Bot vs Wiki

Marcos Cordeiro de Brito Jr
Programa de Pós Graduação em Engenharia Elétrica e Computação (PPGEEC)
Universidade Presbiteriana Mackenzie
São Paulo, Brasil

***Abstract*—Este documento tem como objetivo descrever o processo de tratamento de texto para analisar artigos alterados por *bots* na Wikipedia em português. O estudo mostrará quais termos são mais alterados por robôs automatizados que atuam em modificações de artigos na Wikipedia, utilizando técnicas de processamento de textos para apresentar os resultados obtidos com algorítimos de agrupamento e frequência de palavras mais utilizadas.**

***Palavras-chave*—Wikipedia, bot, robôs,**

***Abstract*—This document aims to describe the word processing process for analyzing articles altered by bots on Wikipedia in Portuguese. The study will show which terms are most altered by automated robots that work on modifying articles on Wikipedia, using text processing techniques to present the results obtained with the most frequently used grouping and word frequency algorithms.**

***Index Terms*—Wikipedia, bot, robots,**

I. Introdução

Desde sua criação, em 15 de janeiro de 2001, a Wikipedia tornou-se o maior site de referência do mundo, atraindo 1,5 bilhão de visitantes únicos mensalmente em março de 2020. Atualmente, possui mais de 53 milhões de artigos em mais de 300 idiomas [1].

De acordo com os dados do próprio site, em Maio de 2020 existiam: 6091066 artigos de conteúdo, 50486092 páginas, 884861 *uploads* de arquivos, 39179043 usuários registrados e 955605050 arquivos editados.

Em um estudo realizado em 2014 por *Thomas Steiner* do Google na Alemanha [2], ele estudou a comparação de quantidade de artigos alteradas por usuários identificados na Wikipedia, usuário anônimos e robôs automatizados conhecidos como *bots*[1]. Neste estudo, ele constatou que cerca de 15% são alterados de forma automática ou programática por esses *bots*. A comparação feita por ele foi de quantidade de artigos, sem olhar para o que estava sendo alterado.

Tendo como influência a publicação do *Thomas Steiner*, esse estudo irá explorar quais artigos são mais alterados por *bots*, coletando os dados da própria Wikipedia e fazendo análises de tratamento de texto para mostrar através de algorítimos de agrupamento e frequências de palavras, os temas com mais alterados.

O estudo irá mostrar análises através de algorítimos de agrupamentos como *K-Means* e *DBScan* e análise de frequência para contabilizar palavras utilizadas nos títulos dos artigos. Será mostrado também como foi feita a preparação dos títulos coletados antes de aplicar as análises mencionadas. Serão mostrados os gráficos e indicadores gerados durante o desenvolvido da estudo e, ao final, a conclusão obtida através dos resultados obtidos.

[1]**Bot**, diminutivo de *robot*, também conhecido como *Internet bot* ou *web robot*, é uma aplicação de software concebido para simular ações humanas repetidas vezes de maneira padrão, da mesma forma como faria um robô.

II. Referência Teórica

Como inspiração para esse estudo, foi seguido a referência utilizada no artigo *Bots vs. Wikipedians, Anons vs. Logged-Ins (Redux) A Global Study of Edit Activity on Wikipedia and Wikidata* [2] onde são comparados artigos alterados por pessoas e *bots*, tendo como foco principal a quantidade de artigos alterados.

As etapas de prepação de texto foi inspirada nos artigos *A Tecnologia de Mineração de Textos* [3], *What Is Text Mining?* [4] e *Getting Started in Text Mining* [5]. Esses estudos serviram como base para analisar o que utilizar para tratar textos e preparar as informações da melhor forma a ser utilizada nas análise escolhidas.

Para entendimento do conceito e técnicas de *Bag of Words*, o artigo *Contextual Bag-of-Words for Visual Categorization* [6] serviu como base de referência e entendimento sobre o assunto.

