UNIVERSIDADE ESTADUAL DE CAMPINAS
MC886 /M444 - Aprendizado de Máquina

Projeto Final: Relatório Final

Emerson José Bezerra Da Silva - RA 233865
Fernanda Garcia Da Lavra - RA 171345
Vitor Kiyoshi Hiratsuka - RA 256589

Resumo

Inicia-se a discussão do projeto com a seguinte problemática: o comportamento depressivo não só está em sintomas intrínsecos que podem apenas ser diagnosticados por um profissional, mas também em ações do cotidiano que podem ser indícios de tal comportamento, seja na maneira de como essas pessoas interagem em conversas ou, como acontece atualmente, em posts e locais onde elas possam expressar sua opinião.

As redes sociais além de servirem como locais de conversas, podem ser utilizadas como ferramentas de auxílio de diagnóstico de depressão de um determinado indivíduo, já que uma das manifestações dos sintomas pode ser dada através das postagens que ocorrem nessas redes.

Tal abordagem assemelhou-se ao contexto de inteligência artificial, na condição de como conseguimos distinguir uma mensagem depressiva de outras mensagens através de um comportamento bem definido. Pensamos em utilizar esse tópico para implementar uma aplicação que, de maneira automatizada, ajude na percepção dos tipos de mensagens e comportamentos relevantes a serem identificados numa pessoa depressiva.

Introdução

Assim, a problemática apresentada é a detecção de mensagens que tenham conteúdo depressivo. Em especial, a abordagem se deu através de tweets, havendo uma verificação se tal abordagem pode ser utilizada em mensagens de outras aplicações, como por exemplo reddit, com o objetivo de apenas identificar a partir de um valor binário se a mensagem possuirá tal caráter.

Logo, foram planejados os seguintes passos: A procura de um dataset de tweets que conseguisse ser extenso o suficiente para conter além de mensagens aleatórias, mensagens que fossem depressivas; realizar um pré-processamento dessas mensagens para que o conjunto estivesse no padrão que gostaríamos para nossa abordagem; e por último uma montagem de uma rede neural que pudesse ajudar na detecção de textos, com a possibilidade de utilizar ferramentas e referências que já trabalham com interpretação textual para nos auxiliar no problema.

Trabalhos Relacionados

Para a aplicação, foram vistos alguns artigos que chegaram a fazer este tipo de projeto, assim nos inspiramos na maneira como fizeram o pré-processamento bem como discutimos as maneiras de

como abordar a rede neural. Em especial, a rede neural foi inspirada pelo autor Tulasi Ram no Medium [1], que no artigo publicado a respeito de depressão em tweets nos auxiliou em como devíamos tratar os dados bem como indicação do dataset *sentiment 140* [2]. Esse dataset foi criado com auxílio de uma api do twitter no qual foram retirados por volta de 1 milhão e 600 mil tweets durante um certo período de tempo, além de terem já sido distinguidos entre tweets positivos, neutros e negativos.

Algumas redes neurais que tratavam do tema foram vistas para comparação, entre elas a da autora *Pei Jo-Yang* [3] na qual tratou com abordagem semelhante, porém ao invés de utilizar um dataset já criado, que recomendou a utilização de outro dataset feito pela Georgetown University, o que porém ficou impossibilitado já que o dataset necessitava de resposta de email a ser mandado para a universidade, porém mostrou também uma abordagem a partir do mesmo dataset de sentimentos em tweets.

Metodologia

Foi iniciada a metodologia com a análise exploratória, primeiramente com o seguinte problema à ser resolvido: visto que o dataset é composto apenas pelos tweets e uma label em que sabe-se apenas se o conteúdo é positivo, negativo ou neutro, então devemos classificar de uma maneira “manual” as mensagens entre depressivas e não-depressivas, usando os tweets do dataset. Para isso, foram pensados os seguintes passos, primeiramente foi identificado o quão negativa é a mensagem, já que tweets depressivos costumam ser negativos. Segundamente, foram verificadas relações entre certas palavras-chave que, associadas ao índice de negatividade, podem ser consideradas depressivas.

Para o primeiro passo, foi utilizada a biblioteca vaderSentiment, essa é uma ferramenta de análise de sentimentos lexicográfica especialmente utilizada para para redes sociais, sendo assim, ela é capaz de distinguir e analisar também emotes e possíveis expressões com uso de *hashtags*(#). É formada por uma rede que analisa os sentimentos através de regras, isto é, foram criadas regras no modelo que são imutáveis a cada instância analisada, indicando que seus “pesos” na rede são congelados. Ele analisa a mensagem em 2 dimensões, sua positividade e sua negatividade.

