Automating the identification of hate speeches on the Internet using crowdsourcing and Natural Language Processing. Automatizando a identificação de discursos de ódio na Internet utilizando *crowdsourcing* e Processamento de Linguagem Natural.

Abstract: The emergence of the Internet has provided several benefits for us, such as an increase in the speed of information exchange, distance communication in real time and an increase in media that we could express our emotions and opinions. On the other hand, these facilities also allowed malicious people to publish hate speech and intolerance. Due to such problems, public media in the Internet are looking for automated means of blocking, omitting and deleting texts and comments that may fall into these categories. Therefore, in the present work, the objective was to use the scraping method of website comments, to label them with public opinions (crowdsourcing) and, finally, to apply Natural Language Processing and Machine Learning techniques to automatically identify hate speechs written in Brazilian Portuguese.

Resumo: O surgimento da Internet proporcionou diversos benefícios para a humanidade, tais como o aumento na velocidade da troca de informações, comunicação a distância em tempo real e o aumento de mídias que pudessem expressar suas emoções e opiniões. Por outro lado, essas facilidades também permitiram que pessoas mal-intencionadas publicassem discursos de ódio e intolerância. Devido a tais problemas, as mídias públicas buscam meios, automatizados, de bloquear, omitir e excluir textos e comentários que possam se encaixar nessas categorias. Portanto, no presente trabalho, objetivou-se utilizar o método de raspagem dos comentários de *websites (webscraping)* de mídias sociais, rotulá-los com opiniões púbicas (*crowdsourcing*) e, por fim, aplicar técnicas de Processamento de Linguagem Natural e Aprendizado de Máquina para identificar automaticamente os discursos ofensivos escritos em português do Brasil.

Palavras-chave: Processamento de Linguagem Natural, Discurso de Ódio, *Toxic Comments*, *Natural Language Processing*, *Crowdsourcing*, *Web Scraping*.

1. Introdução

O ser humano expressa seus pensamentos ou sentimentos através de uma linguagem, sendo que tudo o que se fala, escuta e escreve na vida diária também está na forma de linguagem natural, como, por exemplo, diálogos de filmes, conversas no WhatsApp, Facebook e Twitter e comentários de produtos em lojas virtuais. O Processamento de Linguagem Natural é um campo da Ciência da Computação, da Inteligência Artificial e da Linguística Computacional relacionado às interações entre computadores e linguagens humanas (naturais) e pode ser definido como o processamento automático (ou semiautomático) da linguagem natural humana, ou seja, consiste em dar a capacidade de tecnologias computacionais de processar linguagem natural humana, como já é possível ver nos produtos existentes das principais empresas de tecnologia do mundo, como o Assistente da Google, a Siri da Apple e assim por diante (THANAKI, 2017).

Na última década, as pessoas começaram a usar as mídias sociais para expressar e compartilhar seus sentimentos sobre produtos, pessoas e serviços. Os textos publicados em mídias sociais se tornaram importantes fontes de informação. Hoje, tem-se que a análise desses textos é uma poderosa forma de monitorar a opinião e a resposta dos clientes sobre organizações, por exemplo (YANG *et al*., 2017).

O acesso à Internet e sua utilização como um dos principais meios de comunicação, embora positiva, apresenta também efeitos colaterais negativos. Infelizmente, não é incomum encontrar comentários ofensivos nestas mesmas mídias sociais. De acordo com SILVA *et al*. (2011), o discurso de ódio caracteriza-se pela manifestação que tende a insultar, intimidar ou assediar pessoas, ou seja, pelo desprezo por pessoas que compartilham de alguma característica que as tornam componentes de um grupo, por exemplo, sua raça, cor, etnicidade, nacionalidade, sexo ou religião. Neste tipo de discurso, essas pessoas são referidas como inferiores ou tidas como indignas da mesma cidadania dos emissores dessa opinião.

O discurso de ódio é utilizado frequentemente no *cyberbullying*, sendo este definido como o uso da tecnologia da informação para prejudicar ou assediar outras pessoas de maneira deliberada, repetida e hostil, e consiste em um sério problema social, especialmente entre os adolescentes (HUANG *et al*., 2014).

Outro exemplo negativo onde os discursos de ódio estão inseridos é na proliferação de grupos de ódio (conhecidos como *haters*, em inglês), os quais exaltam homicídios, adotam ideologias racistas e xenofóbicas, e que defendem a violência. Neste contexto, as redes sociais são particularmente utilizadas por eles para propagar mensagens de ódio, recrutar novos membros e ameaçar usuários desses meios (ALMEIDA *et al*., 2017).

