ANÁLISE DE SENTIMENTOS VIA REDES NEURAIS BASEADAS EM GRAFOS E APRENDIZADO SEMI-SUPERVISIONADO

Palavras-Chave: Análise de Sentimentos, Redes Neurais, Grafos, Aprendizado Semi-supervisionado

Autores/as:

Igor Paulo dos Santos Santana - FT, UNICAMP

Prof. Dr. Guilherme Palermo Coelho (orientador) - FT, UNICAMP

Prof. Dr. Arthur Emanuel de Oliveira Carosia (coorientador) - IFSP – S. J. da B. Vista

INTRODUÇÃO:

A análise de sentimentos é o processo de identificar e categorizar opiniões expressas em um texto para determinar se a atitude do autor em relação a um tópico específico é positiva, negativa ou neutra. Essa técnica tem se tornado uma ferramenta essencial em diversas áreas, desde o monitoramento de opiniões em redes sociais até a avaliação de produtos e serviços. Tradicionalmente, essa análise é feita utilizando métodos de *machine learning* supervisionados, que requerem grandes volumes de dados rotulados. No entanto, a rotulação manual de dados é um processo caro, demorado e sujeito a erros por parte do responsável (Manning, Raghavan, & Schütze, 2008).

Para contornar essas limitações, o presente projeto investiga o uso de *Graph Neural Networks* (GNNs) e aprendizado semi-supervisionado na análise de sentimentos. As GNNs são capazes de explorar as relações complexas entre dados representados em grafos, o que pode melhorar a qualidade da análise sem a necessidade de grandes quantidades de dados rotulados (Yao, Mao, & Luo, 2019).

Este projeto também visa aplicar as GNNs em conjunto com outros modelos de redes neurais, verificando a viabilidade do uso das próprias como modelos auxiliares para rotular grandes quantidades de dados, utilizando uma pequena parcela para seu treinamento e permitindo, desse modo, o uso posterior de outros modelos treinados especificamente com os dados gerados pelas GNN para rotular o sentimento presente em novas amostras dos dados. Neste trabalho, foi utilizado nesta segunda etapa o modelo Long Short-Term Memory (LSTM).

METODOLOGIA:

No aprendizado supervisionado, os modelos são treinados com dados rotulados, o que significa que a cada entrada de dados é associada a uma saída desejada. Já em aprendizado semi-supervisionado, o modelo utiliza uma combinação de dados rotulados e não-rotulados. Esta última abordagem economiza recursos computacionais e humanos ao aproveitar grandes volumes de dados não-rotulados, mas pode comprometer a acurácia e outros parâmetros do modelo, devido à possibilidade de não haver uma boa quantidade de informações relevantes durante seu treinamento, afetando a capacidade do modelo de diferenciar classes de dados ou chegar a um valor preciso em problemas de regressão, por exemplo (Kingma et al., 2014; Müller & Guido, 2016).

Dado esses conceitos, o primeiro passo foi explorar as redes neurais tradicionais, compreendendo os elementos fundamentais como camadas, neurônios, funções de ativação e o processo de backpropagation (Müller & Guido, 2016). Em seguida, o foco foi direcionado para as GNNs, uma extensão das redes neurais para lidar com dados estruturados em forma

de grafos, sendo elas especialmente úteis para dados com relações complexas, como redes sociais e sistemas de recomendação (Yao, Mao, & Luo, 2019).

• Treinamentos e validação dos modelos GNN's

Dentre os tipos de GNN's utilizados, destacam-se os modelos *Graph Convolutional Network* (GCN) e *Graph Attention Network* (GAT). O GCN é um modelo que aplica operações de convolução diretamente em grafos, permitindo a agregação de informações dos nós vizinhos para capturar características estruturais e relacionamentos na rede. Ele utiliza um processo iterativo para atualizar as representações dos nós com base nas suas conexões, proporcionando uma forma eficaz de análise de grafos (Kipf & Welling, 2017).

Já o GAT introduz mecanismos de atenção nos grafos, permitindo que o modelo aprenda a importância relativa das conexões entre nós. Cada aresta do grafo recebe um peso de atenção, o que possibilita ao modelo focar mais nas conexões mais relevantes durante a agregação de informações. Isso resulta em uma representação de nó mais refinada e adaptativa, aprimorando o desempenho em tarefas de aprendizado em grafos (Veličković et al., 2018).

A preparação dos dados envolveu a análise de uma base de notícias cujos sentimentos variam entre positivo e negativo. Inicialmente, foram eliminados elementos desnecessários, como *stop words* e pontuação. Em seguida, os textos foram convertidos em *tokens*. Foi utilizado o modelo pré-treinado *BERTimbau Large* para criar *embeddings* contextuais, representando cada frase de forma numérica em um espaço vetorial contínuo (Devlin et al., 2019).

