



**SERVIÇO PÚBLICO FEDERAL  
UNIVERSIDADE FEDERAL DE SERGIPE  
PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA**

PROGRAMA DE INICIAÇÃO CIENTÍFICA VOLUNTÁRIA - PICVOL

## **Processamento de Linguagem Natural no âmbito do Comércio Eletrônico**

Área do conhecimento: Ciências Exatas e da Terra  
Subárea do conhecimento: Ciência da Computação  
Especialidade do conhecimento: Metodologia e Técnicas da Computação

Relatório Final  
Período da bolsa: de (setembro/2021) à (agosto/2022)

Este projeto é desenvolvido com bolsa de iniciação científica  
voluntária

PICVOL

# Sumário

<b>1</b>	<b>Introdução</b>	<b>3</b>
<b>2</b>	<b>Objetivos</b>	<b>3</b>
<b>3</b>	<b>Metodologia</b>	<b>4</b>
3.1	TF-IDF . . . . .	4
3.2	Google PEGASUS . . . . .	5
3.3	MT5 . . . . .	7
3.4	Base de dados . . . . .	7
3.5	Métrica ROUGE . . . . .	7
<b>4</b>	<b>Resultados e discussão</b>	<b>8</b>
<b>5</b>	<b>Perspectivas de Futuros Trabalhos</b>	<b>12</b>
<b>6</b>	<b>Conclusões</b>	<b>12</b>
<b>7</b>	<b>Referências bibliográficas</b>	<b>12</b>

## 1 Introdução

O Processamento de Linguagens Naturais (PLN) é uma área de estudo da Ciência da Computação que conversa com diversas outras áreas do conhecimento a fim de interpretar a linguagem dos seres humanos. Para isso, utiliza de diversas técnicas, desde métodos estatísticos ao uso de inteligência artificial [1]. O estudo e início do desenvolvimento dessa área teve início na década de 1950 com a união de métodos computacionais à linguística voltada para tradução de textos. Por englobar diversas áreas do conhecimento, o PLN exige que os pesquisadores e desenvolvedores ampliem significativamente sua base de conhecimento multidisciplinar.

Inspirado por esse vasto campo do conhecimento, esse projeto de pesquisa foi desenvolvido visando explorar as diversas técnicas e ideias que envolvem esse campo do PLN. Para isso, foi escolhido para aplicar as tecnologias e conhecimentos aqui desenvolvidos o portal de divulgação científica Saense<sup>2</sup>. Pois, o mesmo possui como objetivo tornar conteúdos científicos de ponta mais fácil e simples de entender, possibilitando que uma maior parte da população brasileira possa compreender e se informar sobre o que há de mais novo na ciência. Portanto, este projeto de pesquisa alia-se com esse objetivo almejando contribuir positivamente para com o portal.

## 2 Objetivos

Esse projeto tem como principal objetivo sumarizar os artigos encontrados na plataforma de divulgação científica Saense visando sua publicação em redes sociais, como o Instagram, a fim de disseminar o conhecimento científico tornando-o mais acessível à população. É perceptível que as redes sociais são plataformas muito utilizadas pelos brasileiros e os textos ali presentes nas publicações apresentam-se de maneira bastante resumida. Portanto, como grande parte das pessoas está habituada a textos curtos, surge a necessidade de sumarizar os artigos do portal, possibilitando que os mesmos sejam divulgados nessas redes alcançando um maior público. Dessa forma, este projeto alinha-se com o objetivo do portal: a disseminação do conhecimento científico de maneira acessível a todos<sup>3</sup>.

Em suma, este projeto de pesquisa tem como intuito explorar as diferentes técnicas e visões dentro do vasto campo do Processamento de Linguagens Naturais. Com base nisso, será implementado dois tipos de algoritmos de sumarização e será realizado a comparação de seus resultados através de uma métrica de avaliação de qualidade de resumos. Além disso, esse projeto abre diversas “portas” para trabalhos futuros, seja na continuidade e aprimoramento do mesmo ou na contribuição em implementações de outros trabalhos.