Algumas plataformas tentam diminuir esses impactos desabilitando comentários, com moderação colaborativa ou com exclusões manuais. Mas, essas abordagens têm se mostrado ineficiente e não escaláveis (HOSSEINI *et al*., 2017). Por isso, existe uma necessidade de pesquisar e desenvolver métodos para identificar automaticamente e em tempo real, comentários que possam ser ofensivos (WULCZYN *et al.*, 2017). Trabalhos relacionados como os de NANDHINI e SHEEBA (2015), DAVIDSON *et al*. (2017), ALMEIDA *et al*. (2017), BRETSCHNEIDER e PETERS (2017), WULCZYN *et al.* (2017) e SCHMIDT e WIEGAND (2017), apresentam resultados satisfatórios do uso do Processamento de Linguagem Natural e Aprendizado de Máquina na resolução do problema proposto, entretanto, todos os trabalhos atuam especificamente no idioma inglês.

No que se refere aos trabalhos que atuaram com *corpus* em português, do Brasil, tem-se BALOCCO e SHEPHERD (2017), os quais analisaram a violência verbal em *posts*. Seu *corpus* foi desenvolvido a partir de fragmentos textuais coletados do *site* de notícias *O Globo.com*, tendo sua anotação baseada no conceito de *flaming* e na categorização de BOUSFIELD (2008, *apud* BALOCCO e SHEPHERD, 2017), sendo sua análise realizada de forma manual e qualitativa. SOOD *et al.* (2012), utilizaram-se também de *crowdsourcing* para anotação do *corpus*, porém para detecção de blasfêmias.

Portanto, pelo exposto, justifica-se o desenvolvimento deste trabalho visto que a identificação automática de discurso de ódio escritos em português do Brasil disponíveis em mídias sociais permite diminuir impactos sociais gerados por grupos sociais de ódio, *cyberbullying* e outros males recorrentes de sua existência. Além disso, o desenvolvimento do trabalho permite a inovação tecnológica a partir da geração de sistemas computacionais, utilizando o estado da arte das técnicas de Processamento de Linguagem Natural e Aprendizado de Máquina, que sejam capazes de identificar os discursos de ódio no idioma raiz.

2. Metodologia

O projeto consiste em uma pesquisa ligada à prática de conhecimento científico para fins explícitos de análise de dados de mídias sociais, objetivando gerar conhecimentos para a aplicação prática, especificamente na solução de problemas relacionados à identificação

automática de discursos de ódio em mídias sociais. Para tanto, realizou-se a seguintes etapas: (*i*) extração de comentários do *site* g1.globo.com a partir de um *web scraping*; (*ii*) desenvolvimento de um *crowdsoucing* usando página *web* para rotulação do *corpus*; (*iii*) tratamento e pré-processamento do *corpus* usando técnicas do Processamento de Linguagem Natural; e (i*v*) criação e teste de modelos de Aprendizado de Máquina.

2.1. *Web Scraping*

Web S*craping* (raspagem de dados na *web*) consiste em uma prática de coleta de dados por qualquer outro meio que não seja um programa que interaja com uma API (Application Programming Interface). Isto é comumente realizado escrevendo um programa automatizado que consulta servidores *web* e solicita dados de forma a extrair a informação necessária (MITCHELL, 2015).

Para realizar esta atividade, utilizou-se as bibliotecas BeautilfulSoup4 e Selenium, sendo a primeira responsável por extrair os dados de um documento HTML (HyperText Markup Language) e a segunda por simular um *browser web*, como o *chromium*, a fim executar de funções JavaScript na página *web* de forma assíncrona para carregar as informações. Utilizando-se do Selenium, efetuou-se automaticamente o clique nos botões da página de notícias G1[1], da Globo, para carregamento dos dados e, por fim, utilizou-se as funções da biblioteca BeautifulSoup4 para extrair os comentários da página HTML carregada.

Para cada notícia carregada, todos seus comentários foram extraídos e armazenados no formato JSON (JavaScript Object Notation) e CSV (Comma-Separated Values) utilizando a biblioteca Pandas. Além dos comentários, foram armazenados também as *tags*, o título e o *link* da notícia, juntamente com a data e a hora de sua publicação.

