DATS6312 Natural Language Processing

Individual Final Project Report

Sentiment Analysis in Social Media

Yue Li

Instructor: Amir Jafari

Fall 2022

Dec 12, 2022

Introduction

Reddit is a platform for users to spontaneously discover and share content that its categories on the site are known as “subreddits”, which is like a community organized for those people who have similar interests on the subreddits, while other audiences can vote or comment to show their opinions on the submitted post. Both the negative and positive of the text evaluation provide more details and information rather than just the number of likes or dislikes. The analysis of positive and negative evaluations is the main application scenario of sentiment analysis, with the objective of identifying and classifying the sentiments in the texts varies with the situation. In this case, the text evaluation would be gathered from two main subreddits: Windows and Mac OS. Three classes of "Positive," "Negative," and "Neutral" are taken into account in the sentiments. Using the pre-trained model of Twitter Roberta base sentiment and distilling BERT base trained on IMDB data, the sentiment would be identified with the score of three emotion classes, and with the application of integrated gradient, the sentiments of the word or sequence with the importance on the predicted sentiment would be determined.

Individual work

I am fine tuning the pre-trained RoBERTa model on the “IMDB” dataset. And I explored the theory of integrated gradient and basic information of BERT, RoBERTa, and DistilBERT models.

BERT stands for Bidirectional Encoder Representations from Transformers. BERT’s state-of-the-art performance is based on two things. First, novel pre-training tasks are called Masked Language Model(MLM) and Next Sentence Prediction (NSP). Second, a lot of data and computing power to train BERT. The MLM pre-training task converts the text into tokens and uses the token representation as an input and output for the training. A random subset of the tokens (15%) is masked, i.e. hidden during the training, and the objective function is to predict the correct identities of the tokens. The NSP task allows BERT to learn relationships between sentences by predicting if the next sentence in a pair is the true next or not. (reference)

RoBERTa was introduced at Facebook, Robustly optimized BERT approach RoBERTa, is a retraining of BERT with improved training methodology, 1000% more data, and compute power. To improve the training procedure, RoBERTa removes the Next Sentence Prediction (NSP) task from BERT’s pre-training and introduces dynamic masking so that the masked token changes during the training epochs. (reference)

DistilBERT learns a distilled version of BERT, retaining 97% performance but using only half the number of parameters. The idea is that once a large neural network has been trained, its full output distributions can be approximated using a smaller network.(reference)

In this project, we selected to use a pre-trained RoBERTa-based model on a Twitter dataset and finetune a DistilBERT-based model on the IMDB dataset. The first model is named Twitter-RoBERTa-base for Sentiment Analysis (*cardiffnlp/twitter-roberta-base-sentiment*) [7]. It is trained on ~58M tweets and finetuned for sentiment analysis with the TweetEval benchmark. The

reason to select this pre-trained model is the RoBERT-based model has the best performance among the other models mentioned above and it is trained on the Twitter dataset. It is a social media data set that matches the Reddit data set in this project. The second model is the DistilBERT-based model named distilbert-base-uncased. This model is a distilled version of the BERT-based model. It is uncased which means this model is not case-sensitive. This model is smaller and faster than BERT. In this project, the IMDB data set was selected to fine-tune from the hugging face website. IMDB data set contains movie reviews and binary sentiment labels. The model has been saved in hugging face and can access with the name: *laihanel/sentiment-analysis_gwu*

Interpretability and Explainability

In the Interpretability and Explainability aspect, the team explores the integrated gradients (IG) method. In this project, a pertained transformer and fine-tuning model is used to do the sentiment analysis on Reddit submissions and comments text corpus. Our project focused on the posts under the categories of MacOS and Windows users. In these two categories, the sentimental analysis of posts is classified as positive, neutral, and negative. The sentiment analysis results came from the transformer model. The architecture of the transformer model has explained above. It is an attention neural network. It is hard to explain the results by only knowing the neural network's architecture. There are a lot of parameters connecting the input and output.

The point is how more explainability of this deep neural network model can be obtained. After getting the results from the model, the team tried to solve the problem of attributing the prediction of a deep neural network to its input features. In natural language processing, the attribution of a text sentiment network's prediction could be words or sentences. If the tokenizer is word-wise, the attribution would be the words, just like in this project. If the tokenizer is sentence-wise, the attribution would be the sentences. The importance of each word or sentence could give the output of the neural network an explanation. Therefore, the interpretation and explanation are a reductive formulation of why this prediction is useful.

