Detecção de Crises Sociais em Textos Utilizando o Algoritmo de Seleção Negativa

Lucas E. Fernandes[1], Matheus A. Ferraria[1], Samuel K. Gomes[1], Vinicius A. Ferraria[1], Leandro Nunes de Castro[1]

[1]Faculdade de Computação e Informática (FCI) – Universidade Presbiteriana Mackenzie (UPM)
Caixa Postal 01302-907 – São Paulo – SP – Brasil

{31823556, 31889824, 31817106, 31829643}@mackenzista.com.br, lnunes@mackenzie.br

Abstract. *With globalization and the growing popularity of social networks, information began to travel the planet at very high speeds, allowing a massive data generation that could be used for social studies. This research aims at utilizing Natural Computing, more specifically Artificial Immune Systems (AIS), through the Negative Selection Algorithm (NSA), implemented using the techniques of Bag of Words (BoW) and Linguistic Inquiry and Word Count (LIWC) to obtain relevant information that characterizes crisis in data from Twitter's social network. Through the results obtained at the end of this research it can be concluded that the use of AIS is a valid solution, though it is necessary a further finetuning of the models e techniques utilized.*

Resumo. *Com a globalização e a crescente popularização das redes sociais, a informação passou a trafegar pelo planeta de forma instantânea, abrindo portas para gerações massivas de dados que podem ser utilizados para estudos sociais. Essa pesquisa visa utilizar a Computação Natural, mais especificamente os Sistemas Imunológicos Artificiais (SIA) através do Algoritmo de Seleção Negativa (ASN), implementado utilizando as técnicas de Bag of Words (BoW) e Linguistic Inquiry and Word Count (LIWC), para obter informações relevantes que caracterizem crises em dados textuais. Os resultados obtidos permitiram concluir que a utilização de SIA é uma solução válida mediante o refinamento adequado de modelo e técnicas.*

1. Introdução

As redes sociais permitem a interação entre pessoas e empresas independentemente de sua localização, criando um ambiente com muita informação e unindo grupos com visões convergentes e divergentes [Crandall et al. 2013] [Imran et al. 2020].

A interação promovida pela tecnologia abriu um leque de possibilidades, trazendo uma quantidade estupenda de benefícios. Entretanto, junto com os benefícios surgem diversos problemas, uma vez que milhares de pessoas podem interagir através de posts e comentários. Um ponto de vista inadequado expresso por uma empresa ou indivíduo pode gerar uma grande crise com inúmeras consequências, tais como inferiorização de certo público, podendo até ser noticiada em meios de comunicação tradicionais como rádio e programas de televisão, amplificando ainda mais o seu alcance, uma vez que

esses meios servem de embasamento para diversas reações encontradas nas redes sociais [Nagy e Stamberger 2012].

A abundância de informação nas redes sociais pode ser ignorada e até considerada um problema para diversas pessoas. Porém, essa quantidade exorbitante de informações pouco estruturada pode ser utilizada para treinar modelos que permitam categorizar o tipo de informação, extrair conteúdos relevantes, como sentimentos, detectar a presença de crises, sejam elas sociais, naturais ou políticas, e entender o processo de suas formações. Com a possibilidade de entender a origem, detectar uma crise e analisar as reações causadas por ela, torna-se possível desenvolver estratégias preventivas [Nagy e Stamberger 2012] [Mukkamala 2015] [Imran et al. 2020].

A acelerada evolução das técnicas de *aprendizado de máquina* [Wason 2018] [de Castro e Ferrari 2016], *redes neurais* e *computação natural* [de Castro 2007], tornou possível a criação de modelos extremamente eficientes para a tarefa de detecção de crises. Apesar do termo Computação Natural não ter ganhado tanto destaque nas mídias, como Inteligência Artificial e Deep Learning, este termo vem crescendo no decorrer dos anos e permite o desenvolvimento de técnicas robustas que prometem solucionar problemas de alta complexidade [de Castro 2007].

A motivação de melhor entender o surgimento de crises, juntamente com o interesse em avaliar a utilização da computação natural neste meio social, deu origem a esta pesquisa. Esse projeto propõe a utilização de um sistema inspirado pela natureza, mais especificamente *Sistemas Imunológicos Artificiais*, e seus processos, como o mecanismo de *Seleção Negativa*, para detectar quando um dado de uma rede social é ofensivo e pode desencadear uma possível crise.

Por intermédio desse sistema imunológico artificial será possível encontrar e desenvolver estratégias mais eficientes para lidarmos com crises, detectar precocemente situações que possam se tornar eventos turbulentos e até mesmo maneiras de prevenir tais eventos.

2. Referencial Teórico

Para um melhor entendimento da pesquisa, dois conceitos principais devem ser revisados: *Detecção de crises em dados sociais* e *Sistemas Imunológicos Artificiais*. Essa seção faz uma breve revisão sobre esses dois temas e os demais conceitos utilizados na pesquisa.

2.1. Detecção de Crises em Dados Sociais

Uma crise pode ser definida como um processo de ruptura conjuntural ou estrutural no funcionamento e na organização de uma sociedade, podendo causar sérios impactos na performance de uma organização e gerar resultados negativos [Crandall et al. 2013].

A expansão da internet e a maior disponibilidade de aparelhos eletrônicos com acesso à internet abriram portas para um crescimento exponencial de usuários e as redes sociais se tornaram casas para grande parte desses usuários, que passaram a publicar parte de suas vidas nestas redes.

A abundância de informações disponíveis nas redes sociais e a popularidade do seu desenvolvimento se tornaram um fenômeno bastante atraente para muitos pesquisa-

dores. Pesquisas recentes mostram que a rede social Twitter possui grande importância para tomada de decisões em situações de crises [Shulz et al. 2013].

As crises em redes sociais podem ser caracterizadas como um aumento nas interações nestes canais [Castillo 2019]. Durante os eventos de uma crise, os usuários das redes tendem a engajar e reagir socialmente às notícias sobre processos de rupturas. A partir desses comportamentos seria possível utilizar essas massas de dados para detectar crises e aprender maneiras mais eficientes de controlá-las [Nagy e Stamberger 2012] [Mukkamala 2015] [Imran et al. 2020].