2.2. Desenvolvimento de um sistema de *crowdsourcing* para anotações do *corpus*

Na segunda etapa do trabalho foi desenvolvido um sistema de *crowdsourcing*, ou colaboração coletiva, o qual é definido como uma forma de buscar avaliações neutras e da maneira mais imparcial possível sobre algum assunto. Em seu trabalho, BEHREN (2011), diz que uma enquete realizada em colaboração coletiva, pode ser mais ética e educacionalmente diversificada.

Uma vez obtidos os comentários na etapa anterior, desenvolveu-se uma aplicação *web*[2] utilizando um *microframework* em Python chamado Flask, apresentada na Figura 1. O sistema possui interface simple, na qual ao usuário acessar, ele se depara com uma caixa de texto com um comentário aleatório da base de dados, assim como o *link* da notícia para que ele saiba do anonimato e imparcialidade do *site* sobre o mesmo.

Ao acessar o sistema, este exibe um comentário aleatório da base de dados para o usuário, o qual pode realizar três possíveis ações, quais sejam, definir se o comentário é tóxico, definir se o comentário não é tóxico (limpo) e atualizar a página para gerar outro comentário caso ele não queira opinar sobre algum. Ao definir o comentário como tóxico, o usuário pode também opinar sobre o tipo de ofensa, sendo eles insulto, outros (genérico), religioso, xenofóbico, sexista e racial.

As anotações dos usuários foram então armazenadas na base de dados, sendo os rótulos dos comentários definidos como valores *booleanos*, ou seja, *True* no caso de comentários

[1] Disponível em: https://g1.globo.com
[2] Disponível em: http://cscomments.us-east-1.elasticbeanstalk.com.

tóxicos e *False*, no caso contrário. Os comentários foram então rotulados de acordo com uma média da quantidade de votos que recebeu como sendo tóxico ou não.

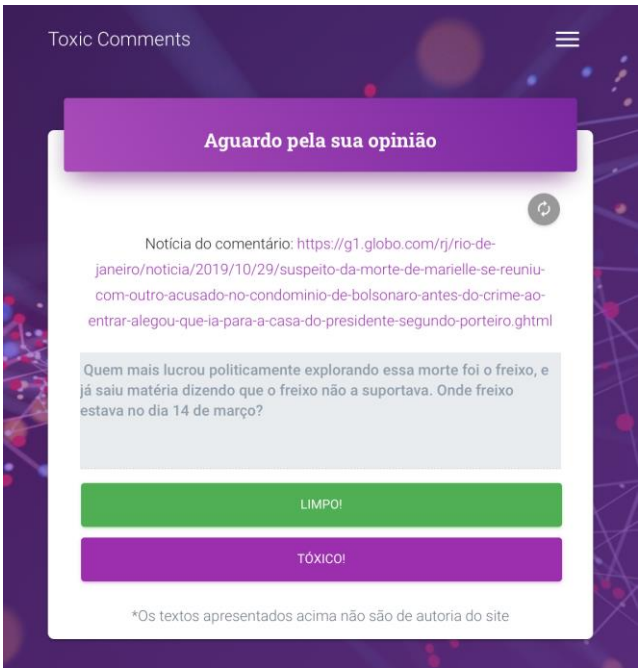

Figura 1. Aplicação *web* desenvolvido para *crowdsourcing*

2.3. Pré-processamento do *corpus*

O Processamento de Linguagem Natural (PLN) é uma subárea da Ciência da Computação, Inteligência Artificial e da Linguística que estuda os problemas da geração e compreensão automática de línguas humanas naturais, tendo sido iniciada na década de 1950, quando Alan Turing publicou o artigo "Computing Machinery and Intelligence", que propunha o que agora é chamado de Teste de Turing como critério de inteligência. Nos dias atuais tem-se uma enorme quantidade de textos disponíveis na Internet, os quais, para serem analisados por máquinas, precisam de uma representação estruturada e é nesse contexto que o Processamento de Linguagem Natural atua neste projeto.

Nesta etapa realizou-se o pré-processamento dos comentários anotados, conforme pode ser visto no DataFrame gerado pela biblioteca Pandas apresentado na Figura 2. Na primeira coluna é dada a classificação do comentário como sendo tóxico (1) ou não tóxico (0), gerada a partir do sistema de *crowdsourcing* descrito na etapa anterior.

