ISNN: 2358-6052

Algoritmo genético aplicado a uma análise de sentimentos: resultados parciais

Breno Costa Dolabela Dias [1]; Alan Cândido de Souza [2]; Bruno Alberto Soares Oliveira [3]; Frederico Gadelha Guimarães [4]

1 Programa de Pós-Graduação em Engenharia Elétrica, UFMG, Belo Horizonte – MG; brenodolabela@gmail.com
2 Programa de Pós-Graduação em Engenharia Elétrica, UFMG, Belo Horizonte – MG
3 Programa de Pós-Graduação em Engenharia Elétrica, UFMG, Belo Horizonte – MG; brunoalbertobambui@ufmg.br
4 Machine Intelligence and Data Science (MINDS) Laboratory, UFMG, Belo Horizonte – MG

RESUMO

À medida que a tecnologia avança, é gerada uma quantidade enorme de dados, especialmente dados textuais. Tornou-se impossível analisar manualmente todos os dados para um propósito específico. Com isso, novas direções de pesquisa emergiram da análise automática desses dados, como a análise automática de sentimentos. A análise de sentimentos é então a tarefa de classificar as informações subjetivas de textos, como opiniões e sentimentos. Essa área atraiu a atenção dos pesquisadores por causa de suas aplicações em diferentes campos. Por exemplo, empresas podem aplicar análise de sentimentos para entender a respeito do impacto de suas campanhas de marketing. A computação evolucionária é uma família de algoritmos para otimização global inspirada na evolução biológica, sendo um subcampo da inteligência artificial. Em termos técnicos, eles são uma família de solucionadores de problemas de tentativa e erro de base populacional com um caráter de otimização metaheurística ou estocástica. Na computação evolucionária, um conjunto inicial de soluções candidatas é gerado e iterativamente atualizado. Cada nova geração é produzida pela remoção estocástica das soluções menos desejadas e pela introdução de pequenas alterações aleatórias. Na terminologia biológica, uma população de soluções é submetida a seleção natural (ou seleção artificial) e mutação. Como resultado, a população evoluirá gradualmente para aumentar à aptidão, neste caso a função de adequação escolhida do algoritmo. Este trabalhou buscou aplicar técnicas de computação evolucionária a uma análise de sentimentos, buscando otimizar a assertividade de classificação para diferentes reviews de filmes. Este algoritmo foi testado considerando diferentes variáveis e seus resultados foram descritos neste trabalho.

INTRODUÇÃO:

A análise de sentimentos tem como principal objetivo desenvolver técnicas automáticas capazes de extrair informações subjetivas em textos, como opiniões e sentimentos, a fim de apoiar a tomada de decisão, conforme descrito por BENEVENUTO et al. (2015). Pode ser feito a partir de duas técnicas: As técnicas supervisionadas e não supervisionadas.

As técnicas supervisionadas utilizam textos com as informações subjetivas previamente classificadas, que a partir de algoritmos de aprendizado, buscam inferir o sentimento de outros textos. Já as técnicas não supervisionadas não utilizam textos previamente classificados. Entre elas, destacam-se as abordagens léxicas, que classificam o texto a partir de um dicionário com palavras previamente classificadas.

Algoritmos evolucionários são um grupo de técnicas estocásticas que buscam abstrair princípios evolucionários da natureza em algoritmos que tentam encontrar a melhor solução de um problema. Tem como princípio que indivíduos dentro de um ambiente com recursos limitados são influenciados pela seleção natural, sobrevivendo aqueles mais aptos (EIBEN e SMITH, 2015).

O algoritmo mais popular entre os algoritmos evolucionários é o algoritmo genético, que modela as possíveis soluções para o problema na forma de uma string, possuindo features em suas diferentes posições. Para tentar chegar à solução ótima são realizadas operações análogas ao cruzamento entre os indivíduos como o bit-string crossover, em que subsets das strings de dois indivíduos são trocadas. E também operações análogas à mutação genética como o bit-flipping mutation, em que um indivíduo tem um de seus bits alterados (SIVANANDAM E DEEPA, 2008).

