

Rede neural classificadora de notícias

Gustavo A. Martini¹, Gustavo Macedo², Vinicius G. Drage³

¹Colegiado de Ciência da Computação - Universidade do Oeste do Paraná
Cascavel, PR – Brasil

²Colegiado de Ciência da Computação - Universidade do Oeste do Paraná
Cascavel, PR – Brasil

³Colegiado de Ciência da Computação - Universidade do Oeste do Paraná
Cascavel, PR – Brasil

gutamen@live.com, gustavomacedo1366@hotmail.com, vinidrage@gmail.com

Abstract. *This article describes the development and functionality of an Artificial Neural Network for classifying news based on its title, using the Python language in conjunction with the TensorFlow library.*

Resumo. *Este artigo descreve o desenvolvimento e funcionalidade de uma Rede Neural Artificial para classificação de notícias com base em seu título, utilizando a linguagem Python em conjunto com a biblioteca TensorFlow.*

1. Problema

O problema escolhido para ser resolvido é a classificação de notícias de acordo com o título da mesma. A RNA foi treinada para identificar 6 categorias de notícias, sendo elas: Política, Clima, Tecnologia, Crime, Economia e Esporte.

2. Arquitetura Utilizada

A arquitetura utilizada na RNA é uma rede densa (fully connected neural network) com uma camada de Embedding para processar os dados de texto, onde a primeira camada da rede é uma camada de Embedding que mapeia cada palavra única no vocabulário para um vetor de embedding de tamanho 80. Essa camada é responsável por representar as palavras como vetores densos de números reais, permitindo que o modelo capture a semântica e o contexto das palavras. Cada título de notícia é representado por uma sequência de vetores de embedding.

Após a camada de Embedding, há uma camada chamada GlobalMaxPooling1D que realiza o max-pooling global na sequência de vetores de embedding gerados. O max-pooling global seleciona o valor máximo em cada dimensão dos vetores de embedding, reduzindo a dimensionalidade da representação. Essa camada é útil para lidar com sequências de comprimentos variáveis, como títulos de notícias com diferentes números de palavras.

Em seguida, há uma camada de Dropout com uma taxa de dropout de 0.3, o que significa que 30% das unidades (neurônios) da camada anterior (GlobalMaxPooling1D) são aleatoriamente desativadas durante o treinamento. O dropout é uma técnica de regularização que ajuda a evitar o overfitting, melhorando a capacidade de generalização do modelo para novos dados.

A última camada da rede é uma camada densa (fully connected) com 6 neurônios, correspondendo ao número de classes diferentes nas quais os títulos de notícias são classificados. A função de ativação usada nesta camada é a softmax, que transforma as saídas da camada em uma distribuição de probabilidade sobre as classes, permitindo que o modelo faça previsões de classificação para cada categoria.

3. Algoritmo de Treinamento

Para o treinamento da RNA, foi utilizado o algoritmo de treinamento Adam (Adaptive Moment Estimation). O otimizador Adam é uma variante do gradiente descendente estocástico (SGD) que combina as vantagens do algoritmo Adagrad e do RMSprop.

O algoritmo Adam ajusta a taxa de aprendizado de cada parâmetro individualmente, adaptando-o com base nas estimativas de momento dos gradientes. Ele é conhecido por ser eficiente e rápido, além de ser capaz de se adaptar a diferentes taxas de aprendizado para diferentes parâmetros, o que é especialmente útil em problemas com superfícies de erro que têm diferentes escalas em diferentes dimensões.

4. Parâmetros Utilizados

A RNA tem como variável de entrada o título da notícia a ser classificado, e sua saída é a categoria na qual a notícia está classificada. As entradas e saídas do programa são feitas via terminal. Os dados de entrada são transformados em um vetor de vetor, onde o segundo vetor recebe cada string, já transformada em um inteiro, que representa seu token no dicionário de palavras da RNA. A saída é um ponto-flutuante que representa o peso final da previsão.

A RNA possui apenas uma camada oculta, sendo ela a GlobalMaxPooling1D, que não possui neurônios no sentido tradicional. Ao todo, a arquitetura da rede possui 80 neurônios na camada de Embedding e 6 neurônios na camada Dense (saída).

A taxa de aprendizado foi utilizada é a taxa padrão usada pelo otimizador Adam, que é ajustada automaticamente durante o treinamento. Por fim, as funções de ativação utilizadas são as seguintes:

4.1. Na camada de Embedding

Não foi especificada uma função de ativação nesta camada. Portanto, a função de ativação padrão do Keras, que é a função linear, é usada por padrão. Embora não tenha sido especificada uma função de ativação nessa camada, ela ainda realiza uma transformação linear, onde as palavras são mapeadas para vetores de embedding. Essa transformação não é afetada por funções de ativação, uma vez que a camada é responsável apenas por criar uma representação densa das palavras do vocabulário.

4.2. Na camada Dense (saída)

É usada a função de ativação softmax. A softmax é comumente usada em problemas de classificação multiclasse, pois converte as saídas da camada em uma distribuição de probabilidade sobre as classes. Isso permite que o modelo faça previsões de classificação para cada categoria.

5. Resultados e Conclusão

A RNA alcançou uma taxa de acerto alta na sua classificação, entre 94% a 88%, utilizando a mesma quantidade de épocas e calibragem, contudo, a categoria de Esportes apresentou uma taxa um pouco menor. Para contornar isso, foi aumentado a base de dados para o treino da RNA, o que resultou em um pequeno aumento nos resultados. Por fim, o grupo concluiu que os resultados da RNA foram satisfatórios para as categorias que foram definidas e funcionaram de forma eficiente na classificação das notícias.

6. Referências

TensorFlow API. https://www.tensorflow.org/api_docs/python/tf. Acesso em julho de 2023.

Rede neural densa em Tensorflow/Keras ou PyTorch. <http://www.lps.usp.br/hae/apostila/densakeras-ead.pdf>. Acesso em julho de 2023.