Extração de dados e Classificação de mensagens spam

Paulo Mateus da Silva
paulomatew@gmail.com

Introdução

Trocas de e-mails são comuns no dia-a-dia de muitas pessoas, porém um dos principais problemas que os usuários se deparam é o recebimento de spams. Segundo a definição encontrada em [1], “Spam é o lixo eletrônico digital: comunicações não solicitadas enviadas em massa pela internet, ou por qualquer sistema de mensagens eletrônicas”.

Para resolver problemas de recebimento de spam, algoritmos de classificação de texto têm sido adotados para sistemas de classificação de e-mails. Os algoritmos mais comuns são o Naive Bayes e Support Vector Machine (SVM). Neste artigo serão apresentadas informações extraídas da base de dados dos e-mails disponibilizada, bem como uma sugestão de classificador para mensagens spam.

Metodologia

Foram realizadas buscas em diversos sites para obter uma visão geral do que é utilizado na resolução do problema em questão, os termos “spam”, “classifier”, “filter”, “algorithm” foram utilizados. Através da leitura de alguns trabalhos, foi possível identificar que o algoritmo de Naive Bayes é um dos mais utilizados para esse tipo de tarefa e o algoritmo de Support Vector Machine também é citado com frequência, portanto, serão utilizados no artigo.

Várias fontes como [6] mostram que hoje, a linguagem mais utilizada para resolver problemas que envolvem mineração de dados é Python, porém o autor do artigo utilizou Java, por familiaridade com a linguagem, para extração dos dados, e o software Weka em [8] versão 3.8.4 para geração dos modelos de classificação. Todos os códigos desenvolvidos podem ser obtidos em [7]. Para a Primeira Etapa, nas questões 1 e 2, o Google Sheets foi utilizado para gerar os gráficos a partir da saída dos arquivos implementados. Para Segunda Etapa, na parte de pré-processamento de dados foi criado o arquivo Weka.arff em [7], no formato padrão reconhecível pelo Weka 3.8.4. Foi removida a coluna texto e separado a data em dois novos atributos “data” e “hora”.

Resultados

Primeira Etapa

A primeira questão propôs que fosse gerado um gráfico com as palavras mais repetidas na base de dados. A imagem A1 foi gerada a partir das 10 palavras mais utilizadas na base. As demais palavras da base de dados podem ser visualizadas no arquivo Questao1.java em [7]. A segunda questão propôs exibir um gráfico com as mensagens comuns e spams separados por mês, é possível ver a resposta para essa questão na Imagem A2, para montar o gráfico foi utilizado a saída do arquivo Questao2.java em [7]. A Questão 3 pede que seja calculado o valor máximo, mínimo, média, mediana, desvio padrão e a variância da quantidade total de palavras (atributo Word_Count) para cada mês da Base, é possível visualizar essas informações na Tabela A1, que foi alimentada pela saída do arquivo Questao3.java em [7]. A questão 4 pede para exibir o dia do mês que possui a maior sequência de mensagens comuns (não spam), os resultados se encontram na Tabela A2, alimentada pelo arquivo Questao4.java em [7].

Imagem A1: As 10 palavras mais utilizadas da base de dados.

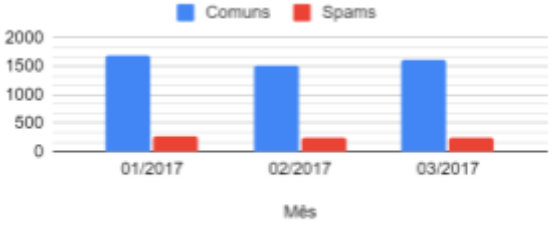

Imagem A2: Somatório de mensagens separadas por mês e pelos tipos comum e spam.

Mês	Máximo	Mínimo	Média	Mediana	Desvio Padrão	Variância
01/2017	190	2	16,3369	13,00	12,5540	157,6825
02/2017	100	2	16,0290	13,00	11,0393	121,9359
03/2017	115	2	16,2853	12,00	11,5731	134,0087

Tabela A1: Informações extraídas do banco de dados.

Mês	Dia	Maior Sequência
01/2017	26	31
02/2017	04	39
03/2017	31	46

Tabela A2: Maior sequência por dia de emails comuns (não spams).

Segunda Etapa

Através da leitura dos trabalhos relacionados, foram utilizados dois classificadores, Naive Bayes e SVM, como visto na Tabela A3. Mais informações sobre os modelos dos classificadores podem ser vistas no diretório “classificadores” em [7].

Classificador	Classificação Correta	Classificação Incorreta
SVM	95.8737%	4.1263%
Naive Bayes	94.4564%	5.5436%

Tabela A3: Informações sobre os classificadores.

Conclusão

A partir da avaliação feita dos dois modelos gerados pelo Weka para os dois classificadores, é possível perceber que o algoritmo SVM apresenta melhores resultados em relação a predição de mensagens de Spam, porém a diferença entre os dois algoritmos é pequena e ambos possuem resultados satisfatórios. Ambos os classificadores podem ser utilizados como um classificador de spam.

Referências

[1] https://www.avast.com/pt-br/c-spam#topic-1; Disponível em 25/01/2021.

[2] JATANA, Nishtha; SHARMA, Kapil. Bayesian spam classification: Time efficient radix encoded fragmented
database approach. In: 2014 International Conference on Computing for Sustainable Global Development
(INDIACom). IEEE, 2014. p. 939-942.

[3] CHAE, M. K. et al. Spam filtering email classification (SFECM) using gain and graph mining algorithm. In: 2017
2nd International Conference on Anti-Cyber Crimes (ICACC). IEEE, 2017. p. 217-222.

[4] HERSHKOP, Shlomo; STOLFO, Salvatore J. Combining email models for false positive reduction. In:
Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining. 2005.
p. 98-107.

[5] YIN, Hu; CHAOYANG, Zhang. An improved bayesian algorithm for filtering spam e-mail. In: 2011 2nd
International Symposium on Intelligence Information Processing and Trusted Computing. IEEE, 2011. p. 87-90.

[6] https://minerandodados.com.br/por-que-o-python-e-a-linguagem-mais-adotada-na-area-de-data-science/; Disponível em 25/01/2021.

[7] https://github.com/TroniPM/senior-challenge ; Disponível em 25/01/2021.

[8] https://www.cs.waikato.ac.nz/ml/weka/; Disponível em 25/01/2021.