Análise de Spam

Wana Batista BArbosa
batista.wana@gmail.com.br

1. Introdução

Com os avanços da internet, o e-mail se tornou uma das ferramentas mais populares par troca de informações, agilizando a comunicação entre as pessoas. Juntamente com seus benefícios, também emergiram problemas para os usuários, como o aumento excessivo do volume de e-mails indesejados. Os spams, como são denominados, causam grandes incovenientes para os usuários, sem os filtros anti-spam seria imensamente trabalho encontrar um e-mail comum dentre tanto lixo-eletrôncio, além das propragandas esses emails também podem conter conteúdos maliciosos (vírus), o que ameaça a segurança das informações do usuário. Com tantos prejuízos criou-se uma urgência na necessidade de desenvolver filtros anti-spam mais confiávei e robustos. Os mais modernos são baseados em métodos de aprendizado de máquina e já comprovam sucesso na detecção de spam [1].

No presente estudo é realizado uma análise exploratória para a base de dados fornecida pela empresa Senior Labs contento e-mails dos três primeiros meses do ano 2017, assim como a investigação de algoritmos de inteligência artificial para a detecção de spams.

2. Metodologia

Utilizou-se as ferramentas *jupyter notebook* e linguagem de programação *Python* para o desenvolvimento da análise e classificação das mensagens, tendo como base de dados o arquivo no formato csv, contento exemplos de mensagens comuns e spams, sendo 4827 mensagens comuns e 747 mensagens spams. O notebook segue uma sequência lógica de desenvolvimento, iniciando com o tratamento do texto encontrato no atributo *Full_text, para esse conteúdo realizou-se a* remoção de caracteres indesejados e *stopwords, por exemplo. Seguido da análise exploratória, onde são extraídos valores estatísticos (fequência de palavras, mínimos, máximos, médias, desvio padrão, variância contagem de classes, etc.) e insights, por meio da visualização gráfica dos resultados e finalizando com a classificação, onde são comparados dois algoritmos, RandomForestClassifier e MultinomialNB assim como métodos para tratar base de dados desbalanceados (Upsample, Downsample).*

3. Resultados

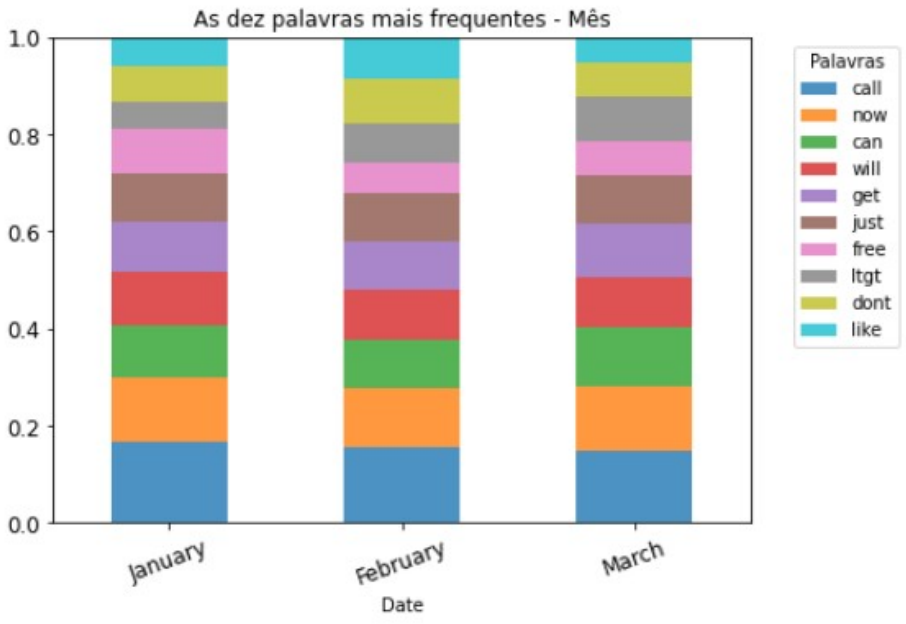
3.1 Análise

A primeira análise feita após o tratamento do texto, foi a vizualização das palavras mais frequentes, por meio de um *Wordcloud* (em português, nuvem de palavras) Figura 1. Nota-se que as palavras *love, free day e time* são as mais frequêntes nas mensagens.

Figura 1. Palavras mais frequentes

A base de dados possui 149 colunas com valores inteiros que indicam a frequência de uma determinada palavra em uma mensagem ("got"... "wan"), as dez (10) palavras frequentes por mês podem ser visualizadas no gráfico normalizado entre 0 e 1 (Figura 2).

Figura 2. As dez palavras mais frequentes - Mês



A análise quantitativa mensal para cada classe, spam e não- spam (comum), são msotradas na tabela e gráficos abaixo, Figura 3. É possível observar uma grande diferença de quantidades entre as classes para todos os meses, isso significa que estamos diante de um dataset desbalanceado, onde a classe de mensagens comuns predomina, como mostrado na Figura 4, o conteúdo não-spam (no) é maioria com 86.6%, contra apenas 13.4% de spam (yes).