Porém, tem-se o seguinte problema: palavras que possuem um radical em comum devem ser classificadas como a mesma “palavra”, um exemplo é para as instâncias “depression” e "depressive", que poderiam ser associadas ao mesmo radical e possível pré-processamento. Devido à isso, também utilizamos uma outra biblioteca para pré-processamento: NLTK (*Natural language tool-kit*), a partir de uma frase, por exemplo (i love this feeling), para os substantivos que podem ser atribuídos um radical, tal biblioteca os modifica para apenas os radicais, ficando (i lov this feel), tal modificação é feita a partir de *stemmers,* que nesse caso foi escolhido o *lancasterStemmer* como modificador das frases, que diferente dos outros parecia o

melhor modificador para encontrar apenas os radicais.

Logo, foram criadas diferentes word clouds (nuvens de palavras) a partir das instâncias nas quais tinham alguma das palavras chaves escolhidas arbitrariamente (palavras essas, ocorrentes em mensagens depressivas) e verificando se o índice de negatividade ao verificar o tweet no vaderSentiment ultrapassa 0,4. Alguns experimentos tomados utilizaram diferentes palavras-chave e diferentes índices, com o índice que melhor atribuía a quantidade necessária de dados para que a rede conseguisse ser treinada. Quanto às palavras, foram colocadas como palavras-chave *depression*, *feeling*, *alone*, *loneliness* e *hate myself*.

Para percorrer cada instância (tweet) do dataset, foi utilizado a biblioteca pandas que auxilia na manipulação de dados especialmente os que são do tipo csv, com representação em dicionários em relação às colunas e atrelar uma coluna 'target' com os valores de 0 para os considerados não depressivos e 1 para os considerados depressivos.

Após essas filtragens e nomeação das instâncias, foram encontradas por volta de 20 mil instâncias que foram consideradas depressivas. Assim, foram misturadas com instâncias não depressivas na seguinte forma:

- Proporção de 10% instâncias depressivas e 90% não depressivas;
- Proporção de 10% validação e 90% de treinamento

Assim, temos:

- 18000 depressivas e 162000 não depressivas no conjunto de treinamento
- 2000 depressivas e 18000 não depressivas no conjunto de validação

Portanto foram definidos os conjuntos de treinamento e

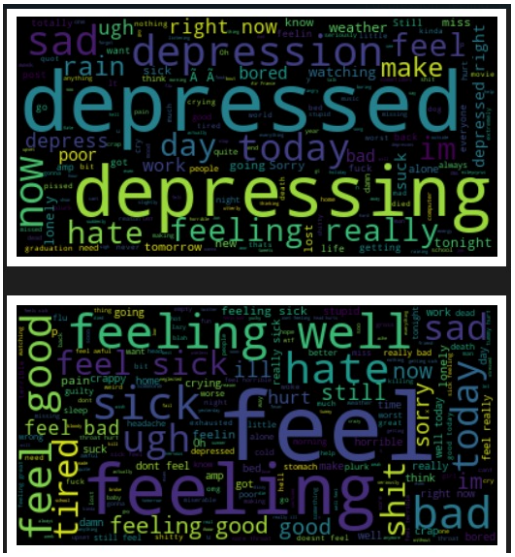

Figura 1- WordClouds para as palavras feeling e depression

Percebe-se que houve uma boa correlação entre as palavras-chave com os tweets de gênero depressivo, com as palavras relacionadas a palavra-chave possuindo valor depressivo também.

validação para assim darmos continuidade no projeto e criarmos a rede neural, com targets sendo 0 para indicar não depressivos, e 1 para os depressivos.

Depois de definido o tratamento dos dados e os objetivos, os dados foram passados por uma rede neural composta pela seguinte estrutura: a primeira camada sendo uma camada de embedding, com os pesos definidos pela vetorização da Google, GoogleNews-Negative300, feita com base num corpo de textos de notícias. Depois disso, para tentar montar uma representação útil para as etapas subsequentes da rede, temos uma camada de convolução unidimensional aliada a uma camada de max-pooling. Com essa representação montada pelas primeiras camadas, aplicamos um dropout para evitar overfitting e passamos o resultado para uma LSTM, escolhida pelo potencial para tratar dados de natureza sequencial, como os textos, que também foi associada a uma camada de dropout, para evitar overfitting. Para produzir a saída no formato correto, depois disso, utilizamos uma função de ativação sigmóide, que conclui a arquitetura da rede.