A aplicação das técnicas de agrupamentos utilizadas e frequência das palavras, além de complemento de entendimento para preparação da base de dados, foi utilizado como referência teórica o livro *Introdução à mineração de Dados* [7].

Para entendimento de medição de métricas do *K-Means*, foi estudado o *paper K-means with Three different Distance Metrics* [8]. Essa publicação serviu para entender melhor como o *K-Means* faz os cálculos de métricas de distância entre e intra grupos.

III. Metodologia

O objetivo principal ao final desse artigo será analisar quais artigos são alterados por *bots*, quais os termos presentes nos títulos desses artigos, quantidade de alterações por termos encontrados e possíveis relações desses termos através de agrupamentos. Além dessas análises, serão feitas comparações com artigos que não foram alterados por *bots*, principalmente para mostrar as diferenças quantitativas entre as duas modalidades.

A. Estratificação dos Dados

Para fazer essa análise, precisamos ter os dados dessas alterações. Essas informações são disponibilizadas pela própria

Wikipedia [9] através de *endpoints Rest* no formato *JSon*, onde qualquer pessoa pode fazer consultas online ou coletar seu conteúdo através de integrações com protocolo HTTP.

Os dados que foram utilizados nesse estudo foram coletados entre os dias 03 de Março de 2020 e 15 de Maio de 2020. Não houve nenhum filtro com os dados antes da utilização das informações e todos os tratamentos ou exclusões serão apresentados na sessão de tratamento dos dados. A quantidade de registros totalizaram 166115 objetos coletados.

O retorno do endpoint traz aproximadamente 31 atributos, onde foram utilizado somente os mais relevantes para o estudo proposto. São eles: *id* enviado pela Wikipedia, usuário da alteração, *timestamp* com a data da alteração, campo *bot* onde diz se foi ou não alterado por *bots* e título do artigo. Os demais campos não foram utilizados por não serem relevantes.

B. Pré-processamento

Após a coleta e armazenamento dos dados recebidos pela Wikipedia, o título será tratado para criação do *Bag of Words* junto com todos os tratamentos necessários. A descrição dessas etapas serão descritas na ordem como foram executadas na implementação deste estudo.

- **Remoção de objetos e atributos com valor ausente:** foram removidos todos os objetos que estavam com todos atributos sem valor ou onde não haviam títulos. Por se tratar do principal campo para o objetivo do estudo, não havendo o título não faz sentido manter esse objeto.
- **Transformação dos títulos para minusculo:** para conseguir comparar palavras iguais, todos os títulos foram transformados para terem os caracteres em caixa baixa.
- **Usuários ausentes ou não identificados:** quando o usuário que faz a alteração do arito não é identificado pela plataforma, esse valor é preenchido com uma sequência numérica gerada pelo próprio site, ou endereço de IP do autor da alteração. Todos esses casos, mais os usuários com dados nulos, foram substituídos pelo valor global *unknown* (desconhecido em inglês).
- **Data de alteração com valor ausente:** quando o campo da data da alteração apresentava valor ausente, esse valor foi substituído pela data 01/01/2020. Essa data foi escolhida por estar fora da data de coleta das informações e ficaria fácil identifica-las caso fosse necessário.
- **Campo *bot* com valor ausente:** juntamente com o título, o campo *bot* é muito importante para a análise proposta pelo estudo, por isso é importante ter um valor preenchido para análise. No caso desse atributo em específico, foi usado o critério de preencher os valores ausentes com o valor de maior frequência da base, que no caso foi o valor **False**.
- **Criar *tokens* dos títulos:** a técnica de criação de **tokens** para análise de textos consiste em separar as sentenças em palavras e armazenar em forma de vetores e/ou dicionários de dados [10]. Essa é uma etapa muito importante e crucial para a conversão dos textos em dados números, ajudando nas análises que serão realizadas. Essa etapa foi feita, aplicando em todos os títulos da base de dados.
- **Remoção de *stopwords*:** foram removidos palavras como: ”de”, ”o”, ”que”, ”para”, ”com”, que fazem parte de uma sentença ou frase normal da língua portuguesa mas que atrapalham em análises textuais. Essas palavras são conhecidas como ”stopwords” e existem bibliotecas e ferramentas que ajudam a remoção dessas palavras. No caso desse projeto foi utilizado a biblioteca *nltk* [11] para linguaem Python.
- *Stemming* O *stemming* é um processo de remoção e substituição de sufixos de palavras para chegar a uma forma raiz comum da palavra. Esse processo deixa as palavras de uma forma genérica, para não ter entendimentos ou interpretações diferentes da mesma palavra. Exemplo: ”Eu gosto de correr” após a utilização dessa ferramenta, ficaria ”Eu gost de corr”.