No trabalho de DRAGONI (2019), foram utilizadas técnicas de computação evolucionária para aplicar análise de sentimento a fim de inferir a polaridade de textos de diversos contextos. A estratégia evolucionária foi feita com o auxílio de ferramentas que exploram o significado e a relação das palavras, como o SenticNet, WordNet e ConceptNet. Os cruzamentos foram feitos de forma a beneficiar indivíduos

ISNN: 2358-6052

que com base em suas polaridades definidas para as diferentes palavras, conseguem de forma mais assertiva inferir sobre a polaridade dos diversos textos. Além disso, as mutações foram aplicadas de forma que a polaridade de palavras com significados parecidos fossem similarmente afetadas, diminuindo assim a sua aleatoriedade.

No trabalho de IQBAL et al. (2019) e ERNAWATI et al. (2018) foram aplicadas técnicas de Algoritmo Genético no processo de Feature Selection. Essa técnica foi aplicada em diversos tipos de reviews, como filmes e-commerce, conseguindo assim uma maior assertividade ao utilizar algoritmos de classificação.

Aplicando técnicas de computação natural este trabalho buscou o inferir o sentimento de reviews de filmes no IMDb. Foram aplicadas técnicas de otimização que ao longo de diversas iterações foram dadas polaridades às diferentes palavras contidas nos textos, buscando assim inferir qual o sentimento contido em cada Review (se é positivo ou negativo).

METODOLOGIA:

DADOS UTILIZADOS E PRÉ-PROCESSAMENTO

Para este trabalho foram utilizados um conjunto de 480 reviews do IMDb. Esses 480 reviews continham 11097 palavras distintas, incluindo números. Para se tornar mais fácil e possível que fosse trabalhado, foi feito um pré-processamento, em que foram removidos números, acentos e pontuações. Além disso, foram removidos nomes, como de empresas, filmes e atores utilizando a biblioteca spaCy, além das stop words que são palavras que não agregam para o significado como “and” e “the”, se tornando assim 5489 palavras.

Ainda com o pré-processamento feito, sobraram 5849 palavras distintas. Essas palavras estavam dispostas em uma Document-term matrix, uma matriz que possui as frequências de cada palavra em cada review. Sendo um número muito alto para se considerar como variável em um algoritmo genético foi necessário diminuir a quantidade de variáveis. Dessa forma, foi aplicado um método de Truncated Singular Value Decomposition (SVD). Esse método, de decomposição é apropriado para matrizes esparsas como a que se tinha em questão. Com isso, as 5849 diferentes palavras se tornaram 200 valores singulares que juntos conseguiam explicar 80% da variação dos textos.

REPRESENTAÇÃO DE CADA INDIVÍDUO E POPULAÇÃO INICIAL

Neste trabalho foi então aplicado o algoritmo genético como sendo a composição da polaridade de cada um dos 200 valores singulares presentes no texto. A população inicial foi gerada aleatoriamente como sendo um número de -1 até 1 para cada uma das 200 variáveis. A inicialização ocorreu na função de início da classe.

CÁLCULO DE FITNESS

Para calcular a fitness, foi somada a polaridade de cada variável no indivíduo multiplicado pelo valor de cada um dos componentes de cada review. E então para cada review, esse produto é somado, e se maior que zero, é considerado então que o review é positivo. Se for menor que zero, é inferida uma polaridade negativa para este review em determinado indivíduo. E então a quantidade de classificações corretas é dividido pelo tamanho total da população, obtendo assim a acurácia. No exemplo representado na Figura 1, um indivíduo [-1,-1,1,1,1,1] ao ter a soma dos seus produtos pelos valores singulares [1,2,1,2,1,2] de um review, obtém a soma 3. Dessa forma, é inferido que é um review positivo.

