Exploração de Técnicas de Regularização em Redes Neurais Recorrentes Uni/Bidirecionais para Análise de Sentimentos

Ricardo Cabral Penteado
Universidade de São Paulo
ricardo.penteado@usp.br

RESUMO

Este trabalho explora a aplicação de Redes Neurais Recorrentes (RNNs), particularmente a arquitetura LSTM, na análise de sentimentos de revisões de produtos. Foi avaliado o impacto de configurações unidirecionais e bidirecionais, bem como diferentes taxas de dropout para mitigar overfitting. O modelo Bidirectional LSTM com uma taxa de dropout de 50.0% mostrou-se o mais promissor, atingindo uma acurácia de teste de 61.3%. A análise revelou desafios na classificação de sentimentos na granulosidade das nuances das classes intermediárias, ao passo que avaliações extremamente positivas ou negativas foram melhor identificadas. Para aprimorar a performance do modelo, sugere-se exploração de arquiteturas avançadas, engenharia de características, análise de erros, ajuste de hiperparâmetros e utilização de dados adicionais. A pesquisa contribui para a compreensão do comportamento de RNNs em tarefas de análise de sentimentos, vislumbrando otimizações futuras para melhor desempenho em tarefas de Processamento de Linguagem Natural (PLN).

1 Introdução

A análise de sentimentos é uma área proeminente dentro da mineração de texto e da aprendizagem de máquina, oferecendo insights valiosos sobre as opiniões e emoções expressas em textos escritos. Com o advento e a evolução das redes neurais recorrentes (RNNs), a análise de sentimentos atingiu novas fronteiras de precisão e eficácia. Dentro do escopo das RNNs, as arquiteturas Long Short-Term Memory (LSTM) emergiram como uma solução robusta para os desafios apresentados pela dependência de longo prazo dos dados sequenciais.

Um dos avanços significativos nas arquiteturas LSTM é a incorporação de camadas bidirecionais, que permitem que a rede capture dependências tanto para frente quanto para trás no tempo. Esta característica é de particular importância na análise de sentimentos, onde o contexto pode ser crucial para entender a semântica das opiniões expressas.

No entanto, o desempenho dos modelos LSTM pode ser significativamente afetado por problemas comuns em aprendizado de máquina, nomeadamente o *underfitting* e o *overfitting*. O *underfitting* ocorre quando o modelo é demasiadamente simples para capturar a estrutura subjacente dos dados, enquanto o *overfitting* manifesta-se quando o modelo aprende demasiadamente a estrutura de ruído nos dados de treino, falhando em generalizar bem para dados não vistos.

Este trabalho visa aprofundar a compreensão dos alunos sobre redes neurais recorrentes bidirecionais em arquiteturas LSTM aplicadas à análise de sentimentos. Em particular, focamos na exploração e mitigação dos problemas de *underfitting* e *overfitting*. Utilizamos técnicas como a inserção de uma camada Dropout durante o treinamento, uma estratégia eficaz para evitar o *overfitting*, mantendo uma arquitetura de modelo robusta.

Além disso, enfatizamos a importância de uma divisão adequada do corpus em conjuntos de treinamento, validação e teste, para garantir uma avaliação honesta e informativa do desempenho do modelo. Ao longo deste exercício, exploramos diversas configurações e hiperparâmetros, proporcionando uma visão detalhada sobre como cada aspecto influencia a capacidade do modelo de aprender eficazmente a tarefa de análise de sentimentos e generalizar para dados não vistos.

A estrutura do restante deste artigo é a seguinte: na Seção 2, detalhamos a metodologia empregada, incluindo as configurações e hiperparâmetros utilizados. Na Seção 3, apresentamos e discutimos os resultados obtidos. Finalmente, na Seção 4, concluímos o artigo, refletindo sobre as descobertas e sugerindo direções para trabalhos futuros.

2 Metodologia

2.1 Configurações e Hiperparâmetros

Para abordar a tarefa de classificação de avaliações de produtos, foi solicitado o uso de uma arquitetura baseada em Long Short-Term Memory (LSTM). As LSTMs são uma variante das Redes Neurais Recorrentes (RNNs), reconhecidas por sua habilidade em lidar com dependências temporais longas nos dados, tornando-se particularmente eficazes para tarefas de processamento sequencial como a análise de sentimentos em textos. As LSTMs foram introduzidas por Sepp Hochreiter e Jürgen Schmidhuber em 1997, no artigo "Long Short-Term Memory”.