Out[8]:

	toxico	comentario	minusculo	whitespacesnone	unidecode	comentario_final_su	comentario_final_wu
0	0	petróleo da Venezuela??? hahahahahahaha, dps ...	petróleo da venezuela??? hahahahahahaha, dps ...	petróleo da venezuela??? hahahahahahaha, dps d...	petroleo da venezuela??? hahahahahahaha, dps d...	petróleo venezuela hahahahahahaha dps dias faz...	petroleo venezuela hahahahahahaha dps dias faz...
1	0	O regime mais genocida é a religião.	o regime mais genocida é a religião.	o regime mais genocida é a religião.	o regime mais genocida e a religiao.	o regime genocida a religião	o regime genocida e a religiao
2	0	UÉ?	ué?	ué?	ue?	ué	ue
3	0	Se alguém é acuso de "UM" crime, por "UMA" pe...	se alguém é acuso de "um" crime, por "uma" pe...	se alguém é acuso de "um" crime, por "uma" pes...	se alguem e acuso de "um" crime, por "uma" pes...	alguém acuso crime pessoa alguém á acusado vár...	alguem e acuso crime pessoa e alguem a acusado...
4	0	O que temos de concreto é a delação de Marco...	o que temos de concreto é a delação de marco...	o que temos de concreto é a delação de marcos...	o que temos de concreto e a delacao de marcos...	o concreto a delação marcos valério o mentor...	o concreto e a delacao marcos valerio o ment...
...	...	...	...	...	...	...	...
1569	1	Que apelação! kkkkkkk,	que apelação! kkkkkkk,	que apelação! kkkkkkk,	que apelacao! kkkkkkk,	apelacão kkkkkkk rindo	apelacao kkkkkkk rindo

Figura 2. *DataFrame Pandas com os pré-processamentos do PLN.*

Antes da definição das colunas geradas, é importante ressaltar que o pré-processamento ocorreu a partir da segunda coluna, ou seja, a geração da terceira coluna dependeu dos dados da primeira e assim por diante. Na segunda coluna do DataFrame (comentário) é apresentado o comentário na sua versão original, ou seja, sem nenhum tipo de tratamento. Na segunda coluna é apresentado o comentário após a conversão de todas as *strings* em letras minúsculas e remoção de parágrafos vazios. Na quarta coluna, chamada de *whitespacenone*, é apresentado o comentário sem os espaços extras. Na quinta coluna, chamada *unidecode*, foram removidos os caracteres que não constam na tabela UTF-8, padrão dos modelos na biblioteca *scikit-learn*. Exemplo: "ç", "^".

A coluna comentário_final_su, em um primeiro processo, receberam como padrão a coluna *whitespacenone* e ambas obtiveram o pré-processamento mais avançado, no qual foi utilizado o método de *tokenização* do texto, que tem como objetivo separar palavras ou sentenças em unidades. Depois, para cada unidade, fez-se a conferência se esta pertencia a um vetor de pontuações para suas remoções. Após essa conferência, comparou-se a unidade com uma lista de *stopwords*, as quais consistem em palavras que podem ser consideradas irrelevantes para o conjunto esperado, no caso, conectivos, artigos da língua portuguesa etc. Por fim, juntou-se as unidades novamente para gerar o comentário. A diferença entre as duas últimas colunas está no uso da coluna auxiliar *unidecode*, que foi utilizada apenas como um comparativo a mais na última coluna, fazendo com que seu pré-processamento se tornasse mais detalhado.

Uma vez realizado o pré-processamento, utilizou-se da biblioteca Scikit-Learn para gerar a Bag of Words (BOW), na qual o texto é representado como um conjunto de suas palavras, desconsiderando a gramática e a ordem das palavras e mantendo apenas as palavras. A partir do uso da BOW é possível treinar os classificadores de Aprendizado de Máquina com a frequência de ocorrência de cada palavra.

2.4. Criação do modelo de Aprendizado de Máquina

O Aprendizado de Máquina é utilizado para extrair padrões complexos de grandes conjuntos de dados. Frequentemente, o objetivo do Aprendizado de Máquina é prever uma determinada resposta com base em uma ou mais variáveis preditoras. Existem muitos modelos de Aprendizado de Máquina, abrangendo métodos básicos, como regressão linear, regressão logística e modelos de árvore, bem como métodos mais sofisticados, como Redes Neurais Artificiais ou Máquinas de Vetores de Suporte (HASTIE *et al*., 2009).Normalmente, o Aprendizado de Máquina não restringe a análise de dados a apenas um modelo, mas compara muitos modelos e escolhe aquele que atinge a melhor predição.