Esses foram os métodos utilizados para fazer a limpeza, normalização e transformação dos títulos antes de iniciar as análises dos dados.

IV. Análises

As primeiras análises dos dados foram quantitativas para entender o tamanho da base que seria trabalhado e porcentagem de artigos alterados por *bots*.

O total de registro após o tratamento de dados foram de 131072, sendo 122290 que não foram alterados por *bots* e 8782 que foram alterados. Esses números estão representados de forma gráfica em porcentuais no gráfico abaixo.

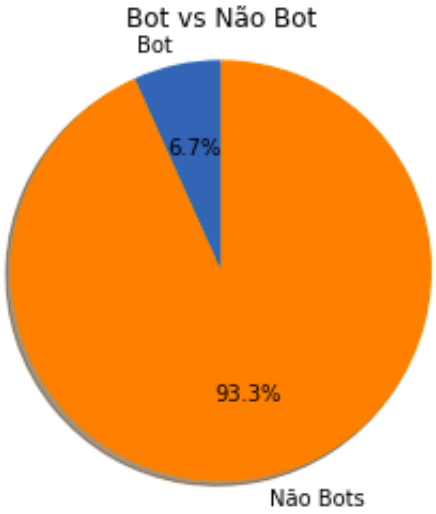

Fig. 1. Alterações por Bots e Usuários

Fazendo uma análise das palavras mais frequêntes, as 10 primeiras que mais aparecerem estão listadas na tabela a seguir.

Como forma de tentar visualizar os agrupamentos das 100 palavras mais utilizadas nos títulos alterados, foi feito uma análise hierárquica para tentar visualizar os *cluster* em forma

Palavra	Frequência
pandemia	763
covid-19	693
brasil	616
futebol	609
eliminacao	582
carecem	532
dados	498
2020	479
rapida	437
campeonato	362

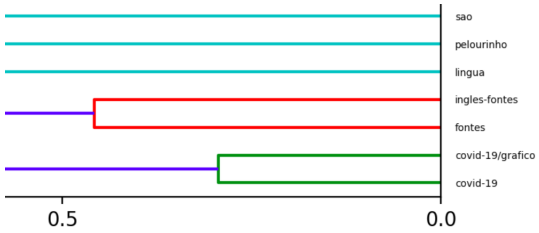

Fig. 3. Dendrograma

de dendrograma. O gráfico abaixo mostra essa visualização onde fica claro que não tem muitos grupos, sendo a maior parte das palavras em um grande grupo com palavras sem relações aparentes.

Em outra visão das palavras mais comuns, foi feita uma nuvem de palavras para ter uma visão lúdica dessas frequências. Essa visão já mostra que existem muitos termos com tamanhos parecidos, o que mostra uma certa diversidade dos título que foram coletados e analisados.