O núcleo da arquitetura LSTM é a célula de memória, que é composta por três portas: a porta de entrada, a porta de esquecimento e a porta de saída. Essas portas

controlam o fluxo de informações dentro e fora da célula de memória, permitindo que a rede mantenha ou descarte informações ao longo do tempo. Essa estrutura permite que as LSTMs armazenem, acessem e processem informações sequenciais de maneira mais eficaz ao longo de longas sequências, tornando-as bem adequadas para tarefas como análise de sentimentos, onde o contexto de longo alcance pode ser crucial para entender o sentimento expresso no texto.

Os modelos foram experimentados em configurações unidirecionais e bidirecionais. As LSTMs bidirecionais, processando a sequência de entrada em ambas as direções, proporcionam ao modelo acesso ao contexto passado e futuro, o que pode ser crucial na análise textual.

A inovação das Redes Neurais Recorrentes Bidirecionais (BiRNNs) foi apresentada no artigo "Bidirectional Recurrent Neural Networks" por Mike Schuster e Kuldip K. Paliwal em 1997. As BiRNNs consistem em duas RNNs, uma processando a sequência de entrada da esquerda para a direita e outra processando a sequência da direita para a esquerda. A saída final em cada ponto de tempo é então uma combinação das saídas de ambas as RNNs.

A técnica de regularização *dropout* foi empregada para mitigar o *overfitting*, com taxas de *dropout* de 0.0, 0.25 e 0.5 sendo testadas. O *dropout* é uma técnica eficaz para mitigar o risco de overfitting em redes neurais, conforme proposto por Srivastava et al. (2014). Ao aplicar o *dropout*, uma fração dos neurônios na rede é "desligada" aleatoriamente durante cada iteração de treinamento, criando efetivamente múltiplas sub-redes que são treinadas em conjunto. Isso evita que qualquer neurônio se torne crítico para a solução, promovendo uma representação mais robusta e distribuída dos dados. O dropout foi aplicado após s camada LSTM e antes da camada densa.

A análise inicial dos dados revelou insights importantes que guiaram a seleção de hiperparâmetros cruciais. O comprimento das avaliações variava, com uma distribuição como demonstrado no histograma gerado. Para garantir que os modelos pudessem processar as revisões de forma eficaz, foi necessário padronizar o comprimento das sequências de entrada. O tamanho máximo de sequência foi definido como 124, com base no percentil de 99% dos comprimentos das revisões. Este valor foi escolhido para garantir que a maioria das revisões estivesse totalmente representada, enquanto mantinha a complexidade computacional sob controle.

A tabela de percentil abaixo demonstra a distribuição dos comprimentos das revisões:

Percentil	Comprimento Máximo
85%	41
90%	50
95%	69
99%	124
100%	794

Tabela 1: Distribuição dos comprimentos das frases e seu percentil

O tamanho do vocabulário original era de 43,893 palavras. No entanto, para gerenciar a complexidade do modelo e acelerar o treinamento, o vocabulário foi truncado para as 20,000 palavras mais frequentes.

O uso de embeddings de palavras pré-treinados é uma prática comum para injetar conhecimento semântico prévio em modelos de aprendizado profundo para tarefas de Processamento de Linguagem Natural (PNL). Os embeddings capturam relações semânticas e sintáticas entre as palavras em um espaço vetorial de baixa dimensão, facilitando assim a aprendizagem de padrões complexos por modelos subsequentes.

O NILC-Embeddings é um repositório que visa armazenar e compartilhar embeddings de palavras gerados especificamente para a Língua Portuguesa, tanto do Brasil quanto de Portugal. Esses embeddings foram gerados a partir de um vasto *corpus* compreendendo 1,395,926,282 *tokens*, extraídos de dezessete *corpus* diferentes, abrangendo fontes e gêneros variados. Os vetores foram treinados utilizando algoritmos consagrados como *Word2vec, FastText, Wang2vec* e *Glove*.