A partir das BOW criadas das duas últimas colunas do DataFrame gerou-se, nesta etapa, um conjunto de dados para treino e teste, que foram aplicados a quatro diferentes modelos de Aprendizado de Máquina. As escolhas de modelos foram feitas seguindo as propostas literárias, tendo os algoritmos *BernoulliNB* e *GradientBoosting* apresentados resultados satisfatórios no trabalho de PAIVA (2020). O algoritmo *LogisticRegression* teve um resultado próximo do *BernoulliNB* no trabalho de GARG e BASSI (2016). Já o algoritmo *XGBoost* foi uma escolha neutra, de forma a adicionar sua funcionalidade à literatura.

3. Resultados

Foram extraídos um total de 13.505 comentários do *site* G1, da Globo, dentre um total de 108 notícias distintas. A partir destes dados, foram anotados um total de 2.352 comentários no processo de *crowdsourcing*, sendo eles 787 anotados como tóxicos e 1.565 não tóxicos.

Dentre essas 787 anotações, 337 (42,8%) foram categorizados pelos mesmos usuários como "insulto".

As métricas utilizadas para validar os resultados para cada modelo foram os F1-score, Acurácia, Precisão e Recall. Essas são medidas bem conhecidas, tendo sido utilizadas nos trabalhos relacionados, e são calculadas com base nos quatro valores gerados por qualquer classificador e apresentados nas matrizes de confusão, quais sejam: positivos verdadeiros (TP), negativos verdadeiros (TN), falsos positivos (FP) e falsos negativos (FN). O TP é o número de instâncias rotuladas corretamente como pertencentes à classe positiva. O TN é o número de instâncias rotuladas corretamente como pertencentes à classe negativa. O FP é o número de instâncias incorretamente rotuladas como pertencentes à classe positiva. O FN é o número de instâncias rotuladas incorretamente como pertencentes à classe negativa. O cálculo das métricas são apresentadas nas equações a seguir.

$$Precisão = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$Acurácia = \frac{TP + TN}{TP + FP + TN + FN}$$

$$F1\_score = 2\,\frac{Precisão \times Recall}{Precisão + Recall}$$

Após o pré-processamento e balanceamento da quantidade de comentários no *corpus*, foram definidos 1.544 comentários anotados para serem utilizados pelos algoritmos classificadores. Para as modelagens, os dados foram divididos em vetores de treino e teste para suas respectivas classes, no qual 1.180 itens foram utilizados para treino e 394 para teste.

O algoritmo *LogisticRegression*, utilizando o *corpus* com maior nível de pré-processamento de dados, gerou um resultado de 66.75% em acurácia, 71.65% em precisão, 65% de *F1-score* e 67% de *recall*. Sua curva de aprendizado atingiu 0.687 de acurácia, conforme podem ser vistos nos gráficos apresentados na Figura 3.

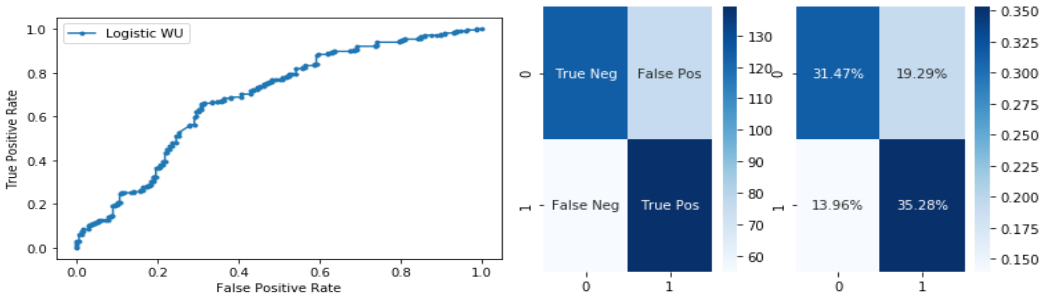

Figura 3. Curva de aprendizado e matriz de confusão para o algoritmo *LogisticRegression.*

Já o algoritmo *XGBoost* apresentou a mesma porcentagem de acurácia para as BOW geradas a partir das colunas *comentario_final_su* e *comentario_final_wu*, sendo ambas de 58.63%. Porém, esta última demostrou melhores resultados em *F1-score* e *recall* (0.01). Em

termos de Precisão, a última teve melhores resultados (63.55%) em relação à anterior, que obteve 61.71%. Os gráficos com a curva de aprendizado e a matriz de confusão para este algoritmo estão apresentados na Figura 4.

