

**COORDENADORIA DE ENGENHARIA COMPUTAÇÃO**

**GUILHERME PROENÇA CRAVO DA COSTA**

**RENATO DRUZIAN**

**SUMARIZAÇÃO DE NOTÍCIAS**

**Sorocaba/SP**

**2020**

**Guilherme Proença Cravo da Costa**

**Renato Druzian**

**SUMARIZAÇÃO DE NOTÍCIAS**

Trabalho de conclusão de curso apresentado ao Centro Universitário Facens como exigência parcial para obtenção do diploma de graduação em Engenharia da Computação.

Orientador: Prof. Johannes von Lotcher

**Sorocaba/SP**

**2020**

FICHA CATALOGRAFICA

ELABORADA PELA “BIBLIOTECA FACENS”

SXXXs

Sobrenome, Nome do Aluno 1; Sobrenome, Nome do Aluno 2; Sobrenome,

Título do trabalho: subtítulo se houver/ por Nome do Aluno.- Sorocaba, SP: [s.n.], 2020.

XX f.; 29cm.

Trabalho de Conclusão de Curso (Graduação) – Centro Universitário Facens, Coordenadoria de Engenharia de Computação – Curso de Engenharia de Computação, 2020.

Orientador: Prof.

1. Palavra-chave. 2. Palavra-chave. 3. Palavra-chave. I. Faculdade de Engenharia de Sorocaba. II. Título.

CDD XXX

**Guilherme Proença Cravo da Costa**

**Renato Druzian**

**SUMARIZAÇÃO DE NOTÍCIAS**

Trabalho de conclusão de curso apresentado ao Centro Universitário Facens como exigência parcial para obtenção do diploma de graduação em Engenharia da Computação.

Orientador: Prof. Johannes von Lotcher

Sorocaba, XX de XXXXXXXXXX de 2020

BANCA EXAMINADORA

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Esp/Me/Dr e Nome

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Esp/Me/Dr e Nome

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Esp/Me/Dr e Nome

**AGRADECIMENTOS**

Frase de um autor que representa o espírito desse trabalho. Frase entre aspas duplas.

Nome do autor

**RESUMO**

**Palavras-Chave**:

**ABSTRACT**

**Key-words**:

**LISTA DE ILUSTRAÇÕES**

[Figura 1 -](#_Toc40185531) *[CboW](#_Toc40185531)* [18](#_Toc40185531)

[Figura 2 -](#_Toc40185532) *[Skip Gram](#_Toc40185532)* [19](#_Toc40185532)

**LISTA DE TABELAS**

[Tabela 1 – Exemplo de vetor](#_Toc41147217) *[Bag of Words](#_Toc41147217)* [16](#_Toc41147217)

**LISTA DE SIGLAS**

IA Inteligência Artificial

BoW *Bag of Words*

S2S *Sequence to sequence*

GloVe *Global Vectors for Word Representations*

CBoW *Continous Bag of Word*

Json *JavaScript Object Notation*

**SUMÁRIO**

[1 INTRODUÇÃO 13](#_Toc41145484)

[2 REPRESENTAÇÃO DE TEXTO 15](#_Toc41145485)

[2.1](#_Toc41145486) *[Bag of Words](#_Toc41145486)* [15](#_Toc41145486)

[2.2](#_Toc41145487) *[Word Embeddings](#_Toc41145487)* [16](#_Toc41145487)

[2.3](#_Toc41145488) *[Skip Gram](#_Toc41145488)* [17](#_Toc41145488)

[2.4](#_Toc41145489) *[Cbow](#_Toc41145489)* [18](#_Toc41145489)

[2.5](#_Toc41145490) *[GloVe](#_Toc41145490)* [19](#_Toc41145490)

[3 MÉTODOS DE SUMARIZAÇÃO 20](#_Toc41145491)

[3.1](#_Toc41145492) *[Sequence to Sequence](#_Toc41145492)* [20](#_Toc41145492)

[4 METODOLOGIA 20](#_Toc41145493)

[5 RESULTADOS 20](#_Toc41145494)

[6 CONCLUSÃO 20](#_Toc41145495)

REFERÊNCIAS.......................................................................................21

# INTRODUÇÃO

O mundo está cada vez mais repleto de informações não-estruturadas, principalmente texto. Mídias sociais, como Twitter e Facebook, tiveram alto crescimento nos últimos anos e influenciam diariamente com opiniões e notícias.

