Relatório Detecção de Discurso de Ódio com Análise de Sentimentos - Abril de 2019

Woshington Valdeci de Sousa

Recife, Brasil

1. Introdução

O objetivo deste trabalho é analisar diversas abordagens para detecção de discurso de ódio, as abordagens variam entre pré-processamento, formas de extração de características e algoritmos de aprendizagem.

Os experimentos deste trabalho utilizam os dados apresentados em **Automated Hate Speech Detection and the Problem of Offensive Language**, o dataset[1] possui 24.783 registros distribuídos em 3 classes: *Hateful*, *Offensive* e *Clean*, sendo 1.430 classificados como *Hateful*, 19.190 como *Offensive* e 4.163 commo *Clean*.

1.1. Pré-Processamento

Com o objetivo de melhorar a extração das caracteristicas do banco de dados foram utilizadas técnicas de *Natural Language Processing*. As técnicas adotadas foram:

1. Remoção de URLS;
2. Remoção de RT;
3. Remoção de tags iniciadas em "@";
4. Remoção de números;
5. Remoção de sequencias com 2 espaços ou mais;
6. Remoção dos simbolos "#"e "&";
7. stemização;

[1] https://data.world/ml-research/automated-hate-speech-detection-data

1.2. Métodos de Extração

Para extração das características foram feitos experimentos com as seguintes técnicas:

1. CountVectorizer - Utiliza aborgadem *Bag-of-Words* para vetorização, não diferencia classe gramatical e ordem, os principais parametros utilizados neste trabalho foram:
   (a) *Range Ngram* - Variando de Bigramas e Trigramas, a quantidade de caracteristicas extraídas foi 4.878 para 2-gram e 5.139 para 3-gram;
   (b) *Stop Words* - Desconsiderando *Stop words* da lingua inglesa;
   (c) *min_df* - Ocorrência minima no corpus, variando de 1 e 5;
2. TFIDF - *Term Frequency-Inverse Document Frequency* classifica palavras com relação a frequência que ela aparece no corpus, quanto menor a ocorrência, maior é a probabilidade de ser relevante, os principais parametros utilizados neste trabalho foram:
   (a) *Range Ngram* - Variando de Bigramas e Trigramas, a quantidade de caracteristicas extraídas foi 4.878 para 2-gram e 5.139 para 3-gram;
   (b) *Stop Words* - Desconsiderando *Stop words* da lingua inglesa;
   (c) *min_df* - Ocorrência minima no corpus, variando de 1 e 5;
   (d) *use_idf* - Variando de *False* para *True* invertendo a frequência do texto no corpus.
3. Word2Vec - Consiste em uma técnica de vetorização baseada em similaridades, a abordagem é denominada *Word Embedding* e utiliza um modelo não supervisionado.

2. Experimentos

Inicialmente foi feita uma segmentação do dataset dividindo-o em 2 partes, 70% para treinameto e 30% para teste, segue a distribuição de cada conjunto.

1. Treinamento - Hateful: 1.003, Offensive: 13.457 e Clean: 2.888
2. Teste - Hateful: 427, Offensive: 5.733 e Clean: 1.2775

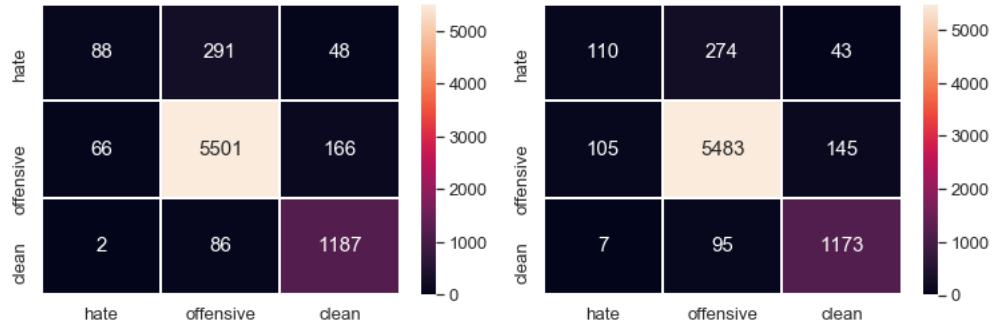
Foram utilizados diversas combinações de algoritmos e técnicas de extração, a Tabela 1 apresenta os resultados.

Modelo	CV	TFIDF	W2V
Logistic Regression	0.91	0.91	-
MultinomialNB	0.88	0.84	-
Ramdom Forest	0.90	0.90	0.83
SVM	0.89	0.89	-
ExtraTreesClassifier	-	-	0.82
DecisionTreeClassifier	-	-	0.73

Tabela 1: Resultados obtidos com combinação de técnicas

As Figuras 1, 2, 3, 4, 5 e 6 apresentam os acertos e erros de cada classificador, a principal informação extraída desses modelos é a necessidade de distinguir conteúdo ofensivo de conteúdo de ódio.

