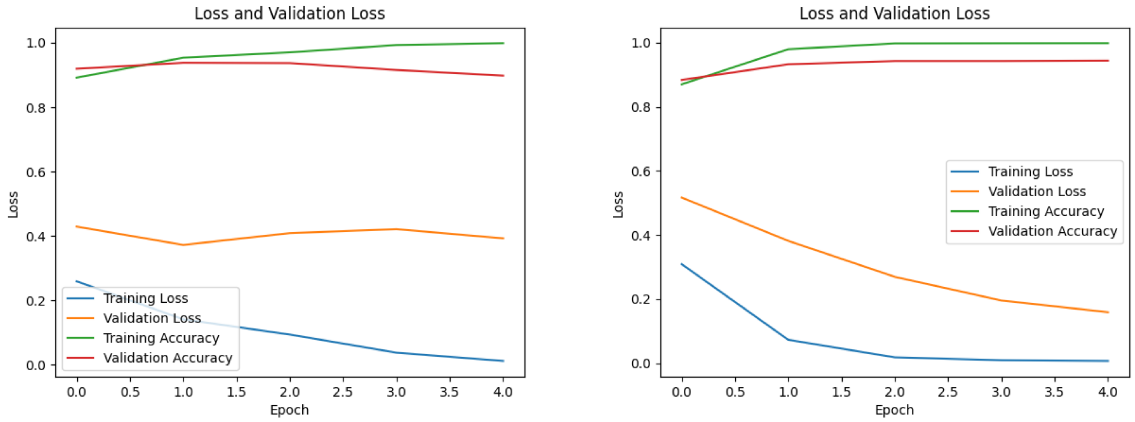

Figura 5 – Gráficos de treinamento com as bases 1 e 2, respectivamente.

Para o conjunto de dados de teste, foram utilizadas três métricas além da acurácia: *F1-score*, *Recall* e *Precisão*. As Tabelas 2 e 3 mostram com detalhes os resultados destas métricas para as bases dados 1 e 2, respectivamente.

	Precisão %	**Recall** %	**F1-Score** %
Classe 0 (Fake news)	97.26	80.08	87.84
Classe 1 (Notícias verdadeiras)	83.49	97.81	90.08

Tabela 2 – Resultados do conjunto de testes da base de dados 1.

	Precisão %	**Recall** %	**F1-Score** %
Classe 0 (Fake news)	94.76	93.31	94.02
Classe 1 (Notícias verdadeiras)	93.25	94.70	93.97

Tabela 3 – Resultados do conjunto de testes da base de dados 2.

Analisando as tabelas é possível observar na base de dados 1, os valores próximos de 80% do *recall* para notícias falsas e da precisão das notícias verdadeiras, indicam que algumas notícias foram erroneamente classificadas como verdadeiras, resultado esse que não aparece na acurácia das notícias verdadeiras. Essa disparidade foi resolvida utilizando a segunda base de dados que com um número significativo de notícias adicionadas conseguiu melhorar não só a acurácia da rede mas também a capacidade de identificar e acertar ambos os tipos de notícias.

A Figura 6 exibe as matrizes de confusão para os conjuntos de teste das duas bases de dados complementando os dados das tabelas 2 e 3. A matriz de confusão mostra qual classe foi classificada pelo modelo destacando que a classe 0 são consideradas as *fake news* e a classe 1 as notícias verdadeiras.