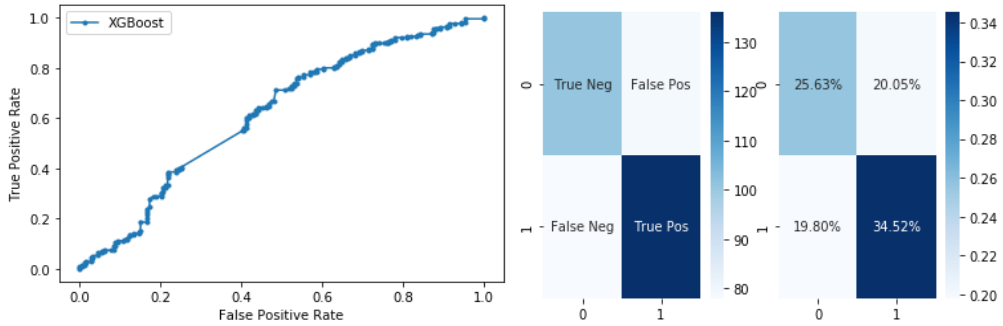

Figura 4. Curva de aprendizado e matriz de confusão para o algoritmo *XGBoost.*

No algoritmo *GradientBoosting* obteve-se *F1-score* e R*ecall* de 68% e uma Precisão de 67.49%. Na Figuras 5 são apresentadas a curva de aprendizado e a matriz de confusão deste algoritmo.

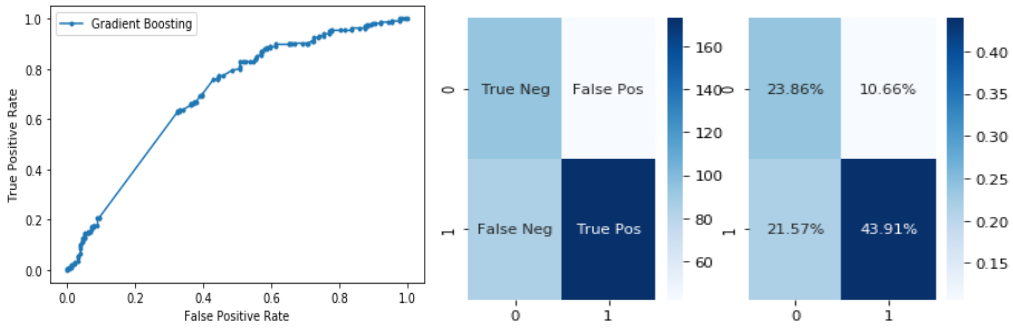

Figura 5. Curva de aprendizado e matriz de confusão para o algoritmo *GradientBoosting*.

Por fim, a última modelagem foi realizada utilizado o algoritmo *BernoulliNB* (Naive Bayes), o qual apresentou um resultado interessante: os melhores valores foram resultantes da base original, sem nenhum tipo de pré-processamento, sendo esses 67.77% em acurácia, 68.97% em precisão, 68% em F1-score e 68% de *recall*. Verifica-se também que sua acurácia na curva de aprendizado apresentou 0.706 e bons níveis em sua matriz de confusão, conforme podem ser observados nos gráficos da Figura 6.

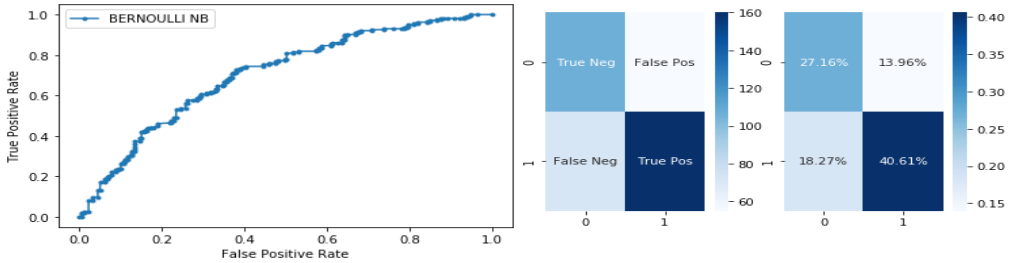

Figura 6. Curva de aprendizado e matriz de confusão para o algoritmo *BernoulliNB*.

4. Conclusão e Trabalhos Futuros

Após analisar os resultados, verificou-se que os melhores valores em acurácia ficaram em um intervalo de 68% e 70%, já os valores em *F1-score* estavam entre 65% e 68%, assim como o *recall*. As matrizes de confusão obtiveram números próximos quando observado os valores de verdadeiros positivos (TP).