Ao mesmo tempo em que as redes sociais são consideradas dádivas da informação por muitos pesquisadores, é necessário considerar que todos os dados obtidos não podem ser prontamente utilizados, visto que estes dados não possuem um padrão e/ou sequer estão estruturados. Portanto, para extrair informações relevantes como, por exemplo, sentimento, contexto e semântica, é necessária a aplicação de diversos algoritmos de Processamento de Linguagem Natural (PLN).

Cabe ressaltar, entretanto, que a utilização de PLN para a extração de características de *tweets* apresenta resultados imperfeitos, uma vez que dados advindos de redes sociais apresentam limitações de caracteres, presença de gírias, abreviações e ambiguidades [Nguyen et al. 2016].

O surgimento de uma crise tem por consequência a geração de conteúdos em quantidades extremamente elevadas. Entretanto, nem todos os dados são favoráveis para a detecção de crises. Logo, para maximizar a quantidade de dados relevantes para o problema é necessária a aplicação de modelos de PLN capazes de identificar e analisar os contextos e semânticas desses dados de maneira eficaz [Saif e Alani 2017].

A utilização de dados oriundos do *Twitter* para detecção de crises apresenta obstáculos como o fato do *Twitter* não ser uma fonte centralizada de informações, visto que costuma ter ligações com outras fontes de dados como *sites* ou até mesmo outras redes sociais. Levando em consideração o escopo da descentralização dos dados e a quantidade exorbitante de dados presentes em redes sociais é possível concluir sanar este obstáculo não seria uma tarefa extremamente simples e poderia desviar o foco dos objetivos propostos por essa pesquisa, portanto, esse trabalho não levará em consideração o aspecto de centralização dos dados presentes nas redes sociais.

2.2. Sistemas Imunológicos Artificiais

A computação natural possui três subáreas: a computação inspirada na natureza; a simulação e emulação da natureza através da computação; e a computação utilizando materiais naturais [de Castro 2007].

Dentre tantos sistemas presentes na natureza cujas características tenham sido percebidas como relevantes no campo da Computação Natural, há um que ganhou destaque nas últimas décadas: os Sistemas Imunológicos Artificiais [Tarakanov e Dasgupta 2000] [de Castro e Timmis 2002].

Sistemas Imunológicos Artificiais (SIA) são um conjunto de algoritmos inspirados no sistema imunológico humano, tendo como objetivo solucionar problemas computacionais [Bendiab e Kholladi 2010].

A pesquisa e desenvolvimento desses sistemas busca aproveitar determinados fenômenos do sistema imune humano, como a detecção de anomalias no corpo, o padrão de reconhecimento de agentes patogênicos (a "memória" do sistema), a tolerância a erros, entre outras, a fim de criar algoritmos baseados em tais características [de Castro e Von Zuben 1999]. Os SIAs podem ser utilizados para desenvolver soluções em diversas áreas, especialmente no campo da Inteligência Artificial.

Um dos mecanismos de defesa mais utilizados em Sistemas Imunológicos Artificiais é a *Seleção Negativa*, cujo funcionamento é inspirado em um processo de reconhecimento e seleção de *linfócitos* T pelo *timo* (glândula linfoide do sistema imunológico humano). Na realidade, este processo consiste em eliminar linfócitos T que reconhecem células pertencentes ao corpo, deixando somente os linfócitos que reconhecem células estrangeiras [González e Dasgupta 2003].

A inspiração natural para o desenvolvimento do *Algoritmo de Seleção Negativa* em Sistemas Imunológicos Artificiais não inibe a existência de problemas de acurácia deste mecanismo. Um dos problemas mais conhecidos deste algoritmo é a presença de conjuntos incompletos do próprio que resulta no amadurecimento de linfócitos que não foram expostos à todas as proteínas presentes no corpo e, consequentemente, estes linfócitos acabam reconhecendo alguma célula do corpo ocasionando uma reação autoimune [Hofmeyr e Forrest 1999].

Diferentemente dos sistemas imunológicos humanos, os SIA estão sujeitos a problemas de desempenho causados pela complexidade e dinâmica necessárias para replicar mecanismos de defesa. O desempenho do sistema está diretamente relacionado ao tamanho do conjunto das células próprias do sistema, uma vez que este tamanho afeta de maneira exponencial o tempo necessário para gerar os detectores do sistema [Hofmeyr e Forrest 1999].

2.3. Análise de Textos

A Análise de Textos refere-se ao processo de extrair informações significativas de textos através de sua análise semântica, entretanto, um computador não é capaz de compreender textos, desta maneira é necessário estabelecer uma interface entre a linguagem dos computadores e a linguagem humana obtida através de representações numéricas computáveis [Figueiredo e Coelho 2020].

O processo de Análise de Textos pertence à área de Processamento de Linguagem Natural (NLP), uma subárea da Computação Natural, e tem como finalidade estudar maneiras de modelar a linguagem humana para representações computacionais [Joshi 1991], permitindo assim que computadores sejam capazes de compreender os textos a serem analisados, garantindo assim resultados mais significativos que podem ser utilizados para outras finalidades como a detecção de crises.

2.3.1. *Bag of Words*

Bag of words (BoW) é um modelo simples e elegante [Zhang et al. 2010] bastante utilizado no Processamento de Linguagem Natural. Essa técnica consiste na criação de um dicionário a partir das sentenças utilizadas como entrada. Para obter um melhor resultado o *BoW* geralmente é utilizado em conjunto com técnicas de pré-processamento para se remover palavras sem muito significado, como *stopwords*, e padronizando os dados

de entrada através da remoção de caracteres especiais e mantendo todas as palavras em maiúsculo ou minúsculo.

2.3.2. TF-IDF

O *Term Frequency Inverse Document Frequency* (TF-IDF) é uma medida estatística com a finalidade de determinar a importância (peso) de cada palavra nos documentos analisados. Esta medida é calculada usando a frequência relativa de cada palavra no documento analisada em relação ao inverso da quantidade de documentos que apresentam a palavra sendo avaliada [Ramos 2003].