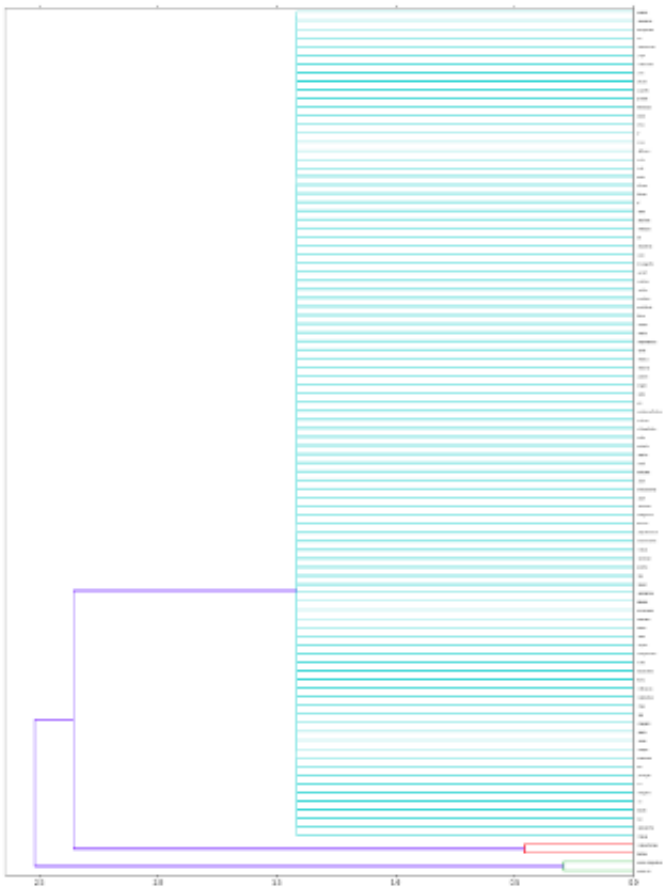

Fig. 2. Dendrograma

Fig. 4. Dendrograma

Comparando as palavras mais utilizadas, separando entre alteradas e não alteradas por *bots*, temos os seguintes resultados.

Bots = True		Bots = False	
Palavra	**Frequência**	**Palavra**	**Frequência**
covid-19	370	brasil	599
pandemia	361	eliminacao	582
dados	359	futebol	578
casos	60	carecem	532
numero	58	2020	460
pagina	48	rapida	437
imagem	42	pandemia	402
testes/1	37	campeonato	338
imagens	37	clube	335
categorias	35	covid-19	323

Olhando mais de perto a parte do dendrograma onde teve uma divisão dos *clusters*, podemos notar as poucas ligações que foram encontradas entre as 100 palavras mais utilizadas.

Olhando essa frequência de forma gráfica, podemos visualizar as duas imagens abaixo para cada uma das situações de alterações de artigos.

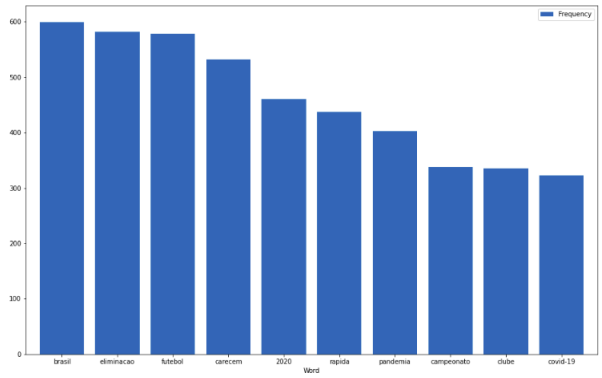

Fig. 5. Bots = False

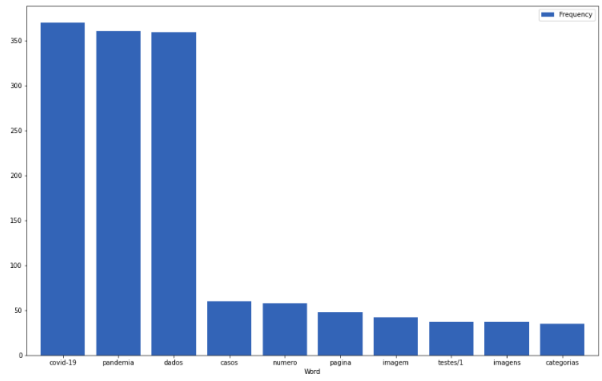

Fig. 6. Bots = True

A. *Análises de Grupos*

Para fazer as análises de grupo, foram executadas duas técnicas para tentar mensurar a quantidade ideal de *clusters*. As técnicas utilizadas foram de Silhueta [12] e Cotovelo [13].