No contexto do presente trabalho, optou-se por utilizar os embeddings de palavras pré-treinados GloVe com 300 dimensões para inicializar a camada de embedding do modelo. A escolha do GloVe foi motivada pela sua capacidade de capturar tanto a estatística global da co-ocorrência de palavras quanto as relações semânticas e sintáticas locais. A inicialização com esses embeddings facilitou a convergência durante o treinamento e potencialmente melhorou a performance do modelo ao incorporar conhecimento semântico prévio.

Os modelos foram treinados com um tamanho de batch de 64, um valor comum que proporciona um bom compromisso entre a eficiência computacional e a precisão do gradiente.

Cada combinação de LSTM bidirecional (ou não), taxa de *dropout*, foi experimentada em modelos separados. O desempenho de cada configuração foi avaliado em um conjunto de dados de teste separado, com os modelos sendo salvos em arquivos .model,

do pacote keras, e carregados a partir de um caminho especificado para reutilização e avaliação futura.

O pré-processamento dos dados de entrada foi crucial para garantir uma representação adequada para o treinamento do modelo. O texto das avaliações foi tokenizado, e as sequências de *tokens* foram padronizadas para um comprimento uniforme.

A avaliação do desempenho do modelo foi focada na acurácia de classificação no conjunto de dados de teste. Relatórios de classificação foram gerados para fornecer uma análise mais detalhada do desempenho do modelo em termos de precisão, revocação e pontuação F1 para cada classe de avaliação.

A seleção e otimização dos hiperparâmetros foram guiadas pelo objetivo de maximizar a acurácia de classificação no conjunto de dados de teste, com um cuidado especial para evitar o *overfitting*, monitorado através dos gráficos de validação gerados durante o treinamento.

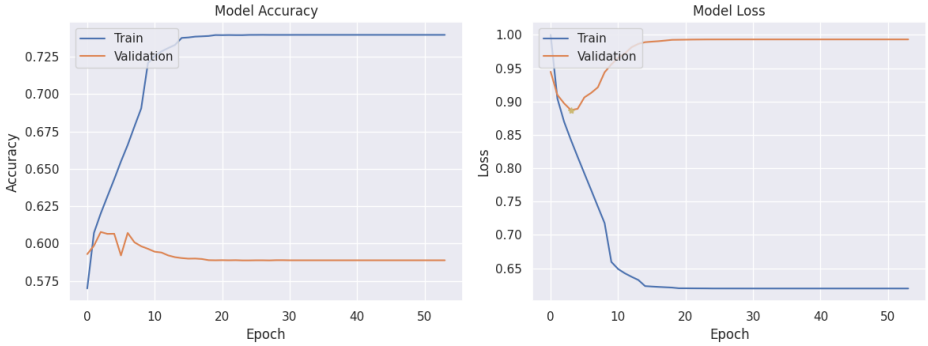
2.2 Validação

A validação e seleção do modelo adequado são etapas essenciais para garantir que o modelo treinado seja robusto e capaz de generalizar bem para dados não vistos. Durante o treinamento, monitoramos a performance do modelo em ambos os conjuntos de treinamento e validação. Utilizamos gráficos de validação para visualizar a evolução das métricas de desempenho - especificamente, a acurácia e a perda - ao longo das épocas de treinamento. Esses gráficos são vitais para identificar sinais de *overfitting* e ajudar na seleção do número ótimo de épocas para treinar o modelo.

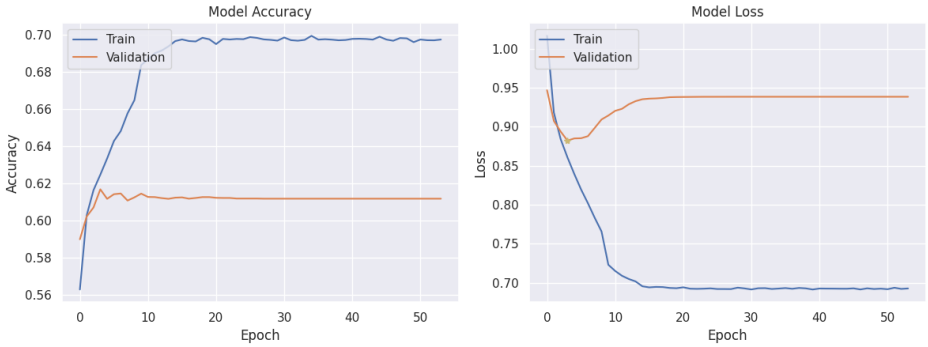
O *overfitting* geralmente ocorre quando a performance do modelo no conjunto de treinamento continua a melhorar, mas começa a deteriorar no conjunto de validação. Isso indica que o modelo está aprendendo padrões específicos do conjunto de treinamento que não são generalizáveis para outros dados. A estratégia de *early stopping* foi adotada, onde o treinamento é interrompido assim que a performance no conjunto de validação começa a deteriorar. Todos modelos estavam programados para rodar 100 épocas com *early stopping* de 50 épocas.