Dessa forma, quanto maior o valor do *TF-IDF*, maior a relevância de uma palavra nos documentos analisados [Ramos 2003].

2.3.3. LIWC

O *Linguistic Inquiry and Word Count* (LIWC) [Pennebaker et al. 2001] é uma ferramenta de análise de texto que, a partir de um dicionário escolhido pelo usuário, realiza a classificação de palavras em diferentes categorias.

Essa ferramenta tem o intuito de analisar e extrair características linguísticas de um documento independentemente do seu contexto. A partir de um texto de entrada, o *LIW*C calcula os valores de cada categoria através da divisão das frequências de cada palavra pelo total de palavras da sentença analisada.

Existem diversos trabalhos que utilizam o *LIWC* para a análise e extração de sentimentos. Em alguns dos trabalhos o dicionário do *LIWC* é até traduzido para outro idioma [Filho et al. 2013].

2.3.4. Similaridade de Cossenos

A *similaridade de cossenos* é um cálculo utilizado para se medir a distância entre dois vetores, onde quanto maior seu resultado maior será a semelhança entre os vetores. A medida de cosseno é bastante eficiente e utilizada em projetos quando se necessita comparar vetores com muitas dimensões [Rahutomo et al. 2012], tornando-se uma medida extremamente importante para a análise de textos, uma vez que permite mapear vetores de sentença e depois compará-los [Li e Han 2013]

2.3.5. Distância Euclidiana

A *Distância Euclidiana* é o cálculo da distância entre dois pontos, podendo ser utilizada tanto no processamento de imagens [Wang et al. 2005], quanto no estudo da sintaxe de determinadas frases [Cancho 2004].

Diferentemente da *similaridade dos cossenos*, a utilização da *Distância Euclidiana* para a geração de detectores no presente trabalho busca a menor distância, visto que a semelhança dos pontos é inversamente proporcional à distância encontrada.

2.3.6. *Crossover*

Crossover é um operador genético utilizado em Algoritmos Genéticos na etapa de reprodução, tendo como principal finalidade a geração de soluções “filhas” a partir de soluções “pais” existentes de forma estocástica [Nazif e Lee 2012].

A geração das soluções dá-se da seguinte forma: a partir de uma população de soluções, um par de soluções "pais" é selecionado e é realizada uma recombinação das suas características, gerando uma solução "filha" com diversas características semelhantes às dos "pais". Este processo então é repetido até que se alcance uma condição de parada determinada pelo algoritmo [Kumar et al. 2010].

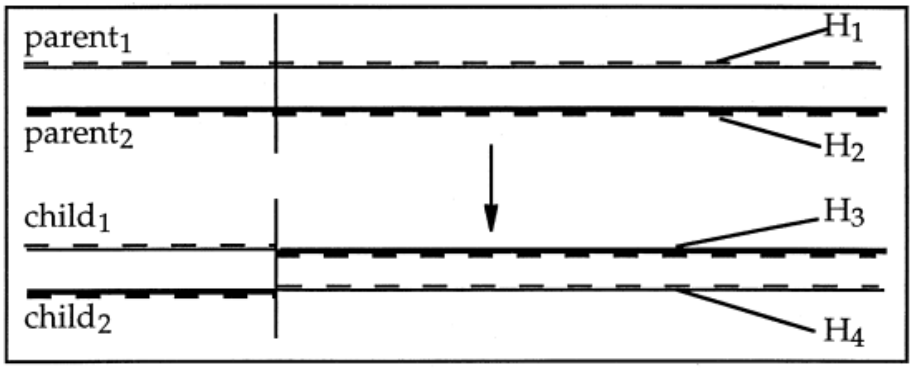

Figura 1. *Crossover*

Fonte: Vrajitoru 1998

3. Modelo Proposto

3.1. Algoritmo de Seleção Negativa

O Algoritmo de Seleção Negativa (ASN) é fundamental para o funcionamento do Sistema Imunológico Artificial (SIA), uma vez que ele é responsável por distinguir entre células do anfitrião e células externas [de Castro e Zuben 1999].

O ASN consiste em diversos detectores, responsáveis por encontrar e identificar células externas, e são divididos em três categorias: imaturos, maduros e memória.

Os detectores imaturos são considerados o estado inicial de um detector e passam por uma fase inicial de testagem, onde todos os detectores que são capazes de detectar o anfitrião são eliminados, ou seja, somente são selecionados os detectores que não identificam o anfitrião, dando assim o nome do algoritmo [Hofmeyr e Forrest 1999].

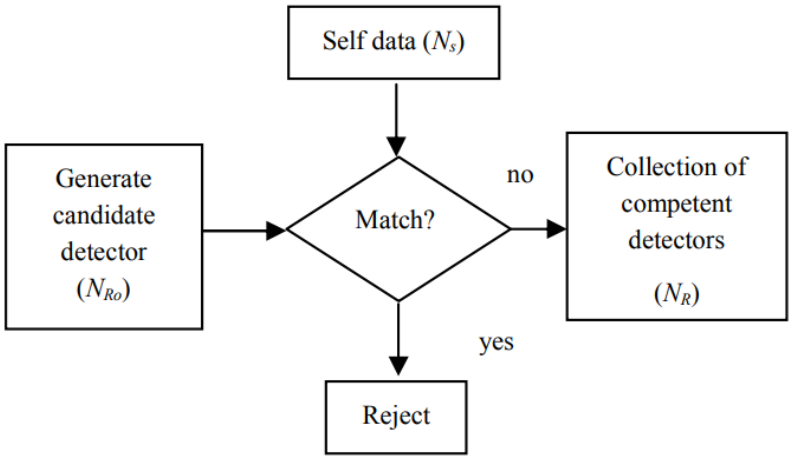

Figura 2. Fluxograma do ASN

Fonte: Ayara et al. 2002

Todos os detectores que sobrevivem à fase de testagem se tornam maduros, e passam por um processo de aprendizagem, onde, caso um detector reconheça quantidades elevadas de células invasoras, tais detectores são transformados em detectores de memória, no qual, diferentemente das outras categorias, seu tempo de vida é maior e possuem um processo mais agressivo para as células estrangeiras [Hofmeyr e Forrest 1999].