Um resultado que chamou a atenção foi o da modelagem do *BernoulliNB*, que apresentou melhor desempenho utilizando a base crua do *corpus*, ou seja, o *corpus* sem nenhum tratamento de dados ou pré-processamento, pois a premissa principal, ao se utilizar modelos estatísticos de classificação para identificação de linguagem natural, é a necessidade de realizar o melhor pré-processamento possível. Essa premissa ainda pode ser considerada válida, pois o melhor desempenho em acurácia, curva de aprendizado mais próxima da teoria, assim como balanceamento de verdadeiros negativos e verdadeiros positivos foi utilizando o algoritmo *LogisticRegression* com o pré-processamento. Ou seja, conclui-se que alguns algoritmos, como o *BernoulliNB*, conseguiram identificar nuances no *corpus* não perceptíveis por outros (*LogisticRegression*, por exemplo).

Evidentemente, a quantidade de comentários anotados, assim como o processo de pré-processamento, influencia nos resultados, pois quanto maiores os vetores de treino e teste, maior é a capacidade de identificação dos resultados e de aprendizado.

Propõe-se, como continuidade deste trabalho, a manutenção do *crowdsourcing* para a anotação dos dados obtidos, de forma a aumentar o tamanho do *corpus*, e melhorar os modelos de Aprendizado de Máquina. No que se refere ao Processamento de Linguagem Natural, propõe-se também a aplicação de técnicas no estado da arte, tais como *Word Embeddings* e BERT (*Bidirectional Encoder Representations from Transformers*), tendo ainda como desafio a criação dos vetores de palavras em português do Brasil.

Referência Bibliográfica

ALMEIDA, Thais G.; SOUZA, Bruno Á.; NAKAMURA, Fabíola G.; NAKAMURA, Eduardo F. ***Detecting Hate, Offensive, and Regular Speech in Short Comments***. Webmedia '17: Proceedings of The 23rd Brazillian Symposium On Multimedia And The Web**.** Gramado, Out. 2017. p. 225-228.

BALOCCO, Anna Elizabeth; SHEPHERD, Tania Maria Granja. **A violência verbal em comentários eletrônicos: um estudo discursivo-interacional**. DELTA, São Paulo, v. 33, n. 4, p.1013-1037, dez. 2017.

BEHREND, Tara S.; SHAREK, David J.; MEADE, Adam W.; WIEBE, Eric N.. ***The viability of crowdsourcing for survey research***. Behavior Research Methods, [s.l.], v. 43, n. 3, p. 800-813, 25 mar. 2011. Springer Science and Business Media LLC.

BOUSFIELD, Derek. 2008. ***Impoliteness in interaction***. Amsterdam / Philadelphia: Jhon Benjamins. – Apud BALOCOO and Anna Elizabeth (2017).

BRETSCHNEIDER, Uwe; PETERS, Ralf. ***Detecting Offensive Statements towards Foreigners in Social Media***. In: Proceedings of the 50th Hawaii International Conference on System Sciences, pp. 2213-2222, 2017.

DAVIDSON, Thomas; WARMSLEY, Dana; MACY, Michael; WEBER, Ingmar. ***Automated Hate Speech Detection and the Problem of Offensive Languag***e. In: Proceedings of the 11th International Conference On Web And Social Media (ICWSM'2017), 2017.

GARG, Prateek; BASSI, Vineeta. ***Sentiment Anlysis of Twitter Data using NLTK in Python***. 2016. 42 f. Dissertação (Mestrado) - Curso de Computer Science, Computer Science And Engineering Department, Thapar University, Patiala, 2016.

GEORGAKOPOULOS, S. V., TASOULIS, S. K., VRAHATIS, A. G., and PLAGIANAKOS, V. P. ***Convolutional neural networks for toxic comment classification***. arXiv preprint arXiv:1802.09957, 2018 - Apud SILVA *et al*. (2018).

GRUS, Joel. ***Data Science from Scratch: first principles with python***. Sebastopol: O'reilly Media, 2015.

HASTIE, T.; TIBSHIRANI R.; FIREDMAN, J. H. ***The Elements of Statistical Learning: Data Mining, Inference, and Prediction***. Springer, 2009.

HOSSEINI, Hossein; KANNAN, Sreeram; ZHANG, Baosen; POOVENDRAN, Radha. ***Deceiving Google's Perspective API Built for Detecting Toxic Comments***. **ArXiv**, [*S. l.*], p. 1-4, 27 fev. 2017.

HOWE, Jeff. ***The Rise of Crowdsourcing***. 2006. Disponível em: https://www.wired.com/2006/06/crowds/. Acesso em: 21 mar. 2020.