Sites de notícias são fontes provedoras de informações muitas vezes confiáveis, mas o volume de notícias nem sempre é possível de ser acompanhado por uma pessoa ocupada. (Rino & Pardo, 2003) ”...viajar pelas páginas de notícias a fim de apreender o que é essencial exige tempo, capacidade de identificar o que é relevante, no grande volume de informações disponível, e capacidade de mentalizar, de forma coerente, o conteúdo essencial...”.

Máquinas começaram a ser empregadas para realizar tarefas que antes eram das pessoas, como secretarias que resumem notícias financeiras para os patrões ou agentes de *home brokers* geram insights para investidores, que possibilitou a diminuição do tempo de muitos processos.

Em IA, uma das técnicas mais recentes para tratar de sumarização de texto são redes neurais recorrentes. Este método possui aplicações para solucionar algumas análises de sentimento, entidades nomeadas e sumarização de texto.

Primeiramente, para a sumarização de texto, há a coleta das notícias que para utilização como entrada na rede neural, nessa coleta são captados os títulos e os textos das notícias onde o título fica sendo nosso parâmetro de comparação para a saída da rede neural.

Posteriormente, na etapa de teste, que consiste em dividir os dados coletados em treino e teste, esses dois grupos podem ter tamanho que for necessário. A divisão mais comum é 70% para treino e 30% para teste, a rede aprenderá com o grupo de treino e o resultado obtido pela mesma será analisado com o grupo de teste.

A rede neural aprenderá lendo o texto da notícia e fará uma ligação com o título, já na etapa de teste lerá as notícias do grupo que ela não conhece os dados e tentará gerar um título, ao término será realizado uma análise dessa saída com o título original, com isso será feito a análise, de acordo com (Ferneda, 2006) ”As redes neurais artificiais se diferenciam pela sua arquitetura e pela forma como os pesos associados às conexões são ajustados durante o processo de aprendizado”.

Entretanto, deve-se observar que se a maioria das notícias na parte de treinamento possuírem títulos sensacionalistas ou que não condizem com as reais informações apresentadas no texto, será gerado resultados não confiáveis, podendo com isso gerar um algoritmo enviesado, ou seja, que pensa de forma muito parecida com a fonte das notícias.

Para evitar esse problema e garantir melhor assertividade, é necessário um grande volume de dados e de várias fontes, evitando assim criar algum viés na rede neural.

Por fim, a área científica poderá beneficiar-se para melhor inserção de títulos em artigos, assim como resenhas ou textos gerais em instituições acadêmicas, pois a sumarização possibilita que mais textos sejam lidos em menor tempo, sem que haja perda no sentido para que a mensagem seja transmitida.

# REPRESENTAÇÃO DE TEXTO

Texto e palavras como o ser humano compreende, um computador não consegue entender, para isso, é necessário realizar algumas tratativas para que o computador possa entender palavras e textos.

Realizar apenas o pré processamento do texto, não quer dizer que o computador consiga entender ainda, pois pode-se utilizar sinônimos para manter o mesmo contexto da frase ou texto (Kryscinski, Keshar, McCAnn, Xiong, & Socher, 2019) (Luo, Xu, & Guo, 2014).

Outras problematizações encontradas são na análise semântica, de contexto e léxica, esses são relacionados, por exemplo, que robôs humanoides não entenderam sarcasmo.

O campo da inteligência artificial, que estuda a forma como a comunicação e trabalho são realizados para que máquinas possam comunicar-se com as pessoas, é chamado de Processamento de Linguagem Natural, que por muitas vezes é usando na aprendizagem profunda com o uso de redes neurais. Alguns exemplos de uso do PLN são assistentes pessoais e *chatbots*.

## Bag of Words

Este método é o mais simples, mas com a possibilidade de resultados satisfatórios dependendo do vocabulário. Vocabulário é um conjunto de palavras, onde cada uma delas representa uma amostra de palavra contida no texto, e com esse vocabulário é que realiza-se a análise de outros textos.

Uma análise é iniciada com a criação do vocabulário a partir da primeira amostra, então, cria-se um vetor com 0 e 1, que indica se uma palavra do vocabulário está presente no texto analisado ou não, porém dessa forma é perdido a posição dela na frase e o seu contexto também é alterado (Brownlee, 2017).