Apesar do processo de testagem, é possível que detectores capazes de reconhecer células próprias não sejam eliminados e sejam liberados para percorrer o ambiente, consequentemente podendo gerar reações autoimunes, nas quais o SIA passa a atacar suas próprias células [De Castro e Zuben 1999].

Uma ressalva da utilização do ASN é o custo computacional necessário para a execução desse algoritmo, visto que a geração e eliminação constante de detectores imaturos tende a consumir muitos recursos, tratando-se de um processo bastante longo [Ayara et al. 2002].

Para aumentar a eficácia dos detectores presentes no ASN é necessário replicar uma característica importante dos sistemas imunológicos: sua diversidade. Pesquisas da área comprovam a eficácia da aplicação de mutações nos detectores, permitindo assim que a população de detectores seja extremamente diversa e possua maior probabilidade de identificar células invasoras [Hofmeyr e Forrest 1999] [Ayara et al. 2002].

4. Metodologia

Este trabalho propõe uma pesquisa de natureza qualitativa que tem como finalidade avaliar a eficácia da utilização de um Sistema Imunológico Artificial usando o *Algoritmo de Seleção Negativa* (ASN) para detectar crises em dados textuais. No cenário desta pesquisa o *Bag of Words* foi utilizado para formar o *dBow*, uma matriz de pesos cujas colunas são formadas por todas as palavras presentes no *BoW* e seus valores refletem o cálculo do IDF para cada sentença de entrada, e suas linhas representam as sentenças utilizadas como entrada. Além da utilização do *BoW* também será utilizado o LIWC e,

posteriormente, os resultados dos dois modelos serão comparados com a finalidade de avaliar qual modelo se apresenta mais eficaz no contexto deste trabalho.

5. Avaliação de Desempenho

5.1. Metodologia Experimental

Para a realização dos experimentos foi utilizada a base de dados do *IMDB* [UCI Archive], formada por duas dimensões: os textos contendo as avaliações realizadas por usuários e suas respectivas classificações. Os dados presentes na base só podem ser classificados como positivos, representados pelo valor 0, e negativos representados pelo valor 1. Essa base de dados contém apenas 1000 objetos, sendo 500 positivos e 500 negativos, e apesar de seus dados não estarem relacionados a crises, eles foram utilizados devido à sua classificação binária, permitindo assim a divisão em *próprios* e *não-próprios*, grupos importantes para o funcionamento do ASN.

O processo de pesquisa é composto pela implementação do ASN utilizando a linguagem de programação *Python 3.8.5*, com o auxílio das bibliotecas: *numpy*, *re*, *pandas*, *multiprocessing*, *collections*, *typing*, *seaborn*, *matplotlib*, *spacy*, *LIWC-Python* [LIWC-Python - Repository]. Todos os experimentos foram realizados no ambiente virtual do *Python* em uma máquina com Windows 11 e com a CPU I7-10550U *quad-core*.

Para fazer a integração da funcionalidade do *LIWC-Python* foi necessário obter uma licença temporária da ferramenta LIWC [LIWC2015] e realizar a extração do dicionário proprietário utilizado internamente pela ferramenta.

Após a extração do dicionário foram realizadas comparações entre as saídas obtidas pela ferramenta do LIWC [LIWC2015] e pelo código *Python*, em que se notou uma variação considerável de 2%-5% em algumas categorias decorrentes das diferenças no pré-processamento dos softwares.

Os experimentos foram compostos pela utilização de 500 objetos classificados como *próprios* no treinamento, ou seja, a geração de candidatos a detectores, e a acurácia do modelo foi calculada pela classificação da base em sua totalidade com 1000 sentenças.

Para se ter uma melhor noção do comportamento do modelo foram realizados diversos experimentos variando os seguintes parâmetros: número de detectores gerados pelo sistema, o limiar utilizado para comparação de detectores durante a fase de treinamento e um novo valor do limiar utilizado na etapa de classificação.

Em cada variação de parâmetro foram realizados dois testes: um utilizando a técnica do *BoW* e outro utilizando a técnica do LIWC.

5.1.1. Número de Detectores

O número de detectores é responsável por indicar quantos detectores o sistema deve gerar para posteriormente realizar a classificação de novos dados. A variação na quantidade de detectores foi utilizada para se tentar cobrir diferentes espaços na detecção e estudar como esse parâmetro poderia impactar a acurácia do modelo.

5.1.2. Candidatos a Detectores

5.1.2.1. Geração Aleatória

Os candidatos a detectores serão vetores gerados de forma aleatória, onde todos seus valores estarão entre 0 e 1. Para o modelo proposto com o *BoW*, um candidato será representado por um vetor com o tamanho do dicionário interno, ou seja, a quantidade de palavras distintas encontradas pelo modelo. No caso do modelo proposto com o LIWC, cada candidato a detector terá o tamanho equivalente à quantidade de categorias presentes no dicionário utilizado. Para ambos os modelos os candidatos a detectores só serão considerados válidos caso não reconheçam as sentenças classificadas como *próprias* durante a fase de treinamento.

5.1.2.2. Geração por *Crossover*

Com a finalidade de reaproveitar os dados que contêm crises foi implementada a geração de candidatos a detectores através da técnica de *crossover* de *k-pontos*. Para essa geração são escolhidos, de maneira aleatória, duas linhas da matriz *dBow*, uma delas classificada como crise e outra como não-crise. Após a seleção essas linhas são combinadas usando o *crossover*; a combinação ocorre a partir de uma quantidade *k* de cortes. O parâmetro *k* é inserido na inicialização do modelo e precisa ser inferior à quantidade de colunas presentes na matriz, caso nenhum valor seja definido o valor padrão utilizado será 15% da quantidade de colunas do *dBow*.