Figura 3. Análise quantitativa das classes

Date	spam	comum
2017-01-31	266	1687
2017-02-28	244	1512
2017-03-31	237	1628

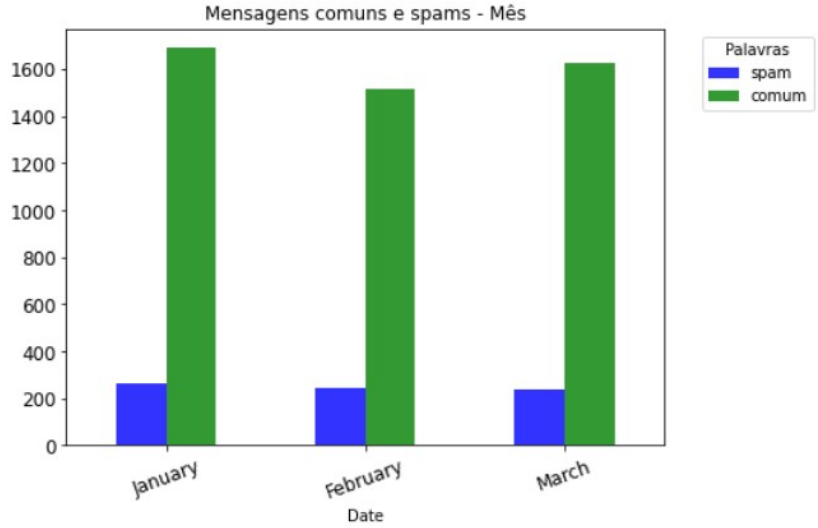
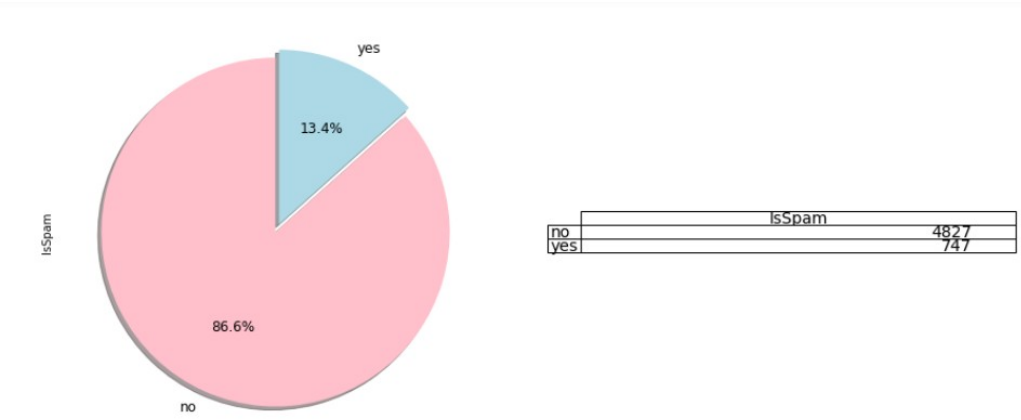


Figura 4. Porcentagem por classe



As informações estatísticas, mínimo, máximo, média, desvio padrão, variância, foram extraídas com base no atributo *Word_Count, coluna* contendo a quantidade total de palavras da mensagem Tabela 1. A Tabela 2, exibe os máximos para cada mês e seu respectivo dia para mensagens comuns.

Tabela 1. Valores estatísticos por mês

Date	max	min	mean	median	std	var
2017-01-31	190	2	16.336918	13	12.557171	157.682535
2017-02-28	100	2	16.029043	13	11.042459	121.935908
2017-03-31	115	2	16.285255	12	11.576213	134.008715

Tabela 2. Máximos por mês e dia

	month	day	comum
12	Feb	13	72
28	Jan	01	69
66	Mar	08	69

3.1 Classificação

Para a classififcação foram comparados dois algoritmos, *RandomForestClassifier e MultinomialNB,* ambos pertencentes ao pacote scikit-learn, e métodos para tratar base de dados desbalanceados *(Upsample, Downsample)*, como visto nas Figuras 3 e 4. O problema de se treinar um algoritmo com base desbalanceada está no fato de que o modelo, possívelmete irá aprender muito bem uma classe e nem tanto a outra, podendo retornar uma predição desproporcional. Upsampling é o processo de duplicar aleatoriamente as observações da classe minoritária para aumentar seu número de amostras (Figura 5), enquanto que downsampling é a redução da amostragem de forma aleatória de observações da classe majoritária aproxímando seu número de amostras ao número de amostras da classe minoritária, Figura 6. Os resultados para os modelos investigados estão na Tabela 3.