Uma sutileza envolvida no processo foi a transformação das palavras recebidas no pré-processamento em tokens, para tratamento mais rápido das informações pelas funções prontas, e a montagem de uma matriz de embedding, que será comentada na parte de experimentos, junto com suas motivações.

Experimentação

A primeira parte da experimentação aconteceu já no pré-processamento, em que todo o método para a classificação dos dados a serem utilizados no treinamento da rede precisou ser definido. A técnica das word clouds mostrada acima, usada para garantir que a classificação sendo montada correspondesse à classificação intuitiva do que seriam mensagens depressivas, foi um primeiro ponto dessa experimentação. Fizeram parte dela, além da observação das word clouds, a manipulação dos valores limites para o vader e o uso de diferentes métodos de manipulação textual, antes de optarmos pela exclusão das stop words usando a NLTK, que apresentou os melhores resultados.

Quando o processo de pré-processamento foi definido, chegou a hora do tratamento da rede em si, e nesse tratamento duas áreas de mudança foram as principais: em primeiro lugar, a rede inicialmente não possuía uma camada de embedding e não fazia uso dos dados de vetorização da google. As tentativas de construir nossa própria vetorização se mostraram lentas ao ponto de tornarem o processo de treinamento inviável, uma das razões para isso sendo, provavelmente, a vetorização estar sendo aplicada fora da rede, com matemática não otimizada, diferente do que acontece com o caso de uma camada de embedding e uma matriz de embedding, que permitem busca rápida para a vetorização de qualquer palavra, com o tamanho da matriz sendo mantido razoável pelo uso de apenas um número

finito de palavras mais utilizadas, tendo sido testado, por razões da viabilidade do treinamento, com 300. A segunda mudança principal foi que, inicialmente, foi pensada uma rede convolucional pura, para testar o poder dessa arquitetura em resolver um problema baseado em tratamento de linguagem natural. Essas tentativas foram bastante iniciais e não apresentaram qualquer potencial, não aproximando nem os dados de treinamento com processos razoáveis para treinar a rede, e foram logo abandonadas, em favor da adição de uma camada de textos LSTM, que vimos ser um padrão para redes simples para tratamento de textos. Esse foi o maior salto de qualidade experimentado pelo projeto e, depois disso, apenas ajustes menores foram tentados para a rede, sem resultados que pudessem ser considerados estatisticamente relevantes. O número de camadas de convolução não apresentou melhoria significativa apesar do significativo aumento do tempo de treinamento que custaria para implementar, enquanto que a troca da função de ativação, apesar de não sofrer do mesmo problema, não trouxe qualquer benefício significativo. Os resultados finais para os dados de treino e validação são apresentados na Figura 2.

Uma última experimentação foi feita com o conjunto de dados, buscando testar o poder de generalização da rede produzida. A rede foi testada contra dados já marcados da rede social reddit, de características bastante diferentes das do twitter, que estavam disponíveis em quantidades bem menores, mas suficientes para um teste numa rede já treinada. Nesse teste, os resultados decaíram e foi observada uma queda da taxa de acertos para 54,77% de acurácia balanceada e 55,14% de acurácia, que será discutida na conclusão.

Conclusão

Os resultados anteriores, para o conjunto de treino e validação, indicam grande potencial para o método, considerando a simplicidade da arquitetura e as restrições quanto ao treinamento. Os resultados da generalização, porém, indicam falhas graves na possibilidade de produção de um modelo genérico, aplicável a diversas redes sociais. Primeiro serão apresentadas algumas possíveis explicações para essas falhas de generalização, tentando encontrar sentido nos resultados, depois, serão apresentadas algumas propostas de possíveis mitigações.

A principal causa provável é a diferença de natureza das redes.

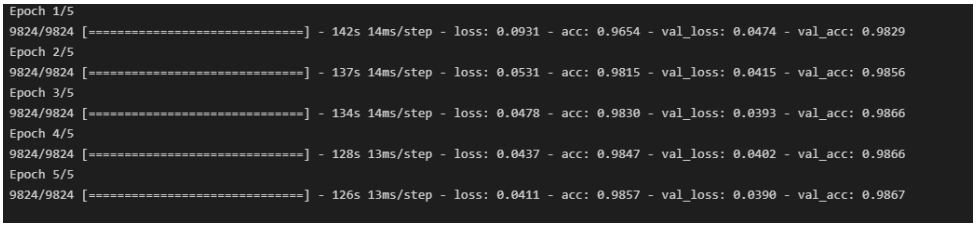

Figura 2 - Épocas e evolução do modelo de acordo.