5.1.3. Limiar de Treinamento

O limiar de treinamento é um valor que ajuda a identificar quando um candidato detector deve ser descartado. Este parâmetro define o limite de semelhança para que um *match* ocorra, ou seja, todos valores abaixo deste limiar são considerados como diferentes. Durante a fase de treinamento esse valor foi utilizado para efetuar as comparações entre candidatos a detectores e elementos *próprios*, caso o resultado da comparação seja maior que o limiar, o candidato será descartado.

5.1.4. Limiar de Classificação

Durante os experimentos realizados foi possível perceber que a utilização do mesmo valor de limiar de treinamento para o processo de classificação resultava numa elevada taxa de Falso Negativo. Para sanar este problema foram utilizados limiares mais baixos para o processo de classificação, permitindo assim que o modelo realizasse classificações de maneira mais eficiente.

5.2. Resultados e Discussão

5.2.1. Resultados Preliminares

Visando avaliar a metodologia inicialmente proposta neste trabalho, utilização do *BoW* em conjunto com o TF-IDF e similaridade de cosseno, foram realizados 15 testes variando os seguintes parâmetros: número de detectores e limiares. Para todos os testes foi utilizada a mesma base composta por 1000 textos, sendo ela formada por 500 sentenças *próprias* e 500 *não-próprias*.

Tabela 1. Resultados obtidos nos experimentos preliminares utilizando o modelo do *BoW*.

Número de	Detectores	Limiar de	Acurácia	Precisão	*Recall*

Sentenças		Treinamento/Classificação			
1000	1.000	65%	50,00%	0,00%	0,00%
		75%	50,00%	0,00%	0,00%
		85%	49,80%	0,00%	0,00%
	2.500	65%	50,00%	0,00%	0,00%
		75%	50,00%	0,00%	0,00%
		85%	50,00%	0,00%	0,00%
	5.000	65%	50,00%	0,00%	0,00%
		75%	50,00%	0,00%	0,00%
		85%	50,00%	0,00%	0,00%
	7.500	65%	50,00%	0,00%	0,00%
		75%	50,00%	0,00%	0,00%
		85%	50,00%	0,00%	0,00%
	10.000	65%	50,00%	0,00%	0,00%
		75%	50,00%	0,00%	0,00%
		85%	50,00%	0,00%	0,00%
	12.500	65%	50,00%	0,00%	0,00%
		75%	50,00%	0,00%	0,00%
		85%	50,00%	0,00%	0,00%

A partir dos resultados obtidos foi possível identificar a necessidade de aprimorar o processo de detecção do modelo em decorrência da baixa taxa de classificação, um resultado misto, visto que a não detecção de *próprios* ocorreu da maneira esperada, sendo representada pela acurácia de 50%, porém a detecção de *não-próprios* simplesmente não estava acontecendo.

Em decorrência dos resultados obtidos foi possível determinar que valores muito altos de limiares para a fase de detecção podem causar uma dificuldade para o modelo, pois ele se tornava extremamente rígido para identificar *não-próprios* e levava ao não reconhecimento de características de possíveis crises que os modelos encontrariam normalmente nas redes sociais.

Outro aspecto identificado durante a execução desses testes preliminares foi um grande processamento computacional sendo gasto para se montar o *dBoW* e posteriormente realizar as comparações necessárias para a geração dos detectores. Este processamento é decorrente da quantidade de categorias presentes no *BoW*, onde cada categoria equivale à uma palavra de seu vocabulário.

Apesar da remoção de *stopwords* e tratamento de caracteres, a quantidade de categorias (*tokens*) se mostrou bastante elevada, o que, consequentemente, poderia gerar futuros impactos computacionais. A partir deste grande número de categorias foi proposta a utilização do modelo LIWC que apresenta uma quantidade fixa de categorias para quaisquer documentos, garantindo assim melhor capacidade de análise léxica.

Além da implementação do LIWC para comparação com o *BoW* também serão implementadas uma nova técnica para a geração de candidatos a detectores e uma nova medida de distância.

Para a técnica de geração de candidatos a detectores será implementado o *crossover* de múltiplos pontos para se tentar sanar o problema de aleatoriedade e tentar representar o *não-próprio* de forma mais verossímil. A técnica consiste em combinar dados *próprios* e *não-próprios* para se gerar melhores candidatos, utilizando assim de maneira mais eficiente os dados *não-próprios* contidos no conjunto treinamento.

A implementação da nova medida de distância como função de comparação tem como finalidade avaliar se a medida de detecção utilizada atualmente é de fato a mais adequada para ativar os detectores.

5.2.2. Resultados Finais

Levando em consideração as mudanças propostas na seção anterior o modelo foi atualizado com a finalidade de contemplá-las. Após a atualização do modelo, foram definidos novos casos de testes a serem realizados para avaliar sua acurácia.

Com a finalidade de realizar testes que condizem melhor com a realidade, as sentenças utilizadas pelo modelo foram alteradas. Nos novos testes a base de 1000 sentenças foi dividida em dois subgrupos: grupo de teste e grupo de treinamento. Ambos os grupos apresentam a mesma quantidade de textos, porém, são compostos de maneiras diferentes.

O grupo de treinamento apresenta 400 sentenças *próprias* e 100 sentenças *não-próprias*. Essas 100 sentenças serão utilizadas para o processo de *crossover* e para garantir que o vocabulário gerado pelo *BoW* não possua um viés próprio, este grupo será utilizado para treinar o modelo.

Já o grupo de testes é constituído pelas 500 sentenças restantes da base original, desta maneira esse grupo apresenta 400 sentenças *não-próprias* e 100 sentenças próprias e será utilizada no processo de detecção do modelo para avaliar sua acurácia.

A abordagem de dividir as sentenças em dois grupos é de suma importância, uma vez que ela não só permite o modelo aproveitar os dados *não-próprios* durante o seu treinamento, mas também permite avaliar o desempenho do modelo para identificar dados *próprios*. Além disso a exposição à dados não conhecidos no processo de teste permite uma representação mais fiel do comportamento do modelo.

Para tornar os testes mais centrados nas novas técnicas implementadas, os números de detectores foram fixados em 10.000 e 12.500.