A partir desses *embeddings*, foram criados os vértices do grafo, e as arestas foram determinadas com base na similaridade de cosseno entre os vértices, ou seja, caso o valor da similaridade entre os *embeddings* de dois vértices seja maior que um limiar definido manualmente, é criada uma aresta entre dois vértices, cuja existência permite a troca de mensagens diretas entre esses vértices durante o processamento dos modelos modelos baseados em grafos, sendo esse o principal mecanismo de aprendizagem de tais modelos (Manning, Raghavan, & Schütze, 2008).

Para o treinamento do modelo, o conjunto de dados, com um total de 1116 amostras, foi dividido em seções de treinamento e teste (20% e 80% para cada, respectivamente), preservando as relações entre os vértices e a proporção de notícias positivas e negativas em cada conjunto (50/50 na base de dados completa). Utilizamos *backpropagation* para ajustar os pesos dos neurônios e a performance foi avaliada usando métricas como acurácia, precisão e *recall* (Graves, Mohamed, & Hinton, 2013). Vale ressaltar que cada resultado, apresentado e discutido posteriormente na seção de Resultados e Discussões, deriva da média e desvio-padrão calculados a partir de um total de 100 execuções de teste.

• Modelo GAT em auxílio de um LSTM

Vale conceituar também o modelo neural *Long Short-Term Memory* (LSTM), que é uma variante das redes neurais recorrentes (RNN) projetada para lidar com a dependência de longo prazo em dados sequenciais. O LSTM utiliza unidades de memória especiais, conhecidas como células LSTM, que podem armazenar e acessar informações ao longo de longas sequências de dados, mitigando o problema do desaparecimento do gradiente encontrado em RNNs tradicionais. Essas células são controladas por três tipos de portas: a porta de entrada, a porta de esquecimento e a porta de saída, que juntas regulam o fluxo de informações e decidem quais dados são mantidos ou descartados durante o treinamento. Isso torna o LSTM particularmente eficaz em tarefas como previsão de séries temporais, tradução automática e processamento de linguagem natural (Hochreiter & Schmidhuber, 1997).

.Após o desenvolvimento e avaliação inicial das GNN's, utilizamos um modelo GAT para verificar a viabilidade do uso das GNN's aliadas a outros tipos de modelos para identificação do sentimento presente nos títulos de notícias. Buscou-se

também identificar se seria possível utilizar a combinação GAT com LSTM, eficientemente, para classificar notícias que não estejam contidas na base de dados em forma de grafo.

O problema que se buscava solucionar com esse teste consiste em, caso a base de dados exija constantes alterações, não precisar construir novamente o grafo inicial, utilizando-se do já existente para continuar classificando novos dados sem a necessidade de reconstruí-lo. Esta é uma etapa necessária sempre que se deseja incluir novos dados nele (seja para treinamento ou teste) e geralmente é muito custosa.

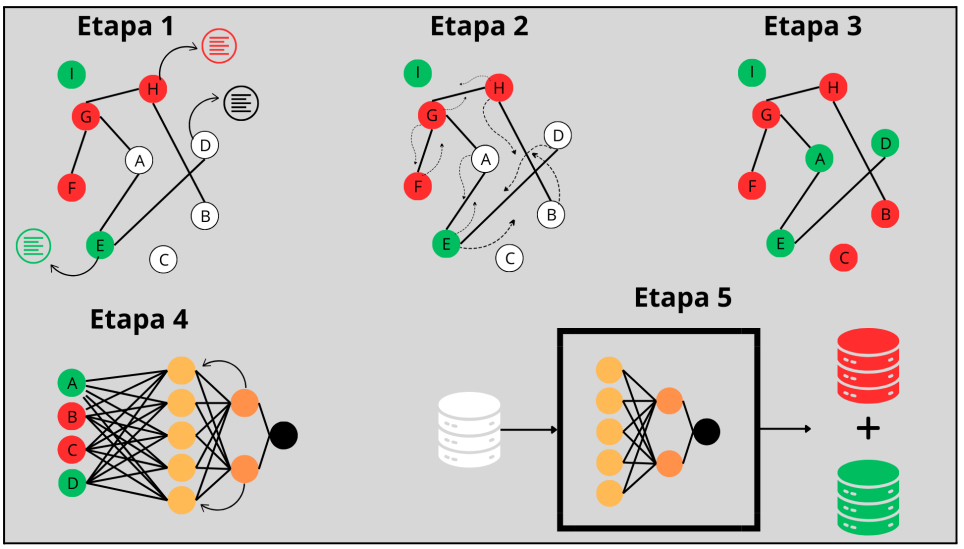

Figura 1 - Demonstração em etapas do uso do modelo GAT em auxílio do modelo LSTM onde as cores representam os sentimentos atrelados aos títulos, sendo o verde associado aos rótulos positivos, vermelho aos negativos e branco aos não avaliados.