Isso se manifesta de duas formas: sendo o Reddit um fórum de discussões enquanto que o Twitter é voltado para interações casuais, os posts depressivos do Reddit são mais tipicamente desabafos, bastante diferentes das declarações mais curtas e diretas presentes no Twitter. A rede pode ter aprendido somente esse segundo tipo e com isso classificado erroneamente o primeiro. Além disso, pela mesma causa, os posts do Reddit são tipicamente mais longos, enquanto que a camada de convolução da rede é treinada para tratar apenas as primeiras 280 palavras e, pelo treinamento diferenciado das palavras com base na posição delas no texto, pode tratar ainda menos do que isso, considerando quão pouco treinamento as últimas posições receberam com base nas mensagens curtas do Twitter, o que pode afetar a capacidade de representação das mensagens mais longas e detalhadas do Reddit, e portanto a capacidade do sistema de as classificar.

Uma segunda causa pode ser uma diferença de vocabulário, considerando a dependência da matriz de embedding para a classificação, que faria que o sistema não fosse capaz de se adaptar bem a um site que usasse linguagem própria diferente da do Twitter.

As propostas de soluções serão as seguintes:

Em primeiro lugar, expandir a matriz de embedding para tratar de mais palavras, permitiria mitigar o problema da mudança de linguagem, tendo como troca um aumento da complexidade da rede.

Em segundo lugar, conforme sugestões, o processamento da membros, foi bastante útil para analisarmos e entendermos o linguagem antes da rede LSTM poderia ser feito utilizando a rede pré-treinada BERT. Isso reduziria o acoplamento entre o modo da representação e a natureza dos dados do Twitter e, pela característica do BERT de ser um modelo extremamente capaz para tratamento de linguagem natural, ele poderia ser alimentado diretamente para a LSTM sem a necessidade das camadas de convolução que foram utilizadas para facilitar o tratamento dos vetores na presente versão.

Por fim, a estratégia de treinamento de uma rede para uso geral poderia ser modificada, isso é, treinar em uma rede social que tenha tanto posts longos como curtos, como o Reddit, e então tentar generalizar para uma que contenha só posts curtos, como o Twitter, como estratégia mais viável.

Diante de tudo isso, o trabalho teve sucesso em resolver o problema específico que se propôs a resolver, que era a classificação de tweets, mas falhou em resolver outro problema de natureza similar, que era a classificação de posts do Reddit. Esse segundo fato é uma oportunidade maior de aprendizado, por dois motivos: primeiro, o teste proposto foi uma experiência bastante valiosa para testarmos os domínios de um problema de aprendizado de máquina, que podem ser bem mais estreitos do que o normalmente imaginado quando se menciona uma ‘rede treinada para classificar postagens’.

Esse conhecimento das possíveis restrições ao domínio e a necessidade de testar para elas foi certamente uma novidade para os envolvidos no projeto, e a ideia do teste, que demorou a ocorrer aos problema. Segundo, a análise do ponto de falha da rede foi permitido

um olhar mais crítico ao projeto como um todo e possibilitou uma auto-avaliação para o que foi realizado, que também foi uma oportunidade considerável para aprendizado, principalmente por forçar a sugestão de explicações para a falha e possíveis melhorias para a correção ou mitigação dela, além de forçar o grupo a olhar não só para o trabalho, mas para o problema em si, com mais atenção, para notar o que exatamente diferia entre o caso de sucesso e o caso de falha.

Ademais, tanto no geral do projeto de redes para aprendizado de máquina como no problema específico do tratamento de postagens de redes sociais para encontrar indicadores emocionais, o grupo saiu mais apto depois da realização do projeto, independente dos resultados finais para a generalização do problema.

Referências

[1]TULASIRAM. Detecting Depression in Social Media Via Twitter Usage. Disponível em: <https://medium.com/swlh/detecting-depression-in-social-media-via-twitter-usage-2d8f3df9b313>.

[2] Sentiment140 dataset with 1.6 million tweets. Disponível em: <https://www.kaggle.com/datasets/kazanova/sentiment140>.

[3] PEIJO. Detect Depression In Twitter Posts. Disponível em: <https://github.com/peijoy/DetectDepressionInTwitterPosts>. Acesso em: 12 dez. 2022.

[4] RAM, T. Detecting Depression using Tweets ! Disponível em: <https://github.com/ram574/Detecting-Depression-using-Tweets/>. Acesso em: 12 dez. 2022