ISNN: 2358-6052

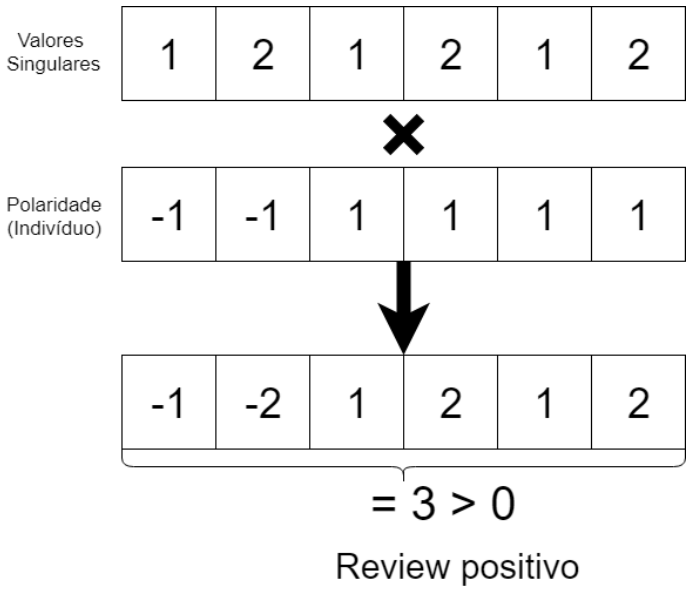

Figura 1: Exemplo de cálculo de fitness.

SELEÇÃO DOS PAIS

Para fazer a seleção dos pais foi aplicado o torneio. Nele, foi selecionada aleatoriamente uma quantidade de indivíduos da população para que dentro dessa amostra, fossem selecionados aqueles dois mais aptos para cruzarem e terem filhos. Ou seja, aleatoriamente selecionou alguns indivíduos, e dentre estes selecionados se escolheu aqueles dois que possuem o melhor fitness.

CRUZAMENTO

Para o cruzamento, foi aplicado o método two-point Crossover. Ele aleatoriamente escolhe pontos nas variáveis dos indivíduos. E então são criados dois novos indivíduos sendo cada um deles um dos pais com as variáveis definidas entre os pontos sorteados de outro pai. Por exemplo, se existe um indivíduo [1,1,1,1,1,1] e outro indivíduo [2,2,2,2,2,2] e são sorteados o terceiro e o quinto ponto, então as novas variáveis serão [1,1,2,2,2,1] e [2,2,1,1,1,2], conforme a Figura 2.

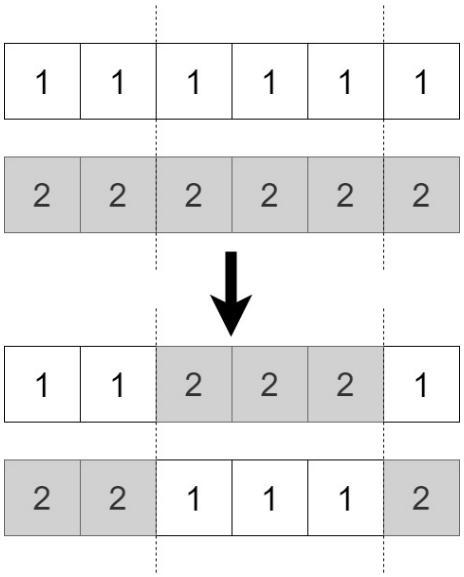

Figura 2: Exemplo de two-point crossover.

ISNN: 2358-6052

MUTAÇÃO

Na mutação, foram selecionados aleatoriamente alguns indivíduos para serem mutados. Estes indivíduos tiveram algumas de suas polaridades modificadas Depois, é aleatoriamente selecionada essa quantidade de polaridades em cada indivíduo sorteado para ter seu valor modificado. Este valor modificado foi somado ou subtraído 0.1. Além desse processo, também ocorreu outra mutação em que o indivíduo com a melhor fitness sofreu mutação ocupando o lugar do indivíduo com a pior fitness.