HUANG, Qianjia; SINGH, Vivek Kumar; ATREY, Pradeep Kumar. ***Cyber Bullying Detection Using Social and Textual Analysis***. In: Proceedings of The 3rd International Workshop On Socially-Aware Multimedia [s.l.], p. 3-6, 2014. ACM Press.

LOPES, L. ; VIERA, R. **Processamento de Linguagem Natural e o Tratamento Computacional de Linguagens Científicas.** In: Cristina Lopes Perna; Heloísa Koch Delgado; Maria José Finatto. (Org.). Linguagens Especializadas em Corpora: modos de dizer e interfaces de pesquisa. Porto Alegre: EDIPUCRS, 2010, p. 183-201

LUBANOVIC, Bill. ***Introducing Python: modern computing in simple packages***. 2. ed. Sebastopol: O'reilly Media, 2015.

MCKINNEY, Wes. ***Python for Data Analysis: data wrangling with pandas, numpy, and ipython***. Sebastopol: O'reilly Media, 2018.

MITCHELL, Ryan. ***Web Scraping with Python: collecting data from the modern web***. Sebastopol: O'reilly Media, 2015.

NANDHINIA, B. SRI; SHEEBAB J. I. ***Online Social Network Bullying Detection Using Intelligence Techniques***. In: Proceedings of International Conference on Advanced Computing Technologies and Applications (ICACTA- 2015), pp. 485-492, 2015.

NOCKLEBY, John T.; LEVY, Leonard W.; KARST, Kenneth L. ***Hate Speech. In Encyclopedia of the American Constitution***. 2. ed. New York: Macmillan, 2000.

PAIVA, Peter Dias; SILVA, Vanecy Matias da; MOURA, Raimundo Santos. ***Hate speech detection using feature vectors applied to a new comment base in Portuguese***. Revista de Sistemas e Computação, Salvador, v. 10, n. 1, p. 59-68, jan/abr. 2020.

PELLE, Rogers Prates; MOREIRA, Viviane P. ***Offensive Comments in the Brazilian Web: a dataset and baseline results***. In: Brazilian Workshop on Social Network Analysis and Mining (BRASNAM), 6., 2017, São Paulo. Anais do VI Brazilian Workshop on Social Network Analysis and Mining. Porto Alegre: Sociedade Brasileira de Computação, jul. 2017. ISSN 2595-6094.

SCHIMDT, Anna; WIEGAND, Michael. ***A Survey on Hate Speech Detection using Natural Language Processing***. In: Proceedings of the 15th International Workshop on Natural Language Processing for Social Media, pp 1–10, 2017.

SILVA, Rosane Leal; NICHEL, Andressa; MARTINS, LEHMANN Anna Clara; BORCHARDT, Carlise Kolbe. **Discursos de Ódio em Redes Sociais: Jurisprudência Brasileira**. Revista Direito GV, vol. 7(2). pp. 445-468, 2011.

SILVA, Samuel C.; SERAPIÃO, Adriane B. S.. **Detecção de discurso de ódio em português usando CNN combinada a vetores de palavras**. In: Symposium on Knowledge Discovery, Mining and Learning, 2018., 2018, São Paulo. **Conference Paper.**

SOOD, Sara Owsley; ANTIN, Judd; CHURCHILL, Elizabeth F. ***Using Crowdsourcing to Improve Profanity Detection***. California: Association For The Advancement Of Artificial Intelligence, 2012.

THANAKI, JALAJ. ***Python Natural Language Processing***. Editora Packt, 2017.

WULCZYN, Ellery et al. ***Ex Machina: Personal Attacks Seen at Scale***. Www'17: Proceedings Of The 26th International Conference On World Wide Web**.** Perth, p. 1391-1399. 25 fev. 2017.

XIANG, Guang; FAN, Bin; WANG, Ling; HONG, Jason I.; ROSE, Carolyn P. **Incorporação do tempo em SGBD orientado a objetos**. *In*: Detecting Offensive Tweets Via Topical Feature Discovery Over a Large Scale Twitter Corpus, 12, 2012, Maui.

YANG, Min; MEI, Jincheng; JI, Heng; ZHAO, Wei; ZHAO, Zhou; CHEN, Xiaojun Chen. ***Identifying and Tracking Sentiments and Topics from Social Media Texts during Natural Disasters***. In: Conference on Empirical Methods in Natural Language Processing, pp. 527–533, Copenhagen, Denmark, 2017.