---

<sup>2</sup><https://saense.com.br/>

<sup>3</sup><http://saense.com.br/sobre-saense-2/sobre>

### 3 Metodologia

Para o desenvolvimento deste projeto inicialmente foi estudado diversas técnicas de Processamento de Linguagem Natural. Com intuito de comparar os resumos obtidos foram escolhidas técnicas divergentes entre si, uma possuindo abordagem extrativa e outra abstrativa. Além disso, em paralelo, foram vistas métricas de avaliação de qualidade de resumos e foram escolhidas, para esse trabalho, a métrica "ROUGE". Dessa forma, ao final teremos material para realizar a comparação e avaliação da qualidade de um resumo gerado por um Humano, um gerado por algoritmo extrativo e outro por uma abordagem abstrativa. Aliado a isso, foi adotado a linguagem de programação Python, por ser uma linguagem prática e com muitos recursos já existentes e disponíveis na área de PLN.

Seguindo essa linha de raciocínio, foi criado um repositório no Github <sup>4</sup> no qual contém todo o código do projeto juntamente com as devidas instruções e dicas para execução dos algoritmos, e também houve a criação do website do projeto <sup>5</sup>. Tal website tem como intuito documentar todo o processo, estado atual, os resultados parciais e finais desse estudo em Processamento de Linguagem Natural. O objetivo é conseguir desenvolver bons algoritmos de sumarização de texto com diferentes abordagens. Dessa forma, terá sido explorado e avaliado diferentes técnicas e pontos de vistas na área de PLN, expandindo e aprimorando conhecimento dos envolvidos e outras pessoas que se interessam pelo assunto.

#### 3.1 TF-IDF

Segundo Anderson Cavalcanti em sua dissertação de mestrado, o termo "TF-IDF" vem do inglês Term Frequency - Inverse Document Frequency, ou seja, Frequência do Termo - Inverso da Frequência dos Documentos. Esse termo representa um valor, obtido através de métodos estatísticos, que tem o intuito de determinar a importância de uma palavra num texto ou em um conjunto de documentos [6]. Tal técnica é muito utilizada na recuperação de informações, na mineração de dados, consultar documentos similares (com cálculo de distância), transformar os textos para alimentar um modelo de aprendizagem de máquina e dentre outras aplicações. O valor "TF" de uma palavra aumenta de maneira proporcional à medida que o seu número de ocorrências aumenta no texto, no entanto, esse valor é equilibrado pelo "IDF" que representa o quão rara é a palavra no texto ou documentos. Portanto, temos:

$$TF = \frac{\text{Número de vezes que o termo "t" aparece no texto}}{\text{Número de termos totais no texto}}$$

$$IDF = \log \frac{\text{Número de documentos ou sentenças no texto}}{\text{Número de documentos em que o termo apareceu}}$$

Portanto, é importante ressaltar que quanto maior o TF-IDF mais raro será o termo, e consequentemente, quanto menor o valor, mais comum será o mesmo.

---

<sup>4</sup><https://github.com/0xdferraz/Saense-PLN>

<sup>5</sup><https://0xdferraz.github.io/Saense-PLN/>

Exemplo:

- Suponha um documento com 100 palavras no qual o termo “IA” aparece 5 vezes. Portanto:  $TF = 5/100 = 0.05$
- Se tivermos 100 documentos ao total e o termo “IA” aparece em 20 desses documentos, logo:  $IDF = \text{Log}(100/20) = 0.69$
- Por fim, o valor TF-IDF do termo “IA” é:  $TF-IDF = 0.05 * 0.69 = 0.034$

Foi utilizado essa técnica para construir o algoritmo “summary\_tf-idf.py”. Assim, foi possível definir quais as sentenças do artigo são as mais importantes, extraí-las e juntá-las em um arquivo, obtendo a sumarização do mesmo.

### 3.2 Google PEGASUS

Diferente do método já abordado, “TF-IDF”, no qual apenas extraímos as sentenças mais importantes do texto, uma abordagem abstrativa possibilita obter um resultado totalmente novo a partir do texto de entrada. Logo:



Figura 1: Representação visual do funcionamento do modelo.

O objetivo é utilizar um modelo “Encoder-Decoder” (Codificador-Decodificador) que apresenta uma arquitetura sequencial (“Seq2Seq”) baseado em Redes Neurais Recorrentes (“RNN”), ou seja, o objetivo é converter uma sequência de palavras em outra sequência. Assim:

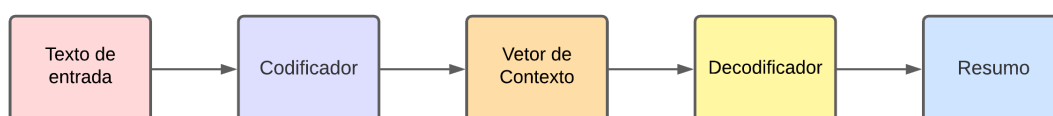


Figura 2: Representação visual detalhada do funcionamento do modelo.