Para ficar mais claro, abaixo um exemplo com duas frases:

1. A FACENS é nota máxima no Enade
2. A FACENS conseguiu obter nota 5 no Enade

Com essas duas frases pode-se criar o vocabulário, o conjunto fica montado da seguinte forma {A, FACENS, nota, conseguiu, obter, máxima, Enade, 5, é, no}. Agora com o vocabulário monta-se o vetor de 1s e 0s, que fica da seguinte forma:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | FACENS | nota | conseguiu | obter | máxima | Enade | 5 | é | no |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |

Tabela 1 – Exemplo de vetor *Bag of Words*

Com essa tabela é possível visualizar como seria uma saída do método *BoW*, onde a primeira linha é representada pelo vocabulário, e cada linha a mais na tabela representa uma frase.

Pode-se observar mais um problema nesse método, quanto maior for o vocabulário criado, maior será a tabela criada, uma vez que o total de linhas sempre será a quantidade de palavras no vocabulário.

## Word Embeddings

Este método é amplamente utilizado nos dias atuais, e consiste em converter cada palavra em um vetor e com isso agrupa palavras similares próximas umas das outras.

Algumas das vantagens deste método são análise semântica e de contexto de uma determinada palavra em um texto, outra forma de análise é a verificação do radical, prefixo e sufixo de uma palavra, com isso é possível encontrar sinônimos para uma palavra.

Para este tipo de processamento são usadas redes neurais para identificar e compreender qual o contexto que a palavra está inserida, e com essa possibilidade é possível resolver problemas como recuperação de informação, classificação de documentos e respostas a perguntas Bolukbasi et al (2016), Kryscinski et al. (2019), Huang et al (2012).

Como todos os métodos anteriores apresentados, esse também possui desvantagem, para que o modelo possa aprender melhor são necessários uma grande quantidade de textos, com isso é possível analisar muitas palavras em diversos contextos e em diferentes posições semântica, quanto maior o vocabulário melhor será o resultado do modelo e maior será o tempo de treino do modelo.

## Skip Gram

Conforme estudo de Chen et al. o *Skip Gram* é um método baseado em vetores, ou seja, cada palavra é transformada em um vetor e com a finalidade de prever quais são as combinações de palavras disponíveis para uso antes e depois da palavra, dado a mesma como entrada.

Assim, é possível analisar o contexto em que a palavra está inserida para encontrar essas combinações. Outro uso deste método é, dado dois vetores e realizar a soma deles, gerar na saída um terceiro vetor com relevância em relação a soma dos dois vetores, como no exemplo abaixo:

Nesse exemplo, realiza-se a conexão das palavras Brasil e Capital pode-se notar que a saída dele é Brasília.

Outro ponto a ser observado é que, ao treinar um modelo com *Skip Gram*, encontra-se uma combinação de palavras que tenha sentido no contexto.

Dado uma sequência de palavras de treino, , ,…,, o *Skip Gram* tem o objetivo de aumentar a probabilidade de uma palavra fazer sentido no contexto de acordo com a Equação 1.

(1)

Onde tem-se que é o tamanho do vocabulário para treino, quanto maior for , maiores são os exemplos de treino, maior o tempo de treinamento e, consequentemente, melhor o resultado do modelo.

A base da fórmula do *Skip Gram* é definida por , que também é escrita como a Equação 2:

(2)

São as representações dos vetores de entradas e saídas dadas por e da palavra , e é o número de palavras no vocabulário, ou seja, o número total de palavras presentes no conjunto de treino.

## Cbow

Como (Luo, Xu, & Guo) disse em seu trabalho, *CBoW* é mais uma técnica de representação de texto que utiliza vetores para simbolizar palavras, esta técnica também faz uso de redes neurais recorrentes para avaliar e prever uma palavra em um determinado contexto.

De uma forma bem simples pode-se dizer que o *CBoW* faz o caminho inverso do *Skip Gram,* onde o último quer, através de uma palavra prever toda a frase, enquanto o *CBoW* possui a frase mas com uma palavra faltando, e o mesmo pretende prever essa palavra faltante.

O uso de redes neurais recorrentes se enquadra, pois as palavra antes são representadas como sendo o passado e as palavra após a lacuna como sendo o futuro, e como lacuna a palavra que deseja-se prever.

Abaixo há duas imagens que mostram de maneira mais clara a diferença entre eles na Figura 1 e Figura 2.