Figura 5. Resultados para a metodologia Upsampling

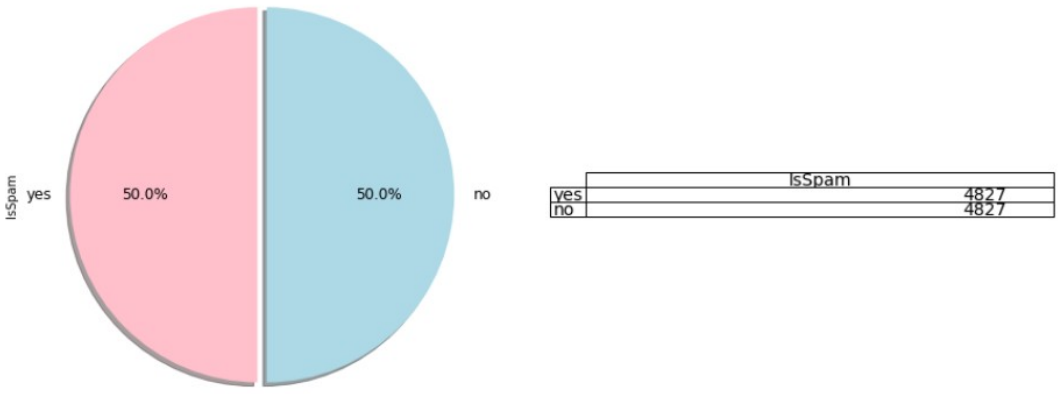


Figura 6. Resultados para a metodologia Downsampling

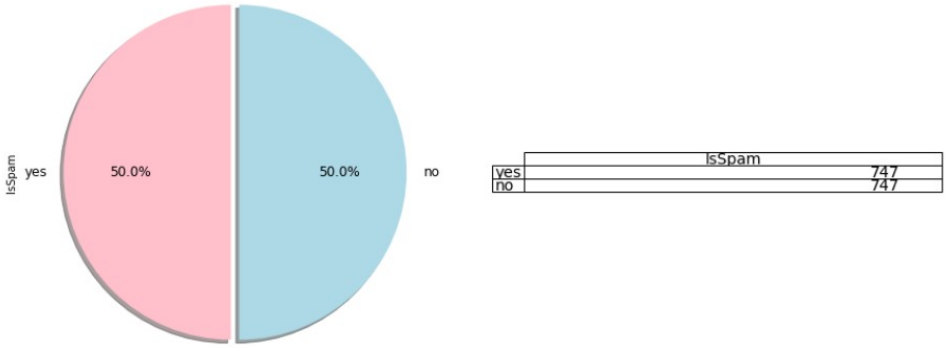


Tabela 3. Resultados obtidos para os modelos e datasets utilizados

Modelo	Base original		Upsampled		Downsample	
	Precisão	Tempo	Precisão	Tempo	Precisão	Tempo
MultinomialNB	0.95	0.012	0.92	0.014	0.89	1.48
RandomForestClassifier	0.96	0.5	0.97	0.86	0.91	1.82

Podemos observar que para o dataset original, mesmo extremamente desbalanceado, os resultados da classificação foram relativamente bons (acertando acima de 80% ambas as classes para ambos os modelos), entretanto é notório que uma classe está sendo acertada mais que a outra. Esse é um dos efeitos do desbalanceamento de classes. De acordo com os resultados, pode-se observar que existe um equilibrio de acertos para ambas as classe, Tando para o Upsampled quanto para Downsampled, sendo o Upsampled juntamente com o algoritmo RandomForestClassifier os responsáveis pelo melhor resultado com 0.97 de precisão.

Conclusão

Neste estudo, investigou-se métodos de classificação para mensagens de e-mails com o intuito de identificar conteúdos de *spam*. É notório a complexidade desse desafio, visto que ao longo dos anos vem ocorrendo uma evolução desses conteúdos que buscam se camuflar para passar evitar os filtros anti-spams. Os filtros anti-spam são essênciais para que continue sendo possível a utilização de e-mail como ferramneta de comunicação.

As abordagens utilizadas retornaram resultados promissores para a detecção de spams. Levando em consideração que são resultados preliminares, provenientes de testes realizados em apenas uma base de dados de fonte única. É prudente citar a necessidade de realizar pesquisas e testes adicionais com bases maiores e mais diversificada para validar os resultados obtidos e melhorá-los. Dentro desse contexto, vale ressaltar a eficácia e influência das técnicas de tratamento de dados, as quais possibilitaram um melhor desempenho dos modelos treinados.

Referências

[1] Dada, E. G., Bassi, J. S., Chiroma, H., Abdulhamid, S. M., Adetunmbi, A. O., & Ajibuwa, O. E. (2019). *Machine learning for email spam filtering: review, approaches and open research problems. Heliyon, 5(6), e01802.* doi:10.1016/j.heliyon.2019.e01802