Para isso, o “Encoder” irá analisar o texto dado como input e irá gerar o que chamamos de “Context vector” (vetor de contexto) que é uma representação numérica do texto. Ao passar os dados para o “Decoder” seu trabalho é decodificar o vetor de contexto de forma a produzir o resumo desejado.

Para que o modelo funcione adequadamente precisamos treiná-lo, mas para isso precisamos de muitos dados, são eles os textos e seus respectivos resumos. Para se ter uma idéia podemos ver a quantidade de dados necessários para treinar alguns modelos feitos pela Google:

Modelo	Quantidade de dados
CNN	90k
NYT	97k
DailyMail	197k
XSum	204k

Figura 3: Quantidade de dados utilizados para treinamento de alguns modelos da Google.

Seria uma tarefa árdua encontrar uma quantidade de dados semelhante. Então, a ideia proposta pelo “Google Pegasus” é de grande valia, pois reúne diversos modelos pré-treinados com milhares de dados. Dessa forma, podemos utilizá-los e realizar apenas um ajuste fino para obter resultados com uma boa acurácia.

No artigo do Pegasus é descrita a hipótese utilizada para obter essa grande quantidade de dados necessários para treinar o modelo. Foi coletado diversos textos da internet (sites de notícias, reddit, blogs, etc) e a partir daí é possível criar um ranking das sentenças mais importantes e é escolhida a que possui maior nota, de maneira similar a sumarização extrativa. O grande diferencial é que ao coletar a sentença removemos a mesma do texto original. Assim, tal sentença representaria o resumo abstrativo. Ou seja, teremos os dados necessários para que o algoritmo “aprenda” a parafrasear e gerar sumarização abstrativa [7]. Assim foram treinados os modelos disponibilizados pela Google, e podemos fazer um processo parecido para gerar os dados para o ajuste fino necessário para o modelo se adequar às nossas necessidades. De acordo com o artigo, necessitamos de aproximadamente 1000 dados (textos e seus resumos) para tal tarefa, um número muito menor de dados comparando com o que seria necessário para treinar do zero um modelo. Por fim, para implementar o algoritmo utilizaremos como apoio o site Hugging Face<sup>8</sup>, que provê os modelos e suas documentações facilitando sua utilização.

Apesar das vantagens apresentadas, no decorrer da implementação utilizando o Google Pegasus, foram constatados alguns problemas que dificultaram a sua utilização, como por exemplo a necessidade de traduzir o texto para o inglês e traduzi-lo novamente para o português o resumo obtido, pois o modelo só aceita entrada no idioma inglês. E esse processo de tradução gerava alguns erros residuais, que inicialmente não atrapalharam o desenvolvimento do projeto, porém com o passar do tempo a biblioteca utilizada para tradução<sup>9</sup> apresentou alguns erros que impediam o algoritmo de executar e a implementação foi temporariamente descartada. Talvez, em um outro momento deste projeto de pesquisa possa ser retomado sua aplicação.

<sup>8</sup><https://huggingface.co/>

<sup>9</sup><https://pypi.org/project/googletrans/>

### 3.3 MT5

Com isso em mente, iniciou-se a busca por outro modelo pré-treinado abstrativo que não necessite utilizar a tradução do texto, evitando assim erros residuais, além de possíveis erros com a biblioteca de tradução como ocorreu no Google Pegasus. Dessa forma, foi encontrado o algoritmo “mT5\_multilingual\_XLSum” disponível também no website “Hugging Face” publicado por “csebuethlp”<sup>10</sup>. Trata-se de um algoritmo originalmente desenvolvido pela Google, mas que foi ajustado utilizando a base de dados “XL-Sum” que, segundo os autores, é um conjunto de dados abrangente e diversificado que compreende 1 milhão de pares de artigos-resumo com anotações profissionais da BBC, extraídos usando um conjunto de heurísticas cuidadosamente projetadas [11]. Dessa forma, o modelo multilíngue pré-treinado de última geração da Google, juntamente com o “XLSum” ajustado pelos autores possibilitou realizar a sumarização de qualidade em 44 idiomas diferentes. Por tal motivo, foi decidido utilizar esse modelo como principal nesse projeto de pesquisa.