Seguem os gráficos do processo de treinamento:

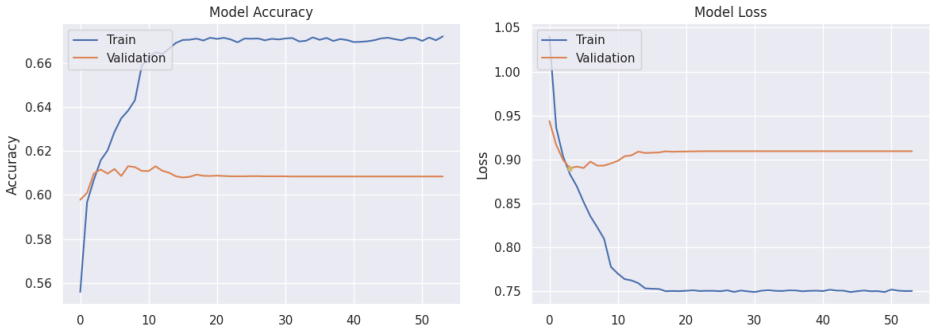
Unidirectional LSTM with 0.0% Dropout



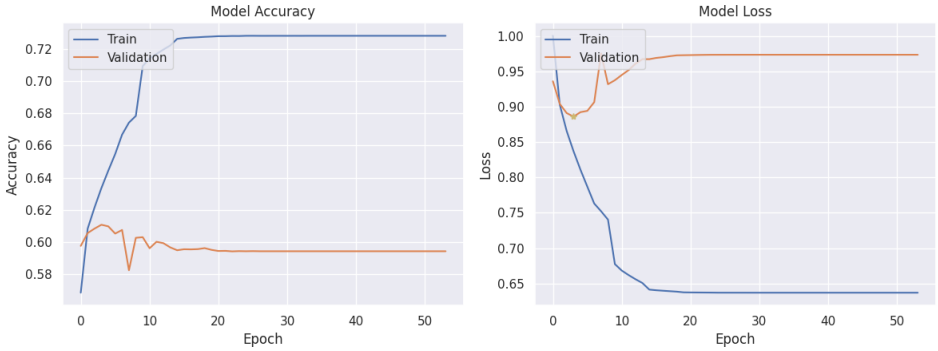
Unidirectional LSTM with 25.0% Dropout



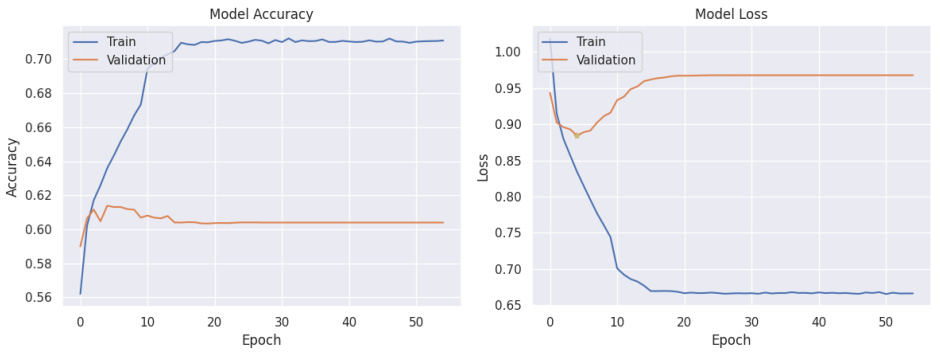
Unidirectional LSTM with 50.0% Dropout



Bidirectional LSTM with 0.0% Dropout



Bidirectional LSTM with 25.0% Dropout



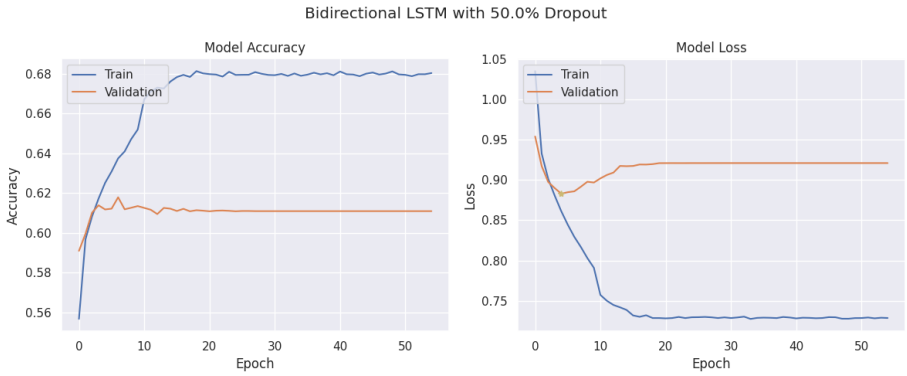


Figura 1: Gráficos de acurácia e perda de todos os modelos testado no conjunto de validação

Os detalhes das épocas selecionadas e as perdas de validação correspondentes para cada configuração do modelo são apresentados a seguir:

Modelo	Melhor época	Validation Loss
Unidirectional LSTM with 0.0% Dropout	4	0.8872
Unidirectional LSTM with 25.0% Dropout	4	0.8823
Unidirectional LSTM with 50.0% Dropout	4	0.8901
Bidirectional LSTM with 0.0% Dropout	4	0.8862
Bidirectional LSTM with 25.0% Dropout	5	0.8847
Bidirectional LSTM with 50.0% Dropout	5	0.8829

Tabela 2: Registro das melhores épocas quanto a perda no conjunto de validação

Como se pode ver, dentre todos os modelos testados, o Unidirectional LSTM com 25.0% de *dropout* mostrou-se o mais promissor, alcançando a menor perda de validação de 0.8823 na época 4.