Utilizando os resultados preliminares como base, o parâmetro do limiar foi diferenciado em dois novos parâmetros: o limiar de treinamento e o limiar de classificação. Essa diferenciação do limiar permite a utilização de valores menores durante a classificação sem afetar a etapa de sensoriamento, na qual os detectores são treinados a não reconhecer dados *próprios*.

Após a implementação de todas as técnicas propostas na seção anterior os parâmetros do algoritmo passaram a ser: modelo utilizado (*BoW* ou LIWC), método de geração de candidatos a detectores (Aleatório e *Crossover*), método de detecção (Distância Euclidiana e Similaridade de Cosseno), número de detectores, limiar de treinamento e limiar de classificação.

Para o limiar de treinamento foram utilizados os valores 65%, 75% e 85%, enquanto para o limiar de classificação foram utilizados 5%, 15%, 25% e 35%. A utiliza-

ção dos valores mencionados anteriormente resultou em uma combinação de 12 limiares a serem testados para cada parâmetro.

Além da divisão da base e da introdução de novos parâmetros os resultados apresentados abaixo correspondem à média após 5 execuções de cada experimento. Devido à elevada quantidade de parâmetros e combinações, os resultados abaixo só irão contemplar os melhores resultados obtidos para cada parâmetro

Os resultados serão divididos mencionados acima serão apresentados nas próximas seções.

5.2.2.1. Resultados LIWC

Visando avaliar o desempenho do LIWC utilizando não só as técnicas inicialmente previstas na metodologia como a Similaridade de Cosseno e a geração aleatória de Candidatos a Detectores, também foram realizados testes utilizando a Distância Euclidiana como nova técnica de comparação e o *Crossover* em conjunto com os novos parâmetros de limiares.

Tabela 2. Resultados obtidos após 5 execuções do experimento utilizando o modelo LIWC e o *Crossover para geração dos Candidatos a Detectores.*

				Acurácia		**Precisão**		***Recall***	
Método de Detecção	**Número de Detectores**	**Limiares de Treinamento**	**Limiares de Classificação**	**Média Obtida**	**Desvio Padrão**	**Média Obtida**	**Desvio Padrão**	**Média Obtida**	**Desvio Padrão**
Similaridade de Cossenos	10.000	65%	5%	79,80 %	0,00%	79,96%	0,00%	99,75 %	0,00%
		75%	5%	79,80 %	0,00%	79,96%	0,00%	99,75 %	0,00%
		85%	5%	79,80 %	0,00%	79,96%	0,00%	99,75 %	0,00%
	12.500	65%	5%	79,80 %	0,00%	79,96%	0,00%	99,75 %	0,00%
		75%	5%	79,80 %	0,00%	79,96%	0,00%	99,75 %	0,00%
		85%	5%	79,80 %	0,00%	79,96%	0,00%	99,75 %	0,00%
Distância Euclidiana	10.000	65%	35%	20,20 %	0,00%	100,00 %	0,00%	0,25%	0,00%
		75%	25%	20,20 %	0,00%	100,00 %	0,00%	0,25%	0,00%
		85%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0,00%
	12.500	65%	25%	20,20 %	0,00%	100,00 %	0,00%	0,25%	0,00%
		75%	25%	20,20 %	0,00%	100,00 %	0,00%	0,25%	0,00%
		85%	5%	20,00 %	0,00%	0,00%	0,00%	0,0%	0,00%

Tabela 3. Resultados obtidos após 5 execuções do experimento utilizando o modelo LIWC e geração aleatória de Candidatos a Detectores.

				Acurácia		Precisão		*Recall*	
Método de Detecção	**Número de Detectores**	**Limiares de Treinamento**	**Limiares de Classificação**	**Média Obtida**	**Desvio Padrão**	**Média Obtida**	**Desvio Padrão**	**Média Obtida**	**Desvio Padrão**
Similaridade de Cossenos	10.000	65%	5%	79,80 %	0,00%	79,96 %	0,00%	99,75 %	0,00%
		75%	5%	79,80 %	0,00%	79,96 %	0,00%	99,75 %	0,00%
		85%	5%	79,80 %	0,00%	79,96 %	0,00%	99,75 %	0,00%
	12.500	65%	5%	79,80 %	0,00%	79,96 %	0,00%	99,75 %	0,00%
		75%	5%	79,80 %	0,00%	79,96 %	0,00%	99,75 %	0,00%
		85%	5%	79,80 %	0,00%	79,96 %	0,00%	99,75 %	0,00%
Distância Euclidiana	10.000	65%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0,00%
		75%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0,00%
		85%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0,00%
	12.500	65%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0,00%
		75%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0,00%
		85%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0,00%

Observando os resultados apresentados acima (Tabela 2 e Tabela 3) se torna evidente que o LIWC apresentou melhores resultados para a classificação de não crises, como é possível perceber através da maior taxa de acurácia, precisão e *recall*. Um ponto a se observar é que melhores resultados obtidos foram usando limiares de classificação extremamente baixos.

Os resultados obtidos permitem observar que a Distância Euliciana não apresentou bons resultados para a classificação. Sua precisão se manteve em 0%, indicando que o modelo sequer chegou a realizar classificação, ou seja, todas suas entradas foram classificadas como *próprio* e devido à base utilizada para teste estar dividida em 20% (100 senteças) *próprios* e 80% *não-próprios* a acurácia apresenta para este método de detecção apenas refletiu a divisão da base.

Devido ao resultados da Distância Euclidiana não apresentarem valores significativos para as análises abaixo será considerado apenas os resultados obtidos utilizando a Similaridade de Cossenos.

Um aspecto interessante analisado nos resultados é que todos os melhores casos de classificação apresentaram a mesma acurácia, precisão e *recall* indicando que os

métodos diferentes de geração de detectores não apresentaram diferenças no modelo LIWC. Além dos resultados semelhantes todos testes também compartilhavam seu melhor resultado utilizando o limiar de classificação mais baixo (*5%*), o baixo valor do limiar implicando em detecções equivocadas de *próprios*, conforme é possível perceber através do valor de *79.96%* de precisão, representando que os 20% *próprios* presentes na base de teste foram identificados como *não-próprios*.