### 3.4 Base de dados

Visando aplicar um ajuste fino no algoritmo abstrativo devemos ter, como visto anteriormente, uma base de dados de pelo menos 1000 textos e seus respectivos resumos. Porém, 1000 dados ainda é um número difícil de se obter. Portanto, inspirado nas ideias vistas no artigo do Google Pegasus, foi desenvolvido um algoritmo em Python chamado “create\_data\_base.py” no qual utiliza uma parte do algoritmo “summary\_tf-idf.py” e de maneira semelhante iremos realizar a sumarização do artigo previamente extraído do portal Saense, com o Script “handle\_site.py”, salvar esse resumo e logo após remover a sentença escolhida do texto original. Assim, ao final da execução do algoritmo teremos uma base de dados pronta para ser utilizada para ajuste fino de algoritmos abstrativos.

### 3.5 Métrica ROUGE

A métrica escolhida para avaliar a qualidade dos resultados obtidos neste projeto de pesquisa foi a métrica “ROUGE”. Tal técnica compara um resumo feito por um Humano com um texto gerado pelo algoritmo, e a partir de métodos estatísticos gera medidas que ajudam a discernir quais resumos geraram resultados com maior acurácia. O Rouge possui tais medidas: Rouge-1 recall, Rouge-1 precision, Rouge-1 F1-score, Rouge-2 recall, Rouge-2 precision, Rouge-L recall e Rouge-1 precision [12]. Tais medidas serão utilizadas para avaliar os resultados desse projeto de pesquisa e são obtidas da seguinte maneira:

Rouge-1: Compara cada palavra do texto de referência, resumo gerado por humano, com o resumo desejado. Possui três métricas:

$$TF = \frac{\text{Número de vezes que o termo "t" aparece no texto}}{\text{Número de termos totais no texto}}$$

---

<sup>10</sup>[https://huggingface.co/csebuethlp/mT5\\_multilingual\\_XLSum](https://huggingface.co/csebuethlp/mT5_multilingual_XLSum)

$$\text{ROUGE-1 recall} = \frac{\text{Número de correspondência de palavras}}{\text{Número de palavras na referência}}$$

$$\text{ROUGE-1 precision} = \frac{\text{Número de correspondência de palavras}}{\text{Número de palavras no resumo}}$$

$$\text{ROUGE-1 F1-score} = 2 \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Rouge-2: Compara pares de palavras do texto de referência, resumo gerado por humano, com o resumo desejado. Possui duas métricas:

$$\text{ROUGE-2 recall} = \frac{\text{Número de correspondência de pares}}{\text{Número pares na referência}}$$

$$\text{ROUGE-2 precision} = \frac{\text{Número de correspondência de pares}}{\text{Número de pares no resumo}}$$

Rouge-L: Compara a mais longa sequência comum (LCS) de palavras entre os textos. A principal vantagem que o Rouge-L possui em relação à Rouge-1 e Rouge-2 é que ele não depende de correspondências consecutivas de n-palavras (apenas uma palavra para o Rouge-1 e pares para Rouge-2) e, portanto, tende a capturar a estrutura da frase com muito mais precisão. Possui duas métricas:

$$\text{ROUGE-L recall} = \frac{\text{LCS(resumo, referência)}}{\text{Número de palavras na referência}}$$

$$\text{ROUGE-L precision} = \frac{\text{LCS(resumo, referência)}}{\text{Número de palavras no resumo}}$$

#### 4 Resultados e discussão

Para esse projeto de pesquisa, foram desenvolvidos 5 algoritmos, um "Crawler", dois algoritmos que realizam a sumarização dos textos, por meio de técnicas diferentes, e um último algoritmo que irá auxiliar na criação de uma base de dados para projetos futuros. O algoritmo de Crawler é composto por dois scripts em Python, "handle\_page.py e handle\_site.py", esses atuam em conjunto para extrair com precisão os textos presentes no site do Saense, salvando os mesmos em arquivos ".txt", além disso, salva informações extras em ".json" e imagens presentes no artigo que podem ser úteis em próximas etapas do projeto. Já o Algoritmo de sumarização extrativa, "summary\_tf\_idf.py", realiza o



resumo dos textos extraídos pelo Crawler utilizando uma técnica de sumarização chamada TF-IDF. Enquanto o segundo algoritmo de sumarização, “abstractive\_summary.ipynb”, utiliza um modelo abstrativo para gerar os resumos. Por fim, o último algoritmo desenvolvido até o momento, “create\_data\_base.py”, foi baseado numa estratégia adotada pela Google em seu modelo Pegasus que consiste em gerar, de maneira simples, uma grande quantidade de dados para treinar modelos abstrativos.