No resultado do método de silhueta, os resultados ideias seriam o mais próximo de 1, mas os eles se mostraram um pouco distantes conforme podemos analisar abaixo.

```
Para n_clusters = 2, silhouette score é 0.17815216287650423)
Para n_clusters = 3, silhouette score é 0.2003784573943305)
Para n_clusters = 4, silhouette score é 0.2098773664412403)
Para n_clusters = 5, silhouette score é 0.20806431972522532)
Para n_clusters = 6, silhouette score é 0.22140765287732425)
Para n_clusters = 7, silhouette score é 0.22135244387045355)
Para n_clusters = 8, silhouette score é 0.24184210035023654)
Para n_clusters = 9, silhouette score é 0.24393808316468427)
Para n_clusters = 10, silhouette score é 0.25975070037460124)
Para n_clusters = 11, silhouette score é 0.2599909373795272)
Para n_clusters = 12, silhouette score é 0.26377373761013784)
Para n_clusters = 13, silhouette score é 0.26494158841940435)
Para n_clusters = 14, silhouette score é 0.2720718795132279)
Para n_clusters = 15, silhouette score é 0.27280564251635925)
Para n_clusters = 16, silhouette score é 0.28239316702511597)
Para n_clusters = 17, silhouette score é 0.2725392634342573)
Para n_clusters = 18, silhouette score é 0.28322414410182867)
Para n_clusters = 19, silhouette score é 0.28238966539535715)
Para n_clusters = 20, silhouette score é 0.2751184133528897)
Para n_clusters = 21, silhouette score é 0.2925817151055534)
Para n_clusters = 22, silhouette score é 0.2902030110582138)
Para n_clusters = 23, silhouette score é 0.2897473925518492)
Para n_clusters = 24, silhouette score é 0.29008962394587334)
Para n_clusters = 25, silhouette score é 0.29493045938391277)
Para n_clusters = 26, silhouette score é 0.2971833469369819)
Para n_clusters = 27, silhouette score é 0.3011844426746769)
Para n_clusters = 28, silhouette score é 0.3047207025985396)
Para n_clusters = 29, silhouette score é 0.301178728214537)
```

Fig. 7. Silhueta Bots = True

```
Para n_clusters = 2, silhouette score é 0.004572732655051248)
Para n_clusters = 3, silhouette score é 0.017574839370025554)
Para n_clusters = 4, silhouette score é 0.018932060220554804)
Para n_clusters = 5, silhouette score é 0.019904880305883194)
Para n_clusters = 6, silhouette score é 0.021291579046003577)
Para n_clusters = 7, silhouette score é 0.02526549559989224)
Para n_clusters = 8, silhouette score é 0.027217798945164223)
Para n_clusters = 9, silhouette score é 0.03315353032403929)
Para n_clusters = 10, silhouette score é 0.032127641041954015)
Para n_clusters = 11, silhouette score é 0.03261514537085327)
Para n_clusters = 12, silhouette score é 0.03353695577385147)
Para n_clusters = 13, silhouette score é 0.033732950532721066)
Para n_clusters = 14, silhouette score é 0.03436986580630375)
Para n_clusters = 15, silhouette score é 0.03603904732260383)
Para n_clusters = 16, silhouette score é 0.04538505040075775)
Para n_clusters = 17, silhouette score é 0.04084684037381227)
Para n_clusters = 18, silhouette score é 0.04238853352787495)
Para n_clusters = 19, silhouette score é 0.04037789167172411)
Para n_clusters = 20, silhouette score é 0.04569796558523745)
Para n_clusters = 21, silhouette score é 0.04821882688740638)
Para n_clusters = 22, silhouette score é 0.047411329754048805)
Para n_clusters = 23, silhouette score é 0.04924558747514093)
Para n_clusters = 24, silhouette score é 0.05338386999809498)
Para n_clusters = 25, silhouette score é 0.04685292485850723)
Para n_clusters = 26, silhouette score é 0.05267282615456227)
Para n_clusters = 27, silhouette score é 0.049748798546870435)
Para n_clusters = 28, silhouette score é 0.05472465182874271)
Para n_clusters = 29, silhouette score é 0.05147622839209209)
```