Como ilustrado na Figura 1, na primeira etapa há a construção do grafo representativo do conjunto de notícias disponível. Este é composto por nós e arestas, sendo eles representados pelos círculos e retas, respectivamente. Já os círculos, que contém textos, representam os *embeddings* associados a cada nó. Vale ressaltar que não se conhece o sentimento de todas as notícias (nós, nesse caso), ou seja, a base não está completamente rotulada.

Na segunda etapa, já dentro do modelo GAT, é representada a troca de informações entre os nós do grafo através das arestas. Na Figura 1, são utilizadas setas pontilhadas como sinal da atuação dos mecanismos de atenção associados a cada uma delas. Este processo é repetido várias vezes durante o treinamento do modelo a depender de como o próprio é definido. Ao final de todas as repetições, conclui-se a sessão de treinamento e o modelo parte para a sessão de testes. Na terceira etapa, todos os nós destinados à sessão de teste são classificados e seus rótulos são retornados pelo modelo.

Na quarta etapa, o conteúdo (*embeddings*, nesse caso) dos nós rotulados anteriormente é utilizado para a sessão do treinamento do modelo LSTM. Vale ressaltar que foram feitas duas versões destes experimentos: a primeira envolve o uso apenas dos textos com rótulos preditos pelo modelo GAT para o treinamento da LSTM e a segunda inclui também os 20% restantes dos dados, que haviam sido rotulados manualmente. A comparação, com o uso do modelo LSTM por si só, segue a mesma regra.

Por fim, na quinta etapa é realizada a validação do modelo LSTM utilizando uma segunda base de títulos de notícias. Seus *embeddings* são inseridos no modelo e os rótulos resultantes são comparados aos definidos à mão. Para representar essas novas notícias, foi utilizada uma nova base de dados de 100 notícias, igualmente balanceada, para validação do modelo LSTM.

RESULTADOS E DISCUSSÃO:

Modelos	Acurácia	Precisão	Recall	F1-Score
GCN 20%	76,00% ± 3,18%	69,00% ± 7%	92,00% ± 9%	78,86% ± 6,9%
GAT 20%	81,55% ± 0,56%	79,00% ± 1%	86,00% ± 2%	82,35% ± 1,3%

Tabela 1 - Modelos GNN's e métricas de desempenho

Modelos	Acurácia	Precisão	Recall	F1-Score
GAT + LSTM 80% (Rótulos preditos)	85,00% ± 2%	89,00% ± 2%	80,00% ± 5%	84,26% ± 3,3%
LSTM 80% (Rótulos manuais)	88,00% ± 2%	95,00% ± 2%	81,00% ± 4%	87,44% ± 2,7%
GAT + LSTM 80% (Rótulos preditos) + 20% (Rótulos manuais)	85,00% ± 2%	90,00% ± 1%	79,00% ± 4%	84,14% ± 2.5%
LSTM 100% (Rótulos manuais)	88,00% ± 1%	95,00% ± 3%	80,00% ± 3%	86,86% ± 3,55%

Tabela 2 - Modelos LSTM + GAT, LSTM e métricas de desempenho

Como é possível visualizar na Tabela 1 e Tabela 2, os resultados obtidos neste projeto destacam a eficiência das *Graph Neural Networks* (GNN's) e do aprendizado semi-supervisionado na análise de sentimentos. Inclusive, vale destacar também a capacidade, observada e avaliada durante a pesquisa, de utilizar modelos baseados em grafos em conjunto com outros a fim de otimizar recursos financeiros, computacionais e de tempo em si.

Antes de discutir os resultados, vale destacar que as porcentagens presentes ao lado da sigla do modelo (coluna *Modelos* na Figura 1) referem-se à porcentagem da base de dados inicial, aquela que contém 1116 amostras, utilizada para treinamento do(s) respectivo(s) modelo(s) baseados em grafos (GCN e GAT). Outro ponto a ser destacado é que os "rótulos preditos" correspondem àqueles que foram gerados pelo modelo enquanto os "rótulos manuais", como a nomenclatura sugere, referem-se àqueles definidos à mão pelos especialistas.