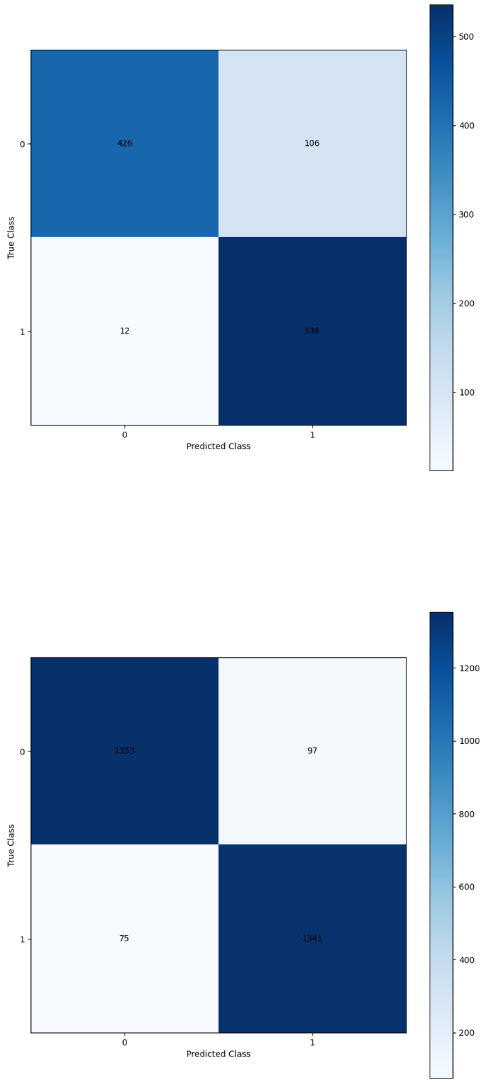

Figura 6 – Matriz de confusão do conjunto de dados para teste das base de dados 1 e 2 respectivamente.

3.5.1 Análise comparativa com outro modelo

Nesta seção, será feita a análise comparativa de dois modelos de sistema de classificação de *fake news* com a base de dados Fake.br (MONTEIRO et al., 2018), a qual é constituída de 7200 notícias rotuladas, sendo 50% para fatos verdadeiros e 50% para fatos falsos. Os modelos que serão comparados são de duas arquiteturas de redes neurais diferentes, que são:

- Modelo 1: o trabalho Teles (2023) explorou as *RNAs* dos tipos *LSTMs*, as quais foram implementadas na linguagem de programação *Python* pela biblioteca *Tensorflow*.

- Modelo 2: rede CNN apresentada neste trabalho.

O modelo 1 é composto das redes neurais recorrente e é composta por quatro camadas. A primeira camada, ou a camada de entrada é um vetor de 992 posição, onde cada posição é um numero que representa a palavra da notícia. Esses números são interpretados como chaves e que serão utilizados como referencia no dicionario do *Word2Vec*, pois cada *token* é traduzido por um vetor de 100 posições. Assim, a saída da segunda camada é uma matriz de dimensão 992x100. As duas próximas camadas, denominadas de densas, são constituídas de 128 neurônicos, o qual resultou num vetor de 128 posições que será a entrada da última camada. A função de ativação da última camada é a *softmax*, que com apenas dois neurônios, resultará em dois valores possíveis: 0, para notícia falsa, ou 1 para notícia verdadeira.

Acurácia		
%	LSTM	CNN
70,15,15	93.42	89.07

Tabela 4 – Tabela de resultados da métrica acurácia.

Pela tabela 4 é possível observar que o modelo da CNN para a base de dados Fake.br apresentou uma acurácia um pouco menor do modelo da rede neural LSTM. A tabela 5 mostra outras métricas comparativas entre essas duas arquiteturas de redes neurais.

Métricas de Avaliação				
	LSTM		CNN	
%	Classe 0	Classe 1	Classe 0	Classe 1
Recall	92.96	93.95	80.08	97.81
F1-score	93.70	93.13	87.84	90.08
Precisão	92.96	93.95	97.27	83.49

Tabela 5 – Tabela de resultados das métricas de avaliação dos modelos.

A tabela 5 reforça o que foi observado na tabela 2. Isto é, a rede LSTM não apresentou o mesmo comportamento que a rede CNN, pois ao observar os valores obtidos pelas métricas nota-se equilíbrio na classificação das notícias falsas e verdadeiras. Já, o modelo computacional utilizando as redes CNN apenas obteve esse equilíbrio quando foi testado a base de dados 2, que constitui de uma quantidade maior de notícias.

4 Conclusão

Este trabalho salientou a identificação das *fake news* em notícias digitais publicadas na Internet, utilizando um modelo computacional constituído de técnicas de PLN aliada às CNNs, com a justificativa de criar uma ferramenta auxiliar que possa evitar a proliferação de desinformação na mídia digital. Através de uma exploração sistemática de várias metodologias e técnicas, foi possível construir um modelo robusto e eficaz capaz de discernir entre notícias verdadeiras das *fake news*.