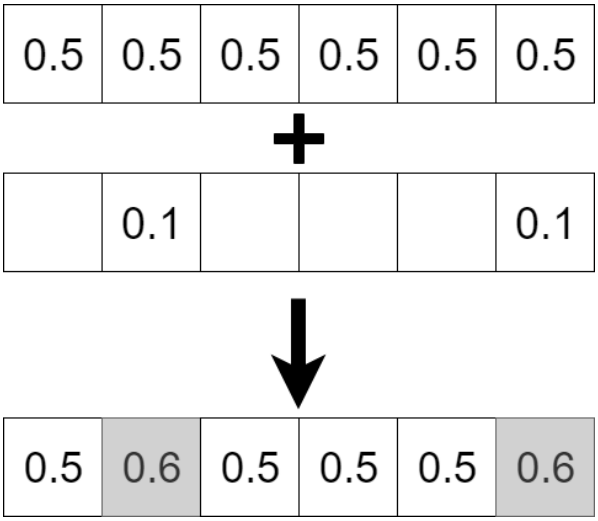

Figura 3: Exemplo de mutação.

SELEÇÃO DOS SOBREVIVENTES

Por último, é calculada a fitness de cada um dos indivíduos. Tendo esse valor calculado, para que o tamanho da população se mantenha igual à quantidade de indivíduos antes do cruzamento é eliminado as duas soluções com o pior valor.

RESULTADOS E DISCUSSÕES:

A implementação do algoritmo foi feita utilizando a linguagem Python 3.7, e Os gráficos dos experimentos foram gerados pelo Software Tableau. Além disso, os testes foram feitos considerando:

- Tamanho da população variando entre 10 e 20.
- Percentual de indivíduos mutados variando entre 10% e 30%.
- Percentual da população levada ao torneio variando entre 30% e 70%.
- Percentual dos valores singulares mutados variando entre 1% e 3%.
- 31 testes para cada combinação.
- 200 gerações.

Foi analisada diferença da média do melhor resultado variando o tamanho da população, conforme mostrado na Figura 4. É possível constatar uma melhora no algoritmo quando a população é 20. Porém se for considerar a quantidade de avaliações de fitness como sendo um fator limitante, ou seja, tendo uma quantidade finita de esforço computacional, os testes com população de tamanho 10 desempenharam melhor. Ao final da 200ª geração a média do melhor fitness foi superior à média do melhor fitness da centésima geração da população de tamanho 20.

ISNN: 2358-6052

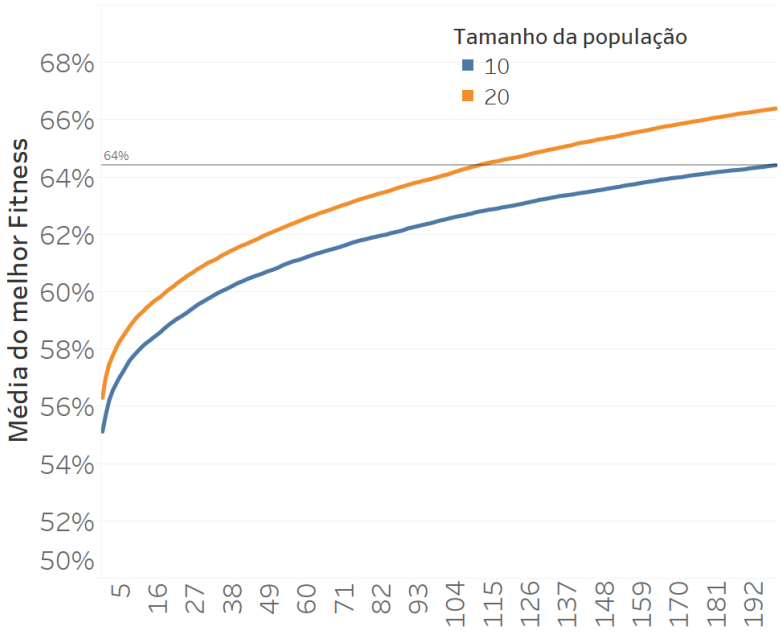

Figura 4: Média da melhor solução do algoritmo em relação ao tamanho da população.