The gradient of the output with respect to the input is a natural analog of the model coefficients for a deep network. Therefore the product of the gradient and feature values is a reasonable starting point for an attribution method. In the computer vision domain, we can calculate the gradient of each patch with respect to the output. A black patch is used to cover an area in the picture. Then the difference in results between those with this black patch and without this black patch shows the importance of the covered area. For example, Δx is a pixel, Δy is the change of result. Then $\frac{\Delta x}{\Delta y}$ shows the importance of this pixel. y_k is the predicted label of an image and x_n is the n-th pixel. $\frac{\partial x}{\partial y}$ is the gradient. This is a typical gradient-based method. One issue of this method is gradient saturation. For example, the Sigmoid function would be saturating when x increases. Integrated gradients can solve this gradient saturation problem.

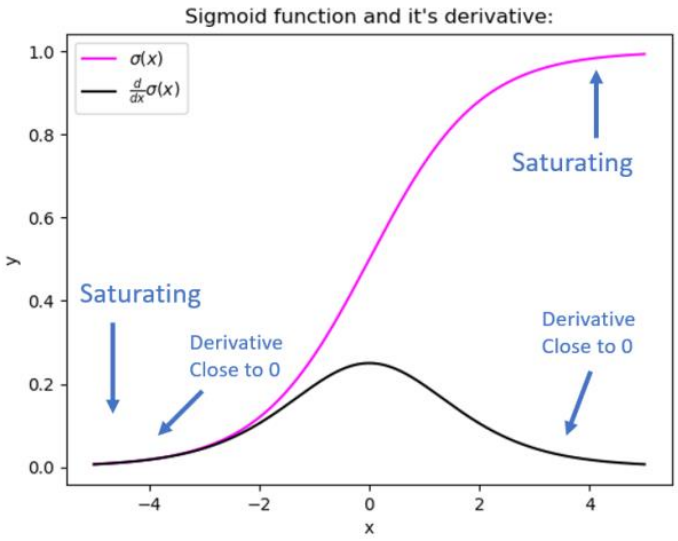

Figure 1. Example of sigmoid function and its first derivative.

The integrated gradient method was introduced to explain the neural network in a paper named Axiomatic attribution for deep networks by Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Two fundamental axioms guide the design of this method. One is sensitivity and the other is implementation invariance. Sensitivity means If for every input and baseline that differ in one feature but have different predictions, then the differing feature should be given a non-zero attribution. Implementation invariance means two networks are functionally equivalent if their outputs are equal for all inputs, despite having very different implementations.

Before implementing an integrated gradient method, it requires a baseline. The baseline is an informationless input for the model. For example, a black image for image models, empty text, or zero embedding vector for text models. Integrated gradients explain the network result of certain value input and the network result of baseline in terms of input features. The equation of the integrated gradients method is below. Instead of calculating the direct gradient of a certain point, the integrated gradients method calculates the integration of a certain input value to the baseline.

$$\begin{aligned}\phi_i^{IG}(f, \mathbf{x}, \mathbf{x}') &= \int_0^1 \frac{\delta f(\mathbf{x}' + \alpha(\mathbf{x} - \mathbf{x}'))}{\delta x_i} d\alpha (x_i - x_i') \\ &= (x_i - x_i') \int_0^1 \frac{\delta f(\mathbf{x}' + \alpha(\mathbf{x} - \mathbf{x}'))}{\delta x_i} d\alpha \end{aligned} \quad (1)$$

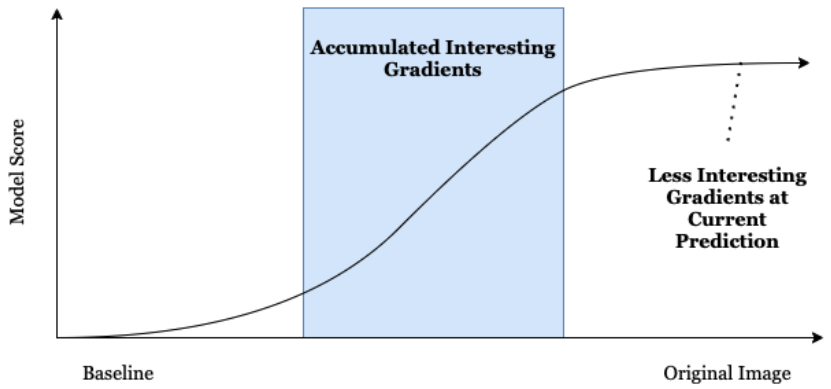

Figure 2. Example of Integrated Gradients for a Deep Learning Model.

In this project, I have learned basic structure and comparison between different models in hugging face, like BERT, RoBERTa and DistilBERT. The method about integrated gradient to explain the result of prediction. I also learned how to complete a project on the right track and schedule. In this way, I have time to explore, summarize and represent the result. It is a good experience working with two other teammates in this project.