Seguindo essa linha de raciocínio, para validar os resultados obtidos nesse projeto de pesquisa foi utilizado a métrica ROUGE, na qual aplicamos e comparamos os textos gerados pelos algoritmos desenvolvidos com um texto gerado por um humano. Veja abaixo:

Cientistas brasileiros estão desenvolvendo uma tecnologia capaz de detectar a quantidade de toxinas que circulam no organismo de pacientes com doença renal chronica.

Resumo gerado por algoritmo abstrativo.

Isso pode significar um contingente grande de brasileiros que morrem sem acesso a tratamento. Agora, a intencao e aprimorar essa tecnologia para desenvolver um teste rapido semiquantitativo que possa ser escalonado. O metodo e destinado a detectar a presenca de uma substancia alvo em uma amostra liquida sem a necessidade de equipamentos especializados e dispendiosos. Atualmente, os pesquisadores estao funcionalizando o biossensor para torna-lo o mais sensivel e estavel possivel. Ela comenta que e necessario mostrar que essa engenharia e estavel em diferentes temperaturas e ao longo do tempo e que essa etapa dura bastante tempo. Entretanto, a exploracao comercial dessa patente e outro trabalho, que ainda deve demorar a acontecer. Existe uma emergencia por formacao de pessoas qualificadas e essa e uma das nossas prioridades, assegura Larissa. Nao e possivel pagarmos tao caro por testes cujos valores de custo sao infinitamente menores. Formar profissionais que possam produzir diferentes biossensores e testes de diagnostico e uma necessidade imediata. Temos condicoes de fazer no Brasil. Obviamente requer investimento, mas requer tambem mao de obra especializada capaz de fazer e de propor ideias. Com dificuldade estamos caminhando para isso, mas sabemos que e um caminho longo a ser trilhado, finaliza.

Resumo gerado por algoritmo estatístico.

Pesquisadores do Departamento de Patologia Basica da Universidade Federal do Parana (UFPR) estao desenvolvendo um prototipo de teste rapido que identifica a quantidade de toxinas uremicas circulando no organismo de pacientes com doenca renal cronica. O acumulo dessas toxinas uremicas na circulacao pode levar a quadros inflamatorios e, por isso, quanto mais rapido se tem conhecimento sobre os niveis circulantes desses componentes, mais breve e possivel iniciar o tratamento e tentar impedir o avanco de estagio da doenca. Dessa forma, com esse protoripo, e possivel verificar de maneira precoce o estado da doenca contruibuindo para o tratamento. Tal prototipo assemelha-se a um teste de gravidez, possibilitando o monitoramento da doenca de maneira mais facil e pratica sem a necessidade de equipamentos especializados e dispendiosos. Atualmente um grande numero de brasileiros morrem sem acesso a tratamento. Portanto, o objetivo agora e aprimorar essa tecnologia para desenvolver um teste rapido semiquantitativo que possa ser escalonado e patenteado segundo a professora de Imunologia, Larissa Magalhaes Alvarenga. Além disso, a professora comenta que existe uma emergencia por formacao de pessoas qualificadas e falta de investimento adequado na area da ciencia. Correlacionando com a Pandemia, Larissa afirma que o preco pago pelo governo por testes rapidos foi muito alto, pois foi pago um preco muito alto por produtos de custo infinitamente menores. Dessa forma, formar profissionais que possam produzir diferentes biossensores e testes de diagnostico e uma necessidade imediata. Larissa ainda afima: "Com dificuldade estamos caminhando para isso, mas sabemos que e um caminho longo a ser trilhado, finaliza."O projeto Producao desses testes rapidos para o monitoramento da Doenca Renal Cronica foi aprovado no Programa Pesquisa para o SUS Gestao Compartilhada em Saude (PPSUS) Edicao 2020/2021, da Fundacao Araucaria, e recebeu um incentivo de 36 mil reais. Alem da coordenadora Larissa, participam da pesquisa a professora, tambem do Departamento de Patologia Basica da UFPR, Juliana Moura e a mestranda Jessica Camargo, do Programa de Pos-Graduacao em Microbiologia, Parasitologia e Patologia.