Fig. 8. Silhueta Bots = False

Para o método do cotovelo, o resultado também não teve a saída como se esperava, como uma curva decrescente e bem definida, ficando mais constante ao final.

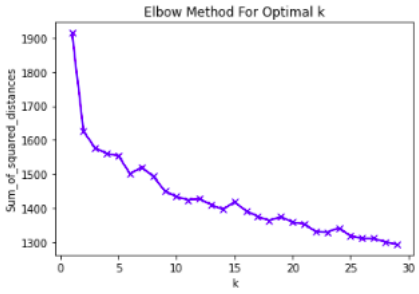

Fig. 9. Cotovelo Bots = True

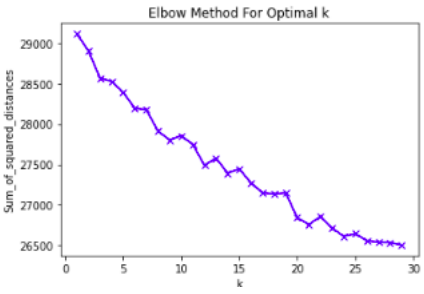

Fig. 10. Cotovelo Bots = False

Esse resultado já mostrou que os dados estão muito dispersos, o que aponta um indicio ruim para criação de grupos. De qualquer foram foram executado duas técnicas para analisarmos os resultados.

B. *K-Means e DBScan*

Abaixo estão os resultados dos *clusters* fazendo a análise de *K-Means*.

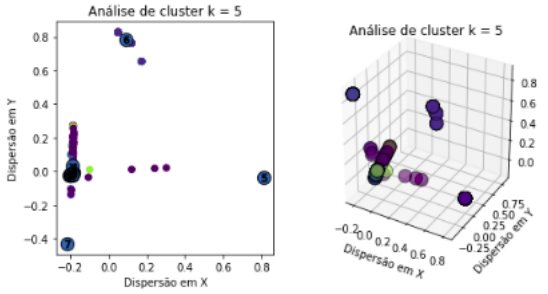

Fig. 11. K-Means Bots = True

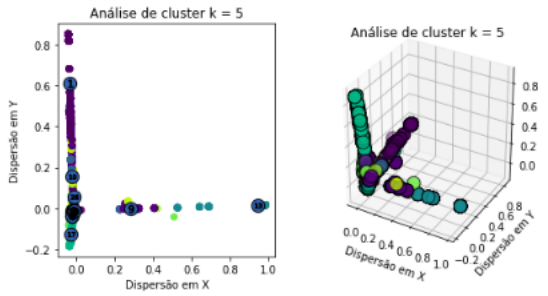

Fig. 12. K-Means Bots = False

Como esperado após as análises do cotovelo e de silhueta, a dispersão dos gráficos do K-Means executando com 5 clusters mostrou grupos muito dispersos. Esse resultado já era esperado pela quantidade de palavras com diferentes contextos.

Esse resultado também foi observado ao executar a técnica DB-SCan, onde se espera pontos mais próximos da reta central.