Analisando inicialmente, dentre somente os dois modelos de GNN's, é possível observar uma clara superioridade do modelo atencional (GAT) frente o convolucional (GCN) para este problema de classificação. Isso se deve ao fato do modelo atencional possuir componentes exclusivamente dedicados a capturar diferentes perspectivas dos vetores de *features* dos nós, permitindo uma melhor balanço entre generalização e especialização na hora de rotular os nós restantes.

Outro ponto a ser destacado é a diferença entre os desvios-padrão dos resultados entre esses modelos, sendo o modelo GAT muito mais estável nesse quesito, tornando-o uma aposta mais segura para esse problema em específico.

Por fim, analisando as quatro últimas linhas da Tabela 1, correspondentes ao uso em conjunto dos modelos GAT e LSTM, observou-se que o uso de rótulos gerados pelo modelo GAT levaram a resultados satisfatórios com a LSTM, mesmo quando aplicados a novos dados, sugerindo que rótulos gerados automaticamente podem ser eficazes em cenários de aprendizado semi-supervisionado.

Adicionalmente, a combinação do GAT com LSTM permitiu uma boa classificação de novas notícias sem a necessidade de reconstrução do grafo. Este método demonstrou ser uma boa opção, apresentando apenas uma pequena perda de precisão (2-3%, na média), mas mantendo robustez na análise de sentimentos.

Comparando o desempenho da LSTM treinada com a mistura de rótulos manuais com os preditos pela GAT, no penúltimo modelo avaliado, e o uso de dados rotulados manualmente no último mostram que, nesse caso, houve uma pequena melhoria das métricas de desempenho.

CONCLUSÕES:

Este estudo demonstrou que a utilização de *Graph Neural Networks* (GNN's) e aprendizado semi-supervisionado na análise de sentimentos é uma abordagem promissora e eficiente. As GNN's mostraram-se capazes de alcançar alta precisão utilizando uma menor quantidade de dados rotulados, economizando recursos computacionais e humanos. O uso dos modelos GAT e LSTM, de forma complementar, mostrou-se uma estratégia eficaz, com apenas pequenas perdas nas métricas de desempenho, variando de 2 a 3%, mas mantendo uma alta precisão na classificação de sentimentos em notícias.

Esses resultados indicam que a abordagem híbrida de GNN's com outras redes neurais, como o LSTM, pode ser aplicada de em cenários onde a rotulação de dados é limitada e onde há necessidade de processamento contínuo de novos dados. A pesquisa abre caminhos para futuras investigações na otimização e aplicação de GNN's em diferentes contextos de análise de sentimentos e outras tarefas de processamento de linguagem natural.

BIBLIOGRAFIA

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, v. 1, p. 4171–4186. DOI: 10.18653/v1/N19-1423.

Diederik P. Kingma, Danilo J. Rezende, Shakir Mohamed, Max Welling. (2014). Semi-Supervised Learning with Deep Generative Models. Proceedings of Neural Information Processing Systems (NIPS), v. 2. DOI: 10.48550/arXiv.1406.5298.

Graves, A., Mohamed, A. R., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 2013, pp. 6645-6649. DOI: 10.1109/ICASSP.2013.6638947.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Comput 1997, v. 9, p. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

Januario, B. A., Carosia, A. E. O., Silva, A. E. A., & Coelho, G. P. (2022). Sentiment Analysis Applied to News from the Brazilian Stock Market. IEEE Latin America Transactions, v. 20, p. 512-518. DOI: 10.1109/TLA.2022.9667151.

Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. Proceedings of the International Conference on Learning Representations (ICLR). DOI: 10.48550/arXiv.1609.02907.

Manning, C. D., Raghavan, P., & Schütze, H. (2008). Introduction to Information Retrieval. Cambridge University Press, v. 1, p. 537. Disponível em: https://nlp.stanford.edu/IR-book/information-retrieval-book.html.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. Proceedings of Workshop at ICLR, 2013, v. 3. DOI: 10.48550/arXiv.1301.3781.

Müller, A. C., & Guido, S. (2016). Introduction to Machine Learning with Python: A Guide for Data Scientists. O'Reilly Media, v. 1, p. 398.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2018). Graph Attention Networks. Proceedings of the International Conference on Learning Representations (ICLR). DOI: 10.48550/arXiv.1710.10903.

Yao, L., Mao, C., & Luo, Y. (2019). Graph Convolutional Networks for Text Classification. Proceedings of the AAAI Conference on Artificial Intelligence, v. 33, n. 01, p. 7370-7377. DOI: 10.1609/aaai.v33i01.33017370.