Durante todas as etapas de desenvolvimento do sistema foi averiguado as distintas funções e capacidades de cada um dos processos para a construção de um modelo responsável em classificar entre os dois tipos de notícias. A rede neural CNN, escolhida como arquitetura do processo da classificação, apresentou algumas limitações e necessidades para garantir melhores precisões, como por exemplo, a presença da camada de Batch Normalization para garantiram convergência estável e mitigar o *overfitting*.

Por outro lado, a utilização da CNN para reconhecimento de *fake news* se mostrou promissora com acurácias 89.07% e 94.00%, para as respectivas bases de dados testadas, o que representa um bom avanço na análise de texto da língua portuguesa. As redes CNN mostraram valorosas capacidades na extração e agregação de características relevantes.

Todavia, a disparidade entre as duas acurácias citadas, mostram uma limitação das CNNs que dependem de grandes bases de dados para poderem demonstrarem melhor sua capacidade de agregação de características relevantes. Ademais, a falta de uma grande base de dados com notícias verdadeiras e falsas na língua portuguesa do Brasil, também foi um fator limitante. A base é extremamente importante, pois viabilizaria que a rede neural CNN gerasse melhores resultados na extração das características da base de dados e, consequentemente, apresentar um treinamento mais robusto e com uma melhor capacidade de generalização.

Por fim, destaca-se que em trabalhos futuros sejam desenvolvidas bases de dados maiores e que se mantenham atualizadas, para que futuras arquiteturas de redes neurais possam ser treinadas e, principalmente, testadas em notícias aleatórias extraídas da Internet. Além disso, outros trabalhos futuros podem expandir para além dos textos digitais, ou seja, criar modelos computacionais capazes de avaliar imagens e vídeos das notícias em conjunto com o texto para abranger mais fontes, pois estes são os tipos de dados presentes nas mídias sociais.

Referências

BRASIL, C. **4 em cada 10 brasileiros afirmam receber fake news diariamente**. 2022. <https://www.cnnbrasil.com.br/nacional/4-em-cada-10-brasileiros-afirmam-receber-fake-news-diariamente/>. [Online; accessed 25-Setembro-2023]. Citado 2 vezes nas páginas 9 e 16.

ChatGPT. 2022. <https://chat.openai.com/auth/login>. Citado na página 14.

CHOI, E. C.; FERRARA, E. **FACT-GPT: Fact-Checking Augmentation via Claim Matching with LLMs**. 2024. Citado na página 16.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018. Citado na página 13.

FUKUSHIMA, K. Neocognitron. **Scholarpedia**, v. 2, n. 1, p. 1717, 2007. Citado na página 12.

G1. **Fake news: entenda como funciona a fábrica de desinformação política no Brasil**. 2023. <https://g1.globo.com/fato-ou-fake/noticia/2022/10/27/fake-news-entenda-como-funciona-a-fabrica-desinformacao-politica-no-brasil.ghtml>. [Online; accessed 27-Setembro-2023]. Citado 2 vezes nas páginas 9 e 16.

_____. **É #FAKE que vacinas mRNA contra Covid causaram 500 mil mortes nos EUA**. 2023. <https://g1.globo.com/fato-ou-fake/coronavirus/noticia/2023/05/08/e-fake-que-vacinas-mrna-contra-covid-causaram-500-mil-mortes-nos-eua.ghtml>. [Online; accessed 27-Setembro-2023]. Citado na página 9.

GARCIA, G.; AFONSO, L.; PAPA, J. Fakerecogna: A new brazilian corpus for fake news detection. In: _____. [S.l.: s.n.], 2022. p. 57–67. ISBN 978-3-030-98304-8. Citado 2 vezes nas páginas 10 e 19.

GAUTAM, H. **Word Embedding: Basics**. 2020. <https://medium.com/@hari4om/word-embedding-d816f643140>. [Online; accessed 5-Abril-2024]. Citado 2 vezes nas páginas 5 e 14.

GLOBO, O. **O GLOBO lança 'Fato ou Fake' para checagem de conteúdo suspeito**. 2018. <https://oglobo.globo.com/fato-ou-fake/o-globo-lanca-fato-ou-fake-para-checagem-de-conteudo-suspeito-22930724>. [Online; accessed 5-Abril-2024]. Citado na página 16.