Resumo gerado por Humano.

Os gráficos gerados pela avaliação ROUGE podem ser vistos abaixo:

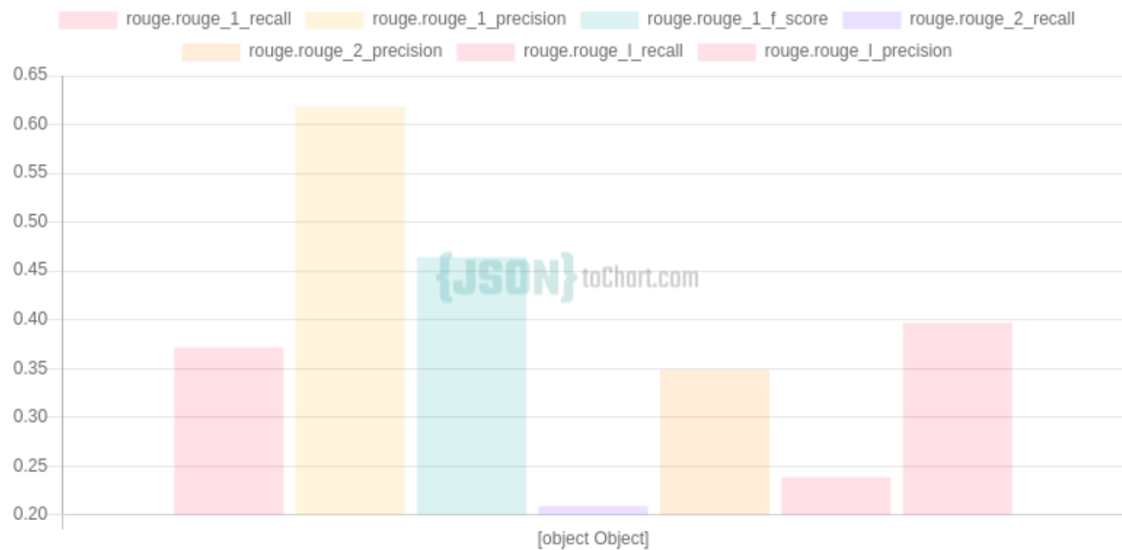


Figura 4: Resultado da Métrica ROUGE aplicado ao algoritmo extrativo com referência no resumo Humano.

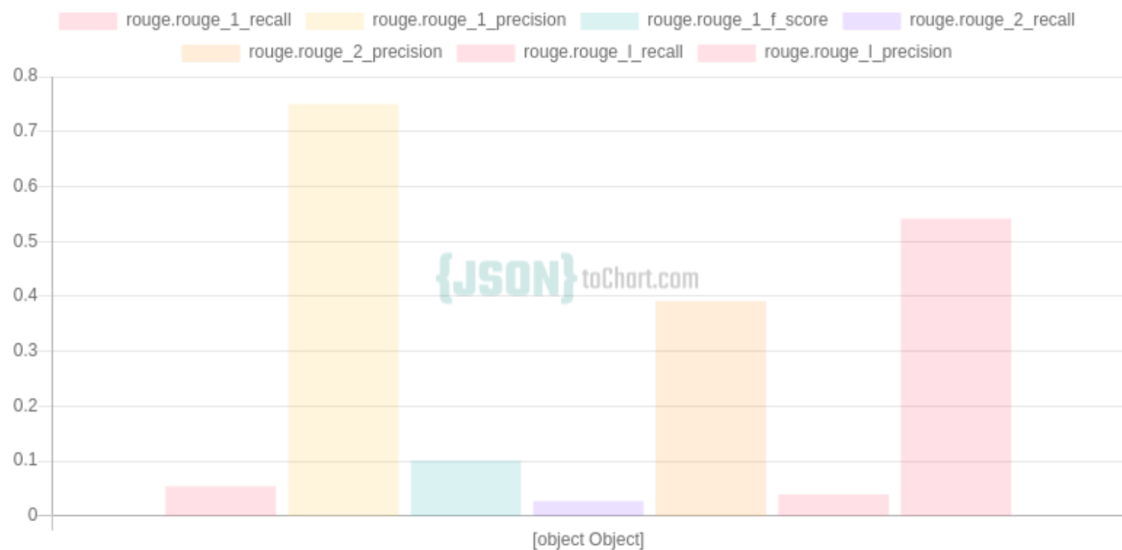


Figura 5: Resultado da Métrica ROUGE aplicado ao algoritmo abstrativo com referência no resumo Humano.