5.2.2.2. Resultados *BoW*

Com a finalidade de tornar os resultados comparáveis os testes utilizando o modelo do *BoW* seguiram o mesmo princípio utilizado nos testes do LIWC, ou seja, utilizando mesma variação de parâmetros e bases de teste e treinamento.

Tabela 4. Resultados obtidos após 5 execuções do experimento utilizando o modelo *BoW* e o *Crossover* para a geração de Candidatos a Detectores.

				Acurácia		**Precisão**		***Recall***	
Método de Detecção	**Número de Detectores**	**Limiares de Treinamento**	**Limiares de Classificação**	**Média Obtida**	**Desvio Padrão**	**Média Obtida**	**Desvio Padrão**	**Média Obtida**	**Desvio Padrão**
Similaridade de Cossenos	10.000	65%	5%	75,60 %	0,00%	79,700 %	0,00%	93,25 %	0,00%
		75%	5%	76,20 %	0,00%	79,83%	0,00%	94,00 %	0,00%
		85%	5%	76,20 %	0,00%	79,83%	0,00%	94,00 %	0,00%
	12.500	65%	5%	76,20 %	0,00%	79,83%	0,00%	94,00 %	0,00%
		75%	5%	76,40 %	0,00%	79,87%	0,00%	94,25 %	0,00%
		85%	5%	76,40 %	0,00%	79,87%	0,00%	94,25 %	0,00%
Distância Euclidiana	10.000	65%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0,00%
		75%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0.00%
		85%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0,00%
	12.000	65%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0,00%
		75%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0,00%
		85%	5%	20,00 %	0,00%	0,00%	0,00%	0,00%	0,00%

Tabela 5. Resultados obtidos após 5 execuções do experimento utilizando o modelo *BoW e geração aleatória de Candidatos a Detectores.*

				Acurácia		**Precisão**		***Recall***	
Método de Detecção	**Número de**	**Limiares de Treinament**	**Limiares de Classificaçã**	**Média Obtid**	**Desvio Padrã**	**Média Obtid**	**Desvio Padrã**	**Médid a**	**Desvio Padrã**

	Detectores	o	o	a	o	a	o	Obtida	o
Similaridade de Cossenos	10.000	65%	5%	67,00%	0,00%	79,16%	0,00%	79,75%	0,00%
		75%	5%	67,00%	0,00%	79,16%	0,00%	79,75%	0,00%
		85%	5%	67,00%	0,00%	79,16%	0,00%	79,75%	0,00%
	12.500	65%	5%	67,00%	0,00%	79,16%	0,00%	79,75%	0,00%
		75%	5%	67,00%	0,00%	79,16%	0,00%	79,75%	0,00%
		85%	5%	67,00%	0,00%	79,16%	0,00%	79,75%	0,00%
Distância Euclidiana	10.000	65%	5%	20,00%	0,00%	0,00%	0,00%	0,00%	0,00%
		75%	5%	20,00%	0,00%	0,00%	0,00%	0,00%	0,00%
		85%	5%	20,00%	0,00%	0,00%	0,00%	0,00%	0,00%
	12.500	65%	5%	20,00%	0,00%	0,00%	0,00%	0,00%	0,00%
		75%	5%	20,00%	0,00%	0,00%	0,00%	0,00%	0,00%
		85%	5%	20,00%	0,00%	0,00%	0,00%	0,00%	0,00%

A partir dos resultados representados acima (Tabela 4 e Tabela 5) foi possível compreender que assim como o modelo do LIWC o método de detecção através da Distância Euclidiana apresentou resultados muito abaixo do esperado, portanto, todas as conclusões abaixo serão voltadas apenas para os resultados obtidos utilizando a Similaridade de Cosseno como método de detecção.

Levando em consideração os diferentes métodos de geração dos Candidatos a Detectores é possível concluir que o *Crossover* apresentou resultados menores, apesar da diferença entres o valor de acurácia e precisão não serem tão muito significativas, quanto os valores de *recall*, os resultados em si apresentaram ganhos consideráveis sobre a geração aleatória.

6. Conclusões e Perspectivas Futuras

Com os resultados gerados ao final da pesquisa foi possível verificar que a metodologia escolhida precisaria ser ainda mais refinada, porém a utilização da computação natural, mais especificamente o Algoritmo de Seleção Negativa, para a detecção de crises se apresentou como uma técnica bastante interessante para a solução do problema proposto por esse trabalho.

A partir dos resultados obtidos se tornou evidente a necessidade de aprimorar tanto a técnica de geração de Candidatos a Detectores quanto a técnica de detecção utilizada, além da realização de novos testes contemplando novos valores de limiares para

eliminar o comportamento autoagressivo identificado, ou seja, a detecção equivocada de *próprios*.

O estudo e aprimoramento da geração de Candidatos a Detectores é de suma importância para aprimorar os resultados, obtidos uma vez que os processos utilizados atualmente, apesar de apresentarem abordagens diferentes, apresentam a mesma falha, ou seja, não é inteligente o suficiente para identificar se o detector gerado já está presente no conjunto de detectores gerados, criando uma redundância na cobertura do modelo.

O trabalho menciona a utilização da rede social *Twitter* como fonte de dados para o modelo, porém devido à acurácia, precisão e *recall* não atingirem valores satisfatórios, acima de 80%, os *tweets* acabaram não sendo utilizados como fonte de dados, portanto, a utilização de dados oriundos do *Twitter* se tornou uma dependência do aprimoramento do modelo proposto nesse trabalho.

Referências

Ayara, M., Timmis, J., de Lemos, R., de Castro, Leandro N. e Duncan, R. (2002). *Negative selection: How to generate detectors. In Proceedings of the 1st International Conference on Artificial Immune Systems (ICARIS)* (Vol. 1, pp. 89-98). University of Kent at Canterbury Printing Unit University of Kent at Canterbury.

Bendiab, E. e Kholladi, M. K. (2010). *The Negative Selection Algorithm: a Supervised Learning Approach for Skin Detection and Classification*. IJCSNS International Journal of Computer Science and Network Security. 10. 86-92.