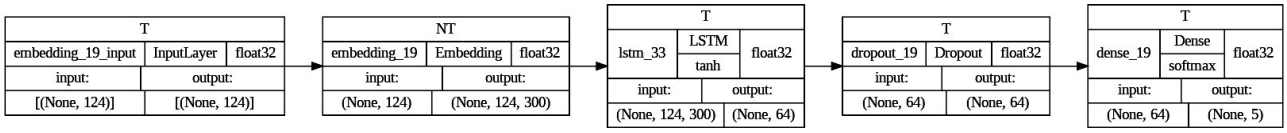


Figura 2: Arquitetura do modelo Unidirectional LSTM com 25.0% de *dropout*

A análise dos resultados sugere que a incorporação do Dropout pode ser crucial para aprimorar a robustez do modelo e a sua capacidade de generalizar além do conjunto de treinamento. Isso é especialmente relevante em tarefas de processamento de linguagem natural como a análise de sentimentos, onde a diversidade e a complexidade dos dados podem rapidamente levar a um *overfitting* sem mecanismos regulatórios adequados.

3 Resultados e discussão

3.1 Acurácias de teste

Nesta seção, apresentamos as acurácias de teste obtidas para cada configuração de modelo. Essas acurácias foram calculadas utilizando o conjunto de teste separado. A tabela a seguir resume as acurácias de teste:

Bidirectional	Dropout Rate	Accuracy
FALSE	0.00	0.606042
FALSE	0.25	0.609295
FALSE	0.50	0.607230
TRUE	0.00	0.608882
TRUE	0.25	0.607436
TRUE	0.50	0.613375

Tabela 3: resultados dos modelos sobre o conjunto de teste

As configurações dos modelos e as taxas de dropout aplicadas são refletidas na acurácia de teste. Notavelmente, o modelo Bidirectional LSTM com uma taxa de dropout de 50.0% alcançou a acurácia de teste mais alta entre todas as configurações testadas, com 61.3%.