É perceptível que a abordagem estatística do “TF-IDF” gerou melhores resultados. De fato, essa técnica produziu resumos de boa qualidade, apesar de apresentar baixa coesão entre as sentenças. Tal problema é uma limitação do próprio método, pois as partes mais importantes do texto são extraídas e logo após é feita a união entre elas sem levar em consideração a coesão entre as mesmas. O algoritmo abstrativo, por sua vez, não teve bons resultados, como pode ser observado nos gráficos, pois está gerando resumos muito curtos, logo, na maioria dos casos, muitas informações importantes são perdidas nesse processo. Apesar de produzir boas sentenças abstrativas, falta realizar ajustes que

possibilitem gerar resumos mais longos. Sendo assim, os resumos provenientes desse método iriam apresentar uma melhor qualidade em comparação com o “TF-IDF”, pois o mesmo cria novas sentenças ao invés de apenas copiar as mais importantes, além de apresentar maior coesão entre as mesmas.

## **5 Perspectivas de Futuros Trabalhos**

Em síntese, futuramente, pode-se aprimorar o atual algoritmo abstrativo, realizando o ajuste fino e aprimorando-o para que seja possível gerar resumos maiores sem erros residuais significativos. Além disso, pode-se voltar a explorar o algoritmo do Google Pegasus, pois provou-se gerar resumos também de boa qualidade. Outra ideia seria implementar um algoritmo que identifique as palavras e sentenças mais importantes de um texto, possibilitando que seja encontrado relações entre outros artigos presentes no portal Saense, para que assim seja possível agrupá-los e identificar relações que até então não seriam facilmente notadas. Essa ferramenta pode ser utilizada, juntamente com os algoritmos de sumarização, para auxiliar na criação de um projeto maior, no qual tem como intenção a criação de Histórias em Quadrinhos tendo como base os artigos da plataforma Saense para que seja feita a distribuição gratuita em escolas públicas, tornando conteúdos científicos acessíveis e mais atraentes para jovens e crianças. Sendo assim, as ferramentas aqui projetadas e implementadas tornam-se valiosas para o artista que irá desenvolver os Quadrinhos, pois o mesmo terá uma lista de palavras chaves, postagens relacionadas e os resumos das mesmas, portanto, facilitando o seu desenvolvimento criativo.

## **6 Conclusões**

Em suma, conclui-se que esse projeto de pesquisa foi de grande importância e contribuição para formação científica e profissional dos envolvidos. Nela, foi explorado diversas técnicas e conceitos que fazem parte do imenso campo de Processamento de Linguagem Natural. Aliado a isso, foi desenvolvido algoritmos, repositório e website de boa qualidade que podem auxiliar diretamente com o projeto do portal Saense e também na construção de projetos futuros, como as Histórias em Quadrinhos citadas anteriormente. Portanto, o objetivo desse projeto de pesquisa foi cumprido e há “portas” abertas para futuras melhorias e aperfeiçoamento. Todo o material desenvolvido está disponível de forma gratuita e de fácil acesso para que possa contribuir expandindo e aprimorando o conhecimento de outras pessoas que se interessam pelo assunto.

## **7 Referências bibliográficas**

[1] NADKARNI, P. M.; OHNO-MACHADO, L.; CHAPMAN, W. W. Natural language processing: an introduction. Journal of the American Medical Informatics Association, v. 18, n. 5, p. 544–551, set. 2011.

[6] CAVALCANTI, A. P. Uma medida de similaridade textual para identificação de plágio em fóruns educacionais. Ufrpe.br, 2016.

- [7] ZHANG, J. et al. PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. arXiv:1912.08777 [cs], 10 jul. 2020.
- [11] HASAN, T. et al. XL-Sum: Large-Scale Multilingual Abstractive Summarization for 44 Languages. Disponível em: <<https://aclanthology.org/2021.findings-acl.413>>.
- [12] FABBRI, A. R. et al. SummEval: Re-evaluating Summarization Evaluation. Transactions of the Association for Computational Linguistics, v. 9, p. 391–409, 2021.