GUARISE, L.; REZENDE, S. O. **Detecção de notícias falsas usando técnicas de deep learning**. Dissertação (Mestrado) — Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo, 2019. Citado 2 vezes nas páginas 9 e 17.

HU, L.; WEI, S.; ZHAO, Z.; WU, B. Deep learning for fake news detection: A comprehensive survey. **AI Open**, v. 3, p. 133–155, 2022. ISSN 2666-6510. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2666651022000134>. Citado na página 16.

HUTCHINS, W. J. The georgetown-ibm experiment demonstrated in january 1954. In: FREDERKING, R. E.; TAYLOR, K. B. (Ed.). **Machine Translation: From Real Users to Research**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004. p. 102–114. ISBN 978-3-540-30194-3. Citado na página 13.

KALIYAR, R. K.; GOSWAMI, A.; NARANG, P.; SINHA, S. Fndnet–a deep convolutional neural network for fake news detection. **Cognitive Systems Research**, Elsevier, v. 61, p. 32–44, 2020. Citado 6 vezes nas páginas 6, 9, 13, 16, 17 e 21.

KIM, Y. Convolutional neural networks for sentence classification. In: **Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)**. Doha, Qatar: Association for Computational Linguistics, 2014. p. 1746–1751. Disponível em: <https://aclanthology.org/D14-1181>. Citado 3 vezes nas páginas 5, 12 e 13.

LECUN, Y.; BOTTOU, L.; BENGIO, Y.; HAFFNER, P. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 86, n. 11, p. 2278–2324, 1998. Citado na página 12.

LUO, P.; WANG, X.; SHAO, W.; PENG, Z. Towards understanding regularization in batch normalization. **CoRR**, abs/1809.00846, 2018. Disponível em: <http://arxiv.org/abs/1809.00846>. Citado na página 22.

MIKOLOV, T.; CHEN, K.; CORRADO, G.; DEAN, J. **Efficient Estimation of Word Representations in Vector Space**. 2013. Citado 2 vezes nas páginas 5 e 15.

MONTEIRO, R. A.; SANTOS, R. L. S.; PARDO, T. A. S.; ALMEIDA, T. A. de; RUIZ, E. E. S.; VALE, O. A. Contributions to the study of fake news in portuguese: New corpus and automatic detection results. In: **Computational Processing of the Portuguese Language**. [S.l.]: Springer International Publishing, 2018. p. 324–334. ISBN 978-3-319-99722-3. Citado 5 vezes nas páginas 9, 10, 17, 19 e 24.

PAíS, E. **Como a desinformação influenciou nas eleições presidenciais?** 2018. <https://brasil.elpais.com/brasil/2018/02/24/internacional/1519484655_450950.html>. [Online; accessed 29-Outubro-2023]. Citado na página 16.

RUSSAKOVSKY, O.; DENG, J.; SU, H.; KRAUSE, J.; SATHEESH, S.; MA, S.; HUANG, Z.; KARPATHY, A.; KHOSLA, A.; BERNSTEIN, M.; BERG, A. C.; FEI-FEI, L. ImageNet Large Scale Visual Recognition Challenge. **International Journal of Computer Vision (IJCV)**, v. 115, n. 3, p. 211–252, 2015. Citado na página 13.

SILVA, R. M.; SANTOS, R. L.; ALMEIDA, T. A.; PARDO, T. A. Towards automatically filtering fake news in portuguese. **Expert Systems with Applications**, Elsevier, v. 146, p. 113199, 2020. Citado na página 9.

SZEGEDY, C.; LIU, W.; JIA, Y.; SERMANET, P.; REED, S.; ANGUELOV, D.; ERHAN, D.; VANHOUCKE, V.; RABINOVICH, A. **Going Deeper with Convolutions**. 2014. Citado na página 13.

TELES, G. **Uso de Redes Neurais Profundas para Detecção de Fake News**. Dissertação (Mestrado) — Faculdade de Computação - Universidade Federal de Uberlândia, 2023. Citado na página 24.

YU, F.; LIU, Q.; WU, S.; WANG, L.; TAN, T. A convolutional approach for misinformation identification. In: **Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17**. [s.n.], 2017. p. 3901–3907. Disponível em: <https://doi.org/10.24963/ijcai.2017/545>. Citado na página 9.