Burel G., Saif, H. e Alani, H. (2017). *Semantic Wide and Deep Learning for Detecting Crisis-Information Categories on Social Media*. In: d'Amato C. et al. (eds) The Semantic Web – ISWC 2017. ISWC 2017. Lecture Notes in Computer Science, vol 10587.

Castillo, C. (2019). *Big Crisis Data: Social Media in Disasters and Time-Critical Situations*.

Cancho, R. F. (2004). Euclidean distance between syntactically linked words. Physical Review E, 70(5), 056135.

de Castro, Leandro N. (2007). *Fundamentals of natural computing: an overview*. Physics of Life Reviews, Volume 4, Issue 1, pp 1-36.

de Castro, Leandro N. e Ferrari, Daniel G. (2016). *Introdução à Mineração de Dados: Conceitos básicos, algoritmos e aplicações*. Editora Saraiva, Edição 1.

de Castro, Leandro N. e Von Zuben, Fernando J. (1999). *Artificial immune systems: Part I–basic theory and applications*. Universidade Estadual de Campinas, dezembro, Tech. Rep 210.

Figueiredo, L. e Coelho, O. (2020) "Uma Ferramenta de Visualização dos Principais Tópicos de Artigos Científicos Baseada em Representações Vetoriais de Palavras Geradas por Meio de Deep Learning", Jornada de Iniciação Científica e Mostra de Iniciação Tecnológica - ISSN 2526-4699.

Filho, P. B., Pardo, T. A. S. e Aluisio, S. (2013). *An evaluation of the Brazilian Portuguese LIWC dictionary for Sentiment Analysis*. Proceedings of the 9th Brazilian Symposium in Information and Human Language Technology.

González, F. A. e Dasgupta, D. (2003). *Anomaly Detection Using Real-Valued Negative Selection*. Genet Program Evolvable Mach 4, 383–403.

Greensmith, J., Whitbrook A. e Aickelin U. (2010). *Artificial Immune Systems*. In: Gendreau M., Potvin JY. (eds) Handbook of Metaheuristics. International Series in Operations Research & Management Science, vol 146. Springer, Boston, MA.

Hofmeyr, S. A. e Forrest, S. (1999). *Immunity by Design: An Artificial Immune System*.

Joshi, A. J. (1991). Natural Language Processing. New Series, Vol 253, No. 5025, pp. 1242-1249.

Imran, A. S., Daudpota, S. M., Kastrati, Z. e Batra, R. (2020). *Cross-Cultural Polarity and Emotion Detection Using Sentiment Analysis and Deep Learning on COVID-19 Related Tweets*. IEEE Access, *8*, 181074-181090.

Kumar, M., Husain, M., Upreti, N. e Gupta, D. (2010). Genetic algorithm: Review and application. Available at SSRN 3529843.

Li, B. e Han, L. (2013). *Distance weighted cosine similarity measure for text classification*. International conference on intelligent data engineering and automated learning, 611-618, Springer, Berlin, Heidelberg.

LIWC-Python – Repository. https://github.com/chbrown/liwc-python. Acessado em 5 de novembro de 2021.

LIWC2015. http://liwc.wpengine.com. Acessado em 1 de novembro de 2021.

Mukkamala, R.R., Sørensen, J.I., Hussain, A. e Vatrapu, R. (2015). *Social set analysis of corporate social media crises on facebook*. In 2015 IEEE 19th International Enterprise Distributed Object Computing Conference (pp. 112-121). IEEE.

Nagy, A. e Stamberger, J.A. (2012). *Crowd sentiment detection during disasters and crises*. ISCRAM.

Nazif, H. e Lee, L. S. (2012). Optimised crossover genetic algorithm for capacitated vehicle routing problem. Applied Mathematical Modelling, 36(5), 2110-2117.

R. Crandall, W., A. Parnell, J. e E. Spillan. J. (2013). *Crisis Management: Leading in the New Strategy Landscape*. SAGE Publications, Incorporated.

Rahutomo, F., Kitasuka, T. e Aritsugi, M. (2012). *Semantic cosine similarity*. The 7th International Student Conference on Advanced Science and Technology ICAST, Vol. 4, No. 1, p. 1.

Ramos. J. (2003). *Using TF-IDF to Determine Word Relevance in Document Queries*. Department of Computer Science, Rutgers University, 23515 BPO Way, Piscataway, NJ, 08855.

Shulz, A., Thanh, T. D., Paulheim, H. e Scheweizer, I. (2013). *A Fine-Grained Sentiment Analysis Approach for Detecting Crisis Related Microposts*.

T. Nguyen, D., A. A. Mannai, K., Joty, S., Sajjad, H., Imran, M. e Mitra, P. (2016). *Rapid Classification of Crisis-Related Data on Social Networks using Convolutional Neural Networks*. Qatar Computing Research Institute – HBKU, Qatar.

Tarakanov, A. e Dasgupta, D. (2000). *A formal model of an artificial immune system*. BioSystems, *55*(1-3), 151-158.

Timmis, J., Knight, T., de Castro, Leandro N. e Hart, E. (2004). An *Overview of Artificial Immune Systems*. In: Paton, R. and Bolouri, H. and Holcombe, M. and Parish, J.H and Tateson, R., eds. Computation in Cells and Tissues: Perspectives and Tools for Thought. Natural Computation Series. Springer, pp. 51-86. ISBN 978-3-540-00358-8.

UCI Archive. https://archive.ics.uci.edu/ml/datasets/Sentiment+Labelled+Sentences. Acessado em 20 de junho de 2021.

Vrajitoru, D. (1998). Crossover improvement for the genetic algorithm in information retrieval. Information processing & management, 34(4), 405-415.

Wang, L., Zhang, Y. e Feng, J. (2005). On the Euclidean distance of images. *IEEE transactions on pattern analysis and machine intelligence*, *27*(8), 1334-1339.

Wason, R. (2018). *Deep learning: Evolution and expansion*. Cognitive Systems Research, 52, 701-708.

Zhang, Y., Jin, R. e Zhou, Z. H. (2010). *Understanding bag-of-words model: a statistical framework*. International Journal of Machine Learning and Cybernetics, 1(1-4), 43-52.