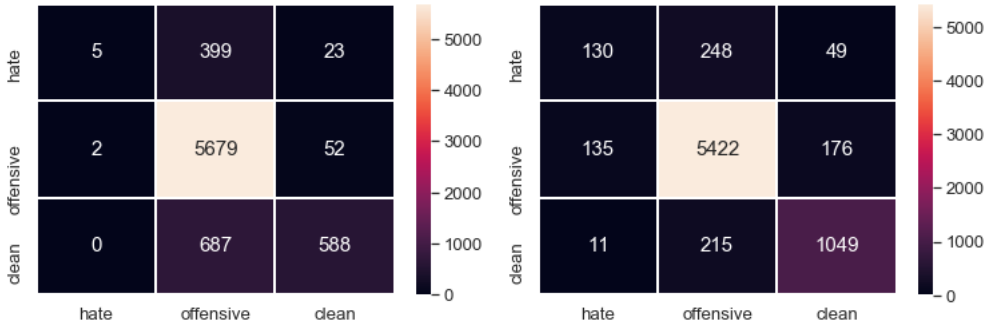
Figura 1: Matriz de confusão Logistic Regression



(a) TFIDF

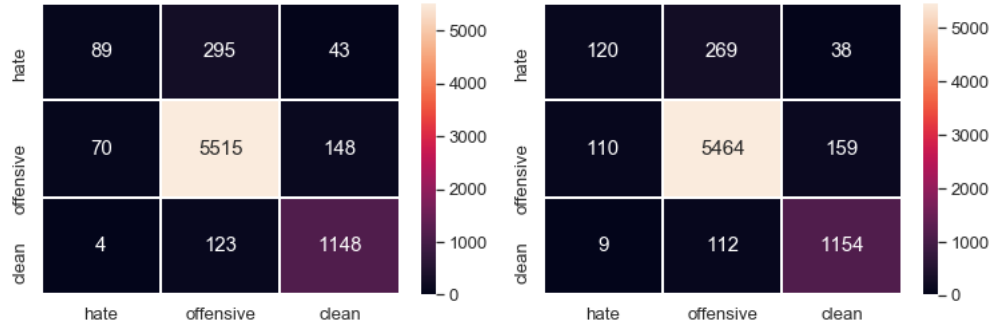
(b) Count Vectorizer

Figura 2: Matriz de confusão MultinomialNB



(a) TFIDF (b) Count Vectorizer

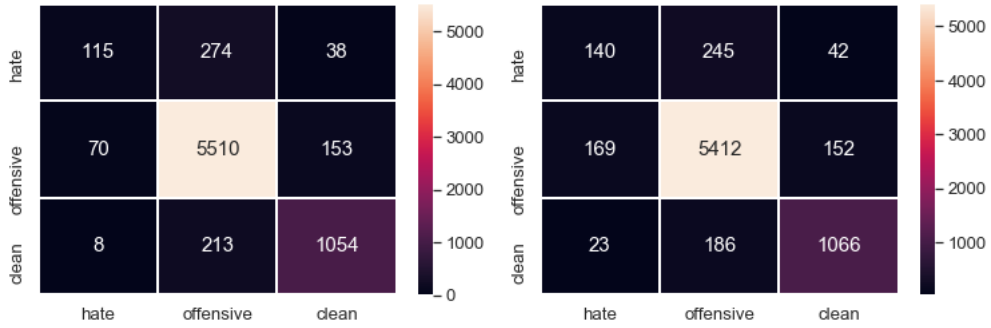
Figura 3: Matriz de confusão Random Forest



(a) TFIDF (b) Count Vectorizer

(c) Word2Vec

Figura 4: Matriz de confusão SVM



(a) TFIDF (b) Count Vectorizer

Figura 5: Matriz de confusão Extra Tree - W2V

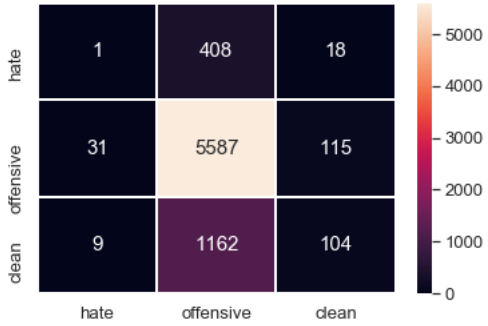
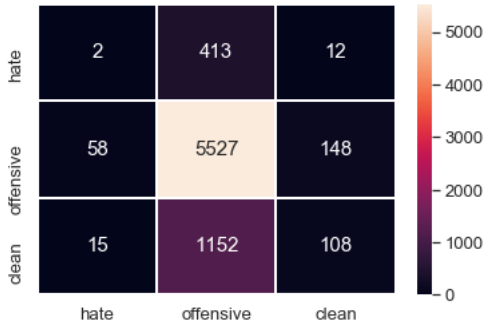


Figura 6: Matriz de confusão Decision Tree - W2V



3. Considerações Finais

Apesar de ter conseguido uma acurácia em torno dos 90% a dintinção entre as classes mais próximas ainda é um problema, não foram encontrados trabalhos que utilizam o mesmo *dataset* o que dificulta a comparação com outros modelos e técnicas, no entanto foi feita a solicitação de conjuntos relacionados a outros trabalhos.

3.1. Atividades Futuras

Para dar proseguimento ao trabalho algumas atividades estão sendo estudadas, as principais são:

1. Avaliar o *script* de limpeza dos dados para averiguar se pode ser melhorado em algum aspecto, houve uma tentativa de tokenização utilizando a técnica PoS Tag, no entanto a precisão diminuiu ao aplicá-la;
2. Buscar outras formas de extrair as caracteristicas, possibilidade de utilizar *glover* e/ou *Bert*;
3. Na literatura revisada até o momento os principais algoritmos utilizados estão sendo testados, no entanto pode-se buscar outros modelos de aprendizagem para buscar melhorar a classificação;
4. Buscar combinar modelos de aprendizagem;
5. Utilizar GridSearchCV do sklearn, ferramenta que permite adicionar várias possibilidades de parametros e retornar a melhor combinação em forma de modelo, essa técnica já foi utilizada nesses experimentos, no entanto apenas para um modelo de aprendizagem;