Figura 1 - *CBoW*

Uma imagem contendo objeto, relógio

Descrição gerada automaticamente

Fonte: Disponível em: https://arxiv.org. Acesso em 12 mai 2020

Figura 2 - *Skip Gram*

Uma imagem contendo objeto, relógio

Descrição gerada automaticamente

Fonte: Disponível em: https://arxiv.org. Acesso em 12 mai 2020

## GloVe

Conforme estudos realizados por (Pennington, Socher, & Manning, 2014) e (Misra) o modelo GloVe, ou *Global Vectors for word Representation*, preserva o contexto global e cria uma matriz de co-ocorrência para estimar se uma palavra vai co-ocorrer junto com outra.

Este modelo é utilizado em problemas de sumarização de textos, para responder perguntas, verificação de fraudes, recuperação de informação.

Quanto maior for o volume de dados ou dicionário criado, melhor será sua saída. Por outro lado, não se pode esquecer que quanto maior o volume de dados para treinar o modelo, maior será o tempo que a rede neural necessita para finalizar o treinamento.

# MÉTODOS DE SUMARIZAÇÃO

## Sequence to Sequence

Genericamente, diversas arquiteturas partem do princípio de *Sequence to Sequence* ou, também conhecido como codificador-decodificador, para servir de base a redes neurais recorrentes, que consiste em processar uma sequência de entrada com o codificador e gerar uma sequência de saída com o decodificador. Para isso, o decodificador calcula o novo estado oculto e incorpora a palavra ao idioma em questão com base no que foi processado anteriormente, e a partir da condição de entrada e saída do codificador.

O funcionamento é parecido com o cérebro humano, no qual uma sequência de informações são mais simples de lembrar do que a partir do meio ou do final para o começo como, por exemplo, a letra de uma música ou um caminho para chegar em algum local. Já no treinamento de uma rede neural recorrente, há maior eficiência e vantagem para prever o próximo passo a ser realizado pois suas camadas ocultas apenas de acordo com os dados de entrada, são baseadas em um fluxo de informações dado por:



Em outras situações, é possível criar camadas ocultas a partir da entrada, assim como, criar camada de saída com a oculta. Este processo muda como a memória guarda as informações, já que há uma combinação entre os dados da entrada no tempo atual com o tempo anterior da camada oculta, como nos exemplos:





A camada oculta sofre diversas alterações ao longo de cada passo de tempo a cada nova entrada, e a única forma de acesso é através de uma sequência correta de entradas. Na prática, a camada oculta "lembra" em qual parte do tempo encontra-se de acordo com os tempos anteriores e consegue diferenciar, por exemplo, textos idênticos de uma música e prever a próxima palavra.

# METODOLOGIA

# RESULTADOS

# CONCLUSÃO

**REFERÊNCIAS**

Adelson, P., Arora, S., & Hara, J. (s.d.). Abstractive text summarization with Quasi-Recurrent neural netwaorks. p. 7.

Bolukbasi, T., Chang, K.-W., Zour, J., Saligrama, v., & Kalai, A. (2016). Man is to computer programmer as woman is to homemaker? Debiasing embeddings. *30th Conference on Neural Information Processing Systems (NIPS 2016)*, (p. 9). Barcelona.

Brownlee, J. (2017). *Machine Learning Mastery*. Fonte: Machine Learning Mastery: https://machinelearningmastery.com/gentle-introduction-bag-words-model/

Chen, K., Corrado, G., Dean, J., Tomas, M., & Sutskever, I. (s.d.). Distribuited representations of words and phrases and their compositionality., (p. 9).

Ferneda, E. (2006). *Redes neurais e sua aplicação em sistemas de recuperação de.* Ribeirão Preto.

Huang, E. H., Socher, R., Manning, C. D., & Ng, A. Y. (2012). Improving word representations via global context and multiple word prototypes. *50th Annual Meeting of the Association for Computational Linguistics.* Jeju: 10.

Kryscinski, W., Keshar, N. S., McCAnn, B., Xiong, C., & Socher, R. (2019). Neural text summarization: A critical evaluation., (p. 13).

Luo, Q., Xu, W., & Guo, J. (2014). A study on the CBOW model's overfitting and stability., (p. 4). Shangai.

Paulus, R., Xiong, C., & Socher, R. (2017). A deep reinforced model for abstractive summarization., (p. 12). Palo Alto.

Rino, L., & Pardo, T. (2003). A sumarização automática de textos principais características.