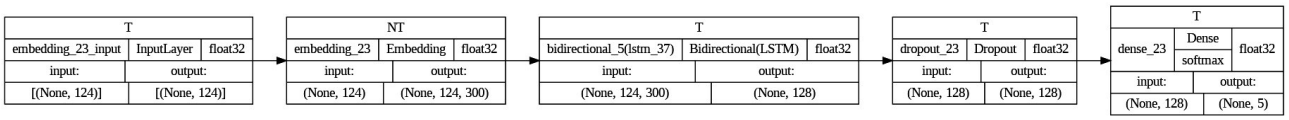


Figura 3: Arquitetura do modelo Bidirectional LSTM com uma taxa de dropout de 50.0%

Além disso, avaliamos as métricas de precisão, recall e F1 *score* para cada classe nas diferentes configurações de modelo. Essas métricas fornecem uma visão mais detalhada do desempenho do modelo em diferentes classes de avaliação. Estas métricas adicionais permitem uma análise mais granular do desempenho do modelo em classificar corretamente as avaliações em suas respectivas categorias. Observa-se uma variação nas métricas de precisão, *recall* e F1 score entre as diferentes configurações, indicando o impacto da arquitetura do modelo e da taxa de dropout na performance geral. As métricas são apresentadas na tabela 4.

3.2 Discussão

A comparação entre os resultados obtidos nos conjuntos de treinamento e validação, frente aos resultados no conjunto de teste, evidencia aspectos importantes sobre a generalização e desempenho dos modelos. O modelo Unidirectional LSTM com 25.0% de *dropout* apresentou a menor perda de validação de 0.8823 na época 4, indicando uma performance promissora durante a fase de validação.

Por outro lado, quando testado no conjunto de teste, o modelo Bidirectional LSTM com uma taxa de dropout de 50.0% emergiu como o melhor entre todas as configurações testadas, alcançando a maior acurácia de teste de 61.3%. Esta configuração demonstrou um equilíbrio eficaz entre a capacidade de generalizar bem para novos dados (não vistos durante o treinamento) e o desempenho de classificação.

Esses resultados indicam que, apesar do Unidirectional LSTM com 25.0% de *dropout* ter apresentado menor perda de validação, foi o modelo Bidirectional LSTM com 50.0% de *dropout* que mostrou uma acurácia de teste superior, refletindo uma melhor capacidade de generalização para dados não vistos. A acurácia de teste é uma métrica crucial pois reflete o desempenho do modelo em dados novos e não vistos, o que é um indicador de como será a perfomance do modelo em cenários reais.

Esta discrepância entre os resultados de validação e teste pode estar relacionada a várias fatores, incluindo a arquitetura do modelo (Unidirectional vs Bidirectional), a taxa de dropout, ou até mesmo as características dos dados nos conjuntos de validação e teste. As redes Bidirecionais, por processarem a sequência de entrada em ambas as direções, têm a vantagem de capturar dependências temporais de longo alcance, o que pode ser crucial para a análise de sentimentos.

Essa análise reforça a importância de avaliar os modelos em diferentes conjuntos de dados (treinamento, validação e teste) para obter uma compreensão mais completa sobre o desempenho do modelo e sua capacidade de generalização, que é essencial para a aplicabilidade prática em tarefas de Processamento de Linguagem Natural.

3.3 Avaliação dos Modelos de Classificação

Os testes conduzidos oferecem uma visão clara do desempenho de diferentes configurações de modelos mediante as métricas de precisão, recall e F1 score, frente às cinco classes de sentimentos. Diversas arquiteturas foram avaliadas, variando a bidirecionalidade (Bi-True ou Bi-False) e as taxas de dropout (0.0, 0.25, 0.5). O que se segue é uma avaliação dos resultados obtidos.