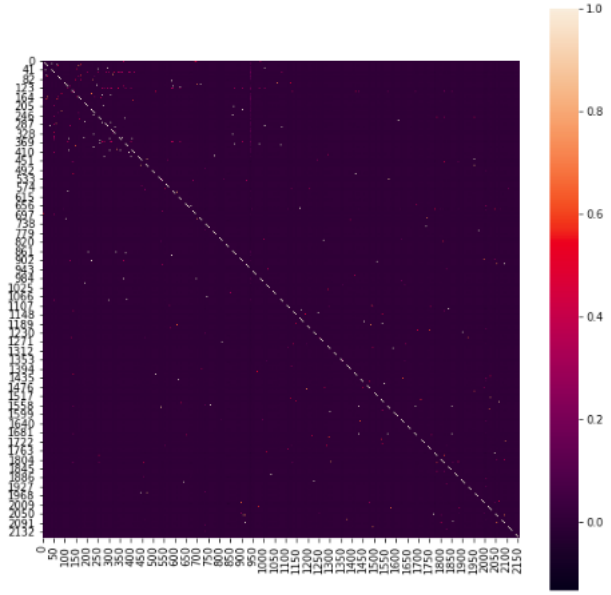

Fig. 13. DBScan

V. Conclusão

Como conclusão ficou claro que os métodos de agrupamento que foram testado para esse estudo não se mostraram eficientes para analisar a frequência de artigos alterados através de grupos.

Os gráficos e análises quantitativas se mostraram mais eficientes para visualizar quais os termos mais frequentes em alterações feitas por *bots*, onde *COVID-19* e *Pandemia* foram os dois mais contabilizados. Já em dados que foram alterados por pessoas estão *brasil, eliminação e futebol.*.

O que dá para concluir com essas informações são que robôs estão alterando dados sobre temas mais falados no momento no Brasil, como a pandemia do **COVID-19** e as pessoas estão mais preocupadas com **futebol**.

VI. Perspectivas futuras

Para melhorar o resultado desse estudo, ele pode ser repetidos com grupos mais específicos de assuntos, tendo uma relação maior entre os temas alterados. Isso pode melhorar o resultado dos *clusters*.

Outra forma de evoluir o estudo seria analisar outros dados disponibilizados pela Wikipedia, tentando extrair informações a partir do contexto do conteúdo dos artigos e não somente dos títulos.

References

[1] Wikipedia:about. [Online]. Available: https://en.wikipedia.org/wiki/Wikipedia:About

[2] T. Steiner, "Bots vs. wikipedians, anons vs. logged-ins (redux) a global study of edit activity on wikipedia and wikidata," in *Proceedings of The International Symposium on Open Collaboration*, 2014, pp. 1–7.

[3] C. Aranha and E. Passos, "A tecnologia de mineração de textos," *JRESI-Revista Elerônica de Sistemas de Informação - Lab.ICA Elétrica PUC-Rio*, no. 2, 2006.

[4] M. Hearst, "What is text mining?" *SIMS,UC Berkeley*, October 2003.

[5] K. B. Cohen and L. Hunter, "Getting started in text mining," *PLOS Computational Biology*, vol. 4, no. 20, January 2008.

[6] T. Li, T. Mei, I. Kweon, and X. Hua, "Contextual bag-of-words for visual categorization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 4, pp. 381–392, 2011.

[7] D. G. FERRARI and L. N. D. C. SILVA, *Introdução a mineração de dados*. Editora Saraiva, 2017.

[8] A. Singh, A. Yadav, and A. Rana, "K-means with three different distance metrics," *International Journal of Computer Applications*, vol. 67, no. 10, 2013.

[9] Wikimedia eventstreams. [Online]. Available: https://stream.wikimedia.org/?doc/Streams

[10] S. Bird, "Nltk-lite: Efficient scripting for natural language processing," in *Proceedings of the 4th International Conference on Natural Language Processing (ICON)*, 2005, pp. 11–18.

[11] Nltk. [Online]. Available: https://www.nltk.org/book/ch02.html

[12] T. B. Ganesan and R. Sukanesh, "Segmentation of brain mr images using fuzzy clustering method with silhouette method," *Journal of Engineering and Applied Sciences*, vol. 3, pp. 792–795, 2008.

[13] P. Bholowalia and A. Kumar, "Ebk-means: A clustering technique based on elbow method and k-means in wsn," *International Journal of Computer Applications*, vol. 105, no. 9, 2014.