Na Figura 5, foi analisado o desempenho dos testes com diferentes percentuais de indivíduos mutados. Os testes em que 30% dos indivíduos foram mutados tiveram um resultado muito superior aos testes em que apenas 10% dos indivíduos foram mutados. Logo, é possível constatar que a variação causada pela maior quantidade de mutações foi benéfica para o problema em questão.

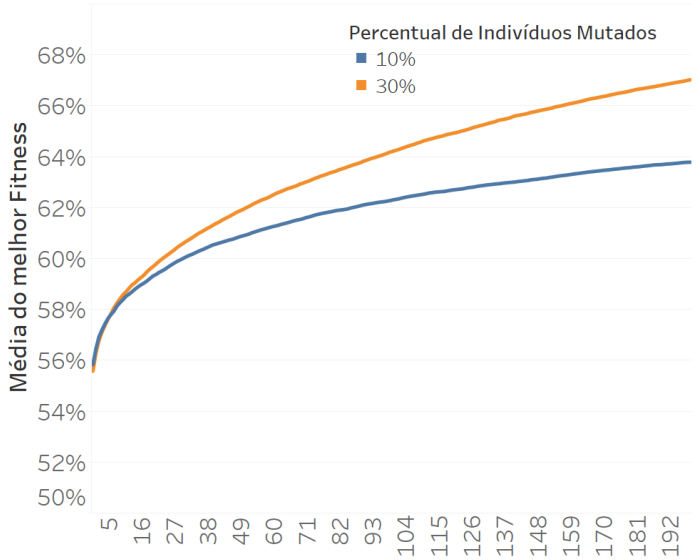

Figura 5: Média da melhor solução do algoritmo em relação ao percentual de indivíduos mutados.

Na Figura 6 foi analisado o desempenho de testes com diferentes percentuais de indivíduos levados para o torneio. Apesar de inicialmente o algoritmo com 70% dos indivíduos sendo levados para o torneio tendo bom resultado, no final quem acabou desempenhando melhor foram os testes que tiveram 30% dos indivíduos levados para o torneio.

ISNN: 2358-6052

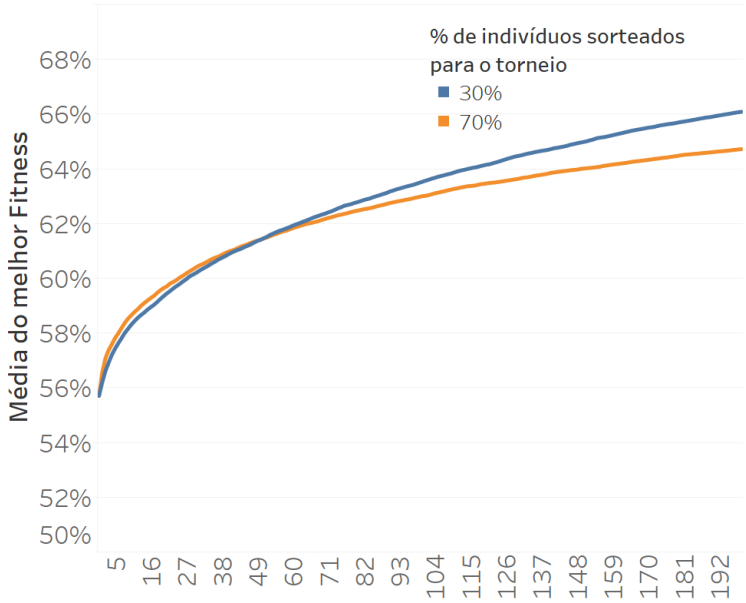

Figura 6: Média da melhor solução do algoritmo em relação ao percentual de indivíduos levados para o torneio.

Foi feita uma análise do desempenho de testes com diferentes percentuais de variáveis mutadas, na Figura 7. O algoritmo se desempenhou melhor quando uma maior quantidade de mutações ocorre nas suas variáveis.