Metric	Bi-False_Drop-0.0	Bi-False_Drop-0.25	Bi-False_Drop-0.5	Bi-True_Drop-0.0	Bi-True_Drop-0.25	Bi-True_Drop-0.5
class_0_precision	0.760632	0.739965	0.752598	0.762115	0.751346	0.742832
class_1_precision	0.386179	0.412698	0.369146	0.373832	0.409326	0.389522
class_2_precision	0.414868	0.431019	0.443545	0.445184	0.443159	0.453699
class_3_precision	0.478836	0.477424	0.475481	0.463420	0.463960	0.474719
class_4_precision	0.639872	0.643008	0.632411	0.666875	0.634742	0.655501
macro_avg_precision	0.536077	0.540823	0.534636	0.542285	0.540507	0.543254
weighted_avg_precision	0.579449	0.579896	0.576757	0.588840	0.576964	0.585744
class_0_recall	0.870942	0.902608	0.886376	0.874667	0.890899	0.903140
class_1_recall	0.156766	0.171617	0.221122	0.231023	0.130363	0.141089
class_2_recall	0.435038	0.411148	0.390193	0.364208	0.369237	0.367561
class_3_recall	0.299112	0.297046	0.270399	0.414790	0.325759	0.384012
class_4_recall	0.807338	0.806362	0.825893	0.745954	0.809012	0.778041
macro_avg_recall	0.513839	0.517756	0.518796	0.526129	0.505054	0.514769
weighted_avg_recall	0.606042	0.609295	0.607230	0.608882	0.607436	0.613375
class_0_f1-score	0.812058	0.813234	0.814027	0.814521	0.815194	0.815180
class_1_f1-score	0.223005	0.242424	0.276574	0.285569	0.197747	0.207147
class_2_f1-score	0.424714	0.420849	0.415162	0.400645	0.402835	0.406113
class_3_f1-score	0.368214	0.366229	0.344746	0.437759	0.382767	0.424575
class_4_f1-score	0.713916	0.715479	0.716317	0.704201	0.711359	0.711534
macro_avg_f1-score	0.508381	0.511643	0.513365	0.528539	0.501980	0.512909
weighted_avg_f1-score	0.580182	0.581233	0.577763	0.595400	0.579206	0.590712
class_0_support	3758.000000	3758.000000	3758.000000	3758.000000	3758.000000	3758.000000
class_1_support	1212.000000	1212.000000	1212.000000	1212.000000	1212.000000	1212.000000
class_2_support	2386.000000	2386.000000	2386.000000	2386.000000	2386.000000	2386.000000
class_3_support	4841.000000	4841.000000	4841.000000	4841.000000	4841.000000	4841.000000
class_4_support	7168.000000	7168.000000	7168.000000	7168.000000	7168.000000	7168.000000
macro_avg_support	19365.000000	19365.000000	19365.000000	19365.000000	19365.000000	19365.000000
weighted_avg_support	19365.000000	19365.000000	19365.000000	19365.000000	19365.000000	19365.000000

Tabela 4: Resultado da performance dos modelos sobre o conjunto de testes

3.3.1 Precisão

A precisão média ponderada mostrou-se mais proeminente para o modelo Bi-True_Drop-0.5, alcançando 0.585744, seguido de perto pelo Bi-True_Drop-0.0 com 0.588840. Esta métrica é central para assegurar a minimização de falsos positivos pelo modelo. É notório que a classe 0 (presumivelmente, a classe de sentimentos muito negativos) apresentou a maior precisão em todos os modelos, sublinhando a competência dos modelos em discernir sentimentos negativos extremos.

3.3.2 Recall

Para a métrica de recall, o modelo Bi-True_Drop-0.5 se destacou ligeiramente com uma revocação de 0.613375. A classe 4 (presumivelmente, a classe de sentimentos muito positivos) teve a maior revocação em todos os modelos, demonstrando a robustez dos modelos em recuperar instâncias positivas extremas.

3.3.3 F1 score

A p média ponderada de F1 *score* atingiu seu ápice com o modelo Bi-True_Drop-0.0, marcando 0.595400, seguido pelo Bi-True_Drop-0.5 com 0.590712. Assim como nas outras métricas, as classes 0 e 4 obtiveram as maiores pontuações F1, ressaltando a eficiência dos modelos em classificar sentimentos extremos.

3.3.4 Impacto da Bidirecionalidade e Dropout:

A bidirecionalidade demonstrou incrementar a precisão, *recall* e F1 *score* em determinadas configurações, com todos os modelos bidirecionais superando seus pares unidirecionais em precisão média ponderada. A taxa de *dropout* também revelou um impacto significativo; por exemplo, uma taxa de dropout de 0.5 no modelo bidirecional resultou na maior revocação média ponderada.

3.3.5 Desempenho por Classe

A discrepância de desempenho entre as classes indica que os modelos podem enfrentar desafios ao identificar sentimentos com mais nuances (classes 1, 2, e 3). Esta constatação é corroborada pelas pontuações relativamente mais baixas de precisão, recall e F1 *score* nessas classes, em contraste com as classes de sentimentos extremos (0 e 4).

3.3.6 Suporte

O suporte para cada classe manteve-se constante em todos os modelos, sinalizando que a distribuição de classes no conjunto de teste é equilibrada, não induzindo viés no desempenho do modelo.

