SUMARIZAÇÃO EXTRATIVA BASEADA EM PROCESSAMENTO DE LINGUAGEM NATURAL

MOURA, Caique Frederico Pereira de caique.moura@fatec.sp.gov.br Fatec Jacareí – Professor Francisco de Moura
CARVALHO, Fabricio Galende Marques de fabricio.carvalho01@fatec.sp.gov.br Fatec Jacareí – Professor Francisco de Moura

1. INTRODUÇÃO

Um dos atuais problemas do mundo em rede é a vasta quantidade de informação disponível e o pouco tempo que os usuários têm para consumi-la. A sumarização de textos se faz necessária ao auxiliar o usuário a compreender com mais agilidade aquilo que ele deseja tomar conhecimento.

2. METODOLOGIA

Foi definido a utilização de uma pipeline de processamento de linguagem natural para a base do funcionamento do algoritmo, sendo esta, uma sequência de tratativas sobre o texto de entrada para que estes sejam passíveis de um processamento normalizado, os artifícios da pipeline são utilizados para converter o texto em caixa baixa, tokenização de palavras e frases, remoção de caracteres repetidos da linguagem (*Stop Words*) e correção de escrita. Alternadamente, foi planejado que a estrutura da aplicação seria orientada a microsserviços, assim sendo, o código foi modularizado para que fosse distinguível o microsserviço de sumarização e o microsserviço de raspagem de dados, assim sendo possível a substituição simplificada, esta é considerada uma boa prática do desenvolvimento de software.
A extração dos dados é feita utilizando-se de bibliotecas de raspagem de dados (*Data Scraping*), em específico, a biblioteca *BeautifulSoup*, para a devida extração do conteúdo web, utilizando esta biblioteca é possível fazer a extração da carga textual de páginas web, dados estes que serão enviados para o microsserviço responsável pela sumarização posteriormente.
Para o algoritmo da sumarização, primeiramente é aplicado a pipeline base de processamento de linguagem natural para normalizar os dados de entrada, após, baseando-se em algoritmos de *TextRank* [1], é definido o modelo de valoração utilizando um léxico e o conceito de *BagOfWords* [2], o léxico acumula todas as palavras inclusas no corpo de texto, desta forma, representando em um dicionário a totalidade do conteúdo das sentenças envolvidas. Com base no léxico, é possível criar vetores binários das frases representando ou não a presença de palavras do léxico na mesma, basicamente, se uma palavra da frase está presente no dicionário, esta é reposta pelo valor 1 no vetor, se não, é reposta por 0, o processo é repetido para cada palavra da frase, definindo o vetor binário que representa aquela frase.
Consumindo os resultados dos vetores binários das frases, foi implementado um cálculo baseado no modelo de similaridade com cosseno [3], onde, os vetores binários são comparados um a um no plano vetorial, desta forma, dentre um intervalo de $0 \leq \cos(\theta) \leq 1$, quanto mais próximo de 1 maior a semelhança entre vetores, o mesmo vale para a proximidade de 0, indicando oposição. Os resultados das comparações são somados, e, ao final, o vetor com maior score é ranqueado em primeiro, o segundo com maior valor em segundo ...

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

Figura 01 – Modelo de similaridade com cosseno

Fonte: Paiva,I. (2019)

Após a construção do mecanismo sumarizador, foi acoplado uma estrutura em e*js (*Embedded Javascript*), utilizada para criar uma interface de visualização onde um usuário pode inserir uma URL, selecionar o range da sumarização, que filtra a quantidade de sentenças retornadas pela resposta e um botão para enviar a requisição ao servidor.

3. RESULTADOS E DISCUSSÕES

O produto deste trabalho foi uma ferramenta capaz de receber texto proveniente de uma raspagem de dados na web, processá-lo e retornar um sumário abrangendo o conteúdo mais relevante contido, que para melhor visualização é exibido em uma interface gráfica, possibilitando testes do usuário.

Figura 02 – Interface de interação

4. Conclusões
É possível observar que a sumarização de textos vem sendo de profundo impacto quando se necessita analisar grandes quantidades de dados, e existe uma crescente demanda por soluções ágeis para que a simples pesquisa de um usuário sobre uma informação tenha uma resposta mais rápida e útil.

REFERÊNCIAS

[1] SARKAR, Dipanjan. Text analytics with python, 2nd ed, New York, Apress, 2019.
[2] CARVALHO, F. G. M. de. Modelos para processamento de linguagem natural, p. 10, 2024.
[3] GUNASUNDARI, S., SHYLAJA J. M. et. al. Eur. Chem. Bull. (Special Issue 6), p. 4654, dez. 2023

AGRADECIMENTOS

À instituição Centro Paula Souza e Fatec Jacareí pela bolsa e apoio, e ao orientador pelo direcionamento fornecido.