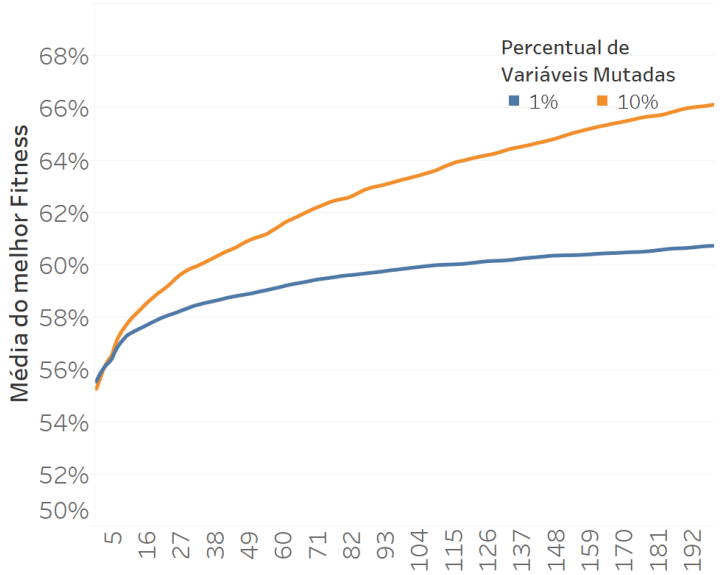

Figura 7: Média da melhor solução do algoritmo em relação ao percentual de variáveis mutadas.

CONCLUSÕES:

Neste trabalho foi possível aplicar técnicas de computação evolucionária em uma análise de sentimentos. Foi possível perceber melhorias no algoritmo para os diferentes parâmetros escolhidos, como por exemplo, a mutação que foi benéfica para a fitness do modelo.

O modelo proposto não conseguiu ao fim de suas iterações ter um bom desempenho. Este desempenho ruim pode ser explicado por diversas razões, como parâmetros inapropriados, métodos de mutação e cruzamento ineficientes, quantidade insuficiente de gerações ou a representação de cada indivíduo e do modelo foi inapropriada. Por se tratar de resultados parciais, acredita-se que com um melhor ajuste de parâmetros será possível obter melhores resultados.

ISNN: 2358-6052

Para trabalhos futuros, pode-se pensar em fazer uma maior quantidade de testes com diferentes combinações dos parâmetros para resolver os problemas. Ou então, é possível fazer uma maior quantidade de testes, uma vez que ao fim das iterações o melhor fitness ainda não havia se estabilizado em um ponto local. Pode-se também aplicar bibliotecas que aplicam sinônimos/hiperônimos/antônimos no modelo, conseguindo assim uma redução de features e mutações mais verossímeis, conforme feito por DRAGONI (2019).

REFERÊNCIAS BIBLIOGRÁFICAS:

BENEVENUTO, Fabrício; RIBEIRO, Filipe; ARAÚJO, Matheus. Métodos para análise de sentimentos em mídias sociais. In: **Brazilian Symposium on Multimedia and the Web (Webmedia), Manaus, Brasil**. 2015.

DRAGONI, Mauro. An Evolutionary Strategy for Concept-Based Multi-Domain Sentiment Analysis. **IEEE Computational Intelligence Magazine**, v. 14, n. 2, p. 18-27, 2019.

EIBEN, Agoston E.; SMITH, James E. What is an evolutionary algorithm?. In: **Introduction to Evolutionary Computing**. Springer, Berlin, Heidelberg, 2015. p. 25-48.

ERNAWATI, Siti et al. Implementation of The Naïve Bayes Algorithm with Feature Selection using Genetic Algorithm for Sentiment Review Analysis of Fashion Online Companies. In: **2018 6th International Conference on Cyber and IT Service Management (CITSM)**. IEEE, 2018. p. 1-5.

IQBAL, Farkhund et al. A Hybrid Framework for Sentiment Analysis Using Genetic Algorithm Based Feature Reduction. **IEEE Access**, v. 7, p. 14637-14652, 2019.

SIVANANDAM, S. N. S. N. Deepa. SN (2008) Introduction to Genetic Algorithms.