A escolha dos hiperparâmetros, incluindo a taxa de dropout e o tamanho máximo da sequência, também demonstrou um impacto significativo no desempenho do modelo. A taxa de dropout, em particular, mostrou-se uma ferramenta eficaz para mitigar o *overfitting*, especialmente nas configurações Bidirectional LSTM.

4 Conclusão

Este trabalho explorou a aplicação de Redes Neurais Recorrentes (RNNs), especificamente a arquitetura LSTM, na tarefa de análise de sentimentos em revisões de produtos. Através de uma investigação sistemática, analisamos o impacto de diferentes configurações e hiperparâmetros no desempenho do modelo, incluindo a utilização de camadas bidirecionais e diferentes taxas de dropout.

O modelo Bidirectional LSTM com uma taxa de dropout de 50.0% emergiu como o mais performático entre as configurações testadas, alcançando a maior acurácia de teste de 61.3%. Esta configuração demonstrou um equilíbrio eficaz entre a capacidade de generalização e desempenho de classificação.

A taxa de dropout mostrou ser uma ferramenta eficaz para mitigar o overfitting, e o ajuste do tamanho máximo da sequência também teve um impacto notável na performance do modelo.

Pode-se perceber, também, que o modelo teve dificuldades em diferenciar sentimentos com mais nuances nas classes intermediárias (1, 2 e 3), enquanto apresentou melhor desempenho em identificar avaliações extremamente negativas e positivas (classe 0 e 4).

Comparando nossos resultados com benchmarks relevantes da literatura, é evidente que ainda há espaço para melhorias. Muitos trabalhos na área de análise de sentimentos conseguem acurácias de teste superiores utilizando arquiteturas de modelo mais complexas (Transformer ou BERT) ou técnicas avançadas de pré-processamento e engenharia de características. No entanto, este estudo serve como um passo inicial importante para entender como diferentes configurações de modelo e hiperparâmetros impactam o desempenho na tarefa de classificação de sentimentos. Porém, estes resultados servem como uma base sólida para a compreensão do comportamento de RNNs em tarefas de análise de sentimentos.

Este exercício proporcionou uma oportunidade valiosa para explorar e compreender o comportamento de redes neurais recorrentes bidirecionais, especialmente no contexto de análise de sentimentos. As lições aprendidas aqui servirão como uma base sólida para investigações futuras na otimização de modelos de aprendizado profundo para tarefas de NLP.

5 Referências

BOJANOWSKI, P.; GRAVE, E.; JOULIN, A.; MIKOLOV, T. Enriching Word Vectors with Subword Information. **Transactions of the Association for Computational Linguistics**, v. 5, p. 135-146, 2017.

GRAVES, A.; MOHAMED, A.-r.; HINTON, G. Speech recognition with deep recurrent neural networks. In: 2013 IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING, 2013, Vancouver. **Anais**... Vancouver: IEEE, 2013. p. 6645-6649.

LING, W.; LUÍS, T.; MARUJO, L.; ASTUDILLO, R. F.; AMIR, S.; DYER, C.; ... & TRANCOSO, I. Finding Function in Form: **Compositional Character Models for Open Vocabulary Word Representation**. arXiv preprint arXiv:1508.02096, 2015.

MIKOLOV, T.; SUTSKEVER, I.; CHEN, K.; CORRADO, G. S.; DEAN, J. Distributed Representations of Words and Phrases and their Compositionality. In: **ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS**, 2013. Anais...

PENNINGTON, J.; SOCHER, R.; MANNING, C. Glove: Global Vectors for Word Representation. In: PROCEEDINGS OF THE 2014 CONFERENCE ON EMPIRICAL METHODS IN NATURAL LANGUAGE PROCESSING (EMNLP), 2014, Doha. **Anais**... Doha: Association for Computational Linguistics, 2014. p. 1532-1543.

SCHUSTER, M.; PALIWAL, K. K. Bidirectional recurrent neural networks. **IEEE Transactions on Signal Processing**, v. 45, n. 11, p. 2673-2681, 1997.

SRIVASTAVA, N.; HINTON, G.; KRIZHEVSKY, A.; SUTSKEVER, I.; SALAKHUTDINOV, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. **The Journal of Machine Learning Research**, v. 15, n. 1, p. 1929-1958, 2014.