110700022 AI HW2 Report

For Part 0:

我先將 text 都轉成小寫，然後將標點符號刪掉，之後刪去 stopwords，將每個 word 都轉成 token，然後將他們個別做 stemming，之後 join 回 text 並將換行符號刪去。

Ex: Here is a dog. -> dog

I love learning math. -> love learn math

The movie is quite good. -> movi quit good

For Part1:

Perplexity is a evaluation metric in NLP to measure how well a language model can predict a given sequence of words.It is a measure of how surprised the model is when it encounters new data that is has not seen during training.

Influenced factors such as the model architecture(LSTM,n-gram..)，training data and vocabulary size.

For Part 2:

The two pre-training steps are MLM(Masked Language Model) and NSP(Next Sentence Prediction) respectively.

MLM:

BERT takes in a sequence of words as input,randomly masking some of them. The model the learns to predict the masked words based on the context provided by the other words in the sequence. This step allows the model to learn contextualized representations of words,which can be useful for a wide range of downstream tasks.

NSP:

BERT takes in two consecutive sentences as input,and learns to predict whether the second sentence follows the first in the original text or not,It allows the model to learn the relationships between sentences,which can be useful for tasks

such as natural language inference and question answering.

The application scenario of BERT:

1. Question Answering:

   By fine-tuning BERT on a large dataset of questions and answers,the model can learn to extract the relevant information from a given passage and generate accurate answers to natural language questions.

2. Sentiment Analysis

   By fine-tuning BERT on a large dataset of labeled sentiment analysis examples,the model can learn to accurately classify text as positive,negative or neutral(doesn’t use in this HW)

3.Document Classification:

   By fine-tuning BERT on a large dataset of labeled examples,the model can learn to accurately classify documents based on their topic or content.This can be useful for a wide range of applications such as content recommendation(youtube) or targeted advertising(google ads).

4.Language Translation:

By fine-tuning BERT on a large dataset of parallel texts in different languages,the model can learn to generate accurate translations of text from one language to another.

THE DIFFERENCE BETWEEN BERT AND DISTILBERT

1. Model size:

   BERT is a larger model whereas DistilBERT is a smaller and more efficient one of BERT.DistilBERT is faster to train and use,and more suitable for applications that require less computational resources.

2. Training Time:

   BERT can take several days to train on multiple GPUs,whereas DistilBERT can be trained in a few hours on a single GPU.

3. Accuracy:

   BERT is a SOTA result on various NLP tasks,but the cost of computational resources is high,so DistilBERT is a

substitution for BERT because it has a similar accuracy and much shorter training time and computational resource.

4. Compression Technique:

DistilBERT uses knowledge distillation to compress BERT,which involves training a smaller student model(DistilBERT) to mimic the behavior of a larger teacher model by transferring its knowledge during the training process.it makes DistilBERT a smaller and more efficient model without significant loss in performance.

THE RELATION BETWEEN BERT AND THE TRANSFORMER AND THE CORE OF TRANSFORMER:

The Transformer is a deep learning model architectuture .It was originally designed for sequence-for-sequence tasks such as machine translation, but has since been applied to a wide range of NLP tasks, such as language modeling and sentiment analysis.

The core of the transformer architecture is the self-attention mechanism which allows the model to weigh the importance of different parts of the input sequence when making predictions. This is achieved by computing attention scores between all pairs of positions in the input sequence, and using these scores to weight the contribution of each position to the output.

The BERT is a pre-trained Transformer model which is trained on a large corpus of text data using MLM task and NSP task. It can be fine-tuned on a variety of downstream NLP tasks. The BERT model consists of a stack of Transformer encoder layers, which process the input sequence in a bidirectional manner to capture contextual information from both the left and right contexts. The model learns to predict the masked words in a sentence using the self-attention mechanism of the Transformer architecture.

For Part3:

Vanilla RNN and LSTM are both a RNN used for sequential data analysis. The way they differ is in their architecture and functionality.

Vanilla RNN is a basic type of RNN. It processes sequential data by feeding the previous time step's outut as an input to the current time step.It can suffer from the vanishing gradient problem,which the gradients of the loss function can become too small during backpropagation,making it difficult for the network to learn long-term dependencies in the data.

LSTM,on the other hand, is a more advanced type of RNN that addresses the vanishing gradient problem. It achieves this by incorporating a memory cell,which is a separate internal state that can store information over long periods of time. The cell uses several gate to regulate the information flow into and out of the memory cell. This architecture allows the LSTM to selectively remember or forget

information,making it more capable of learning long-term dependencies in the data.

So LSTM has a more complex architecture that includes a memory cell and several gates to address the vanishing gradient problem and enable better learning of long-term dependencies.

THE MEANING OF EACH DIMENSION

In Vanilla RNN model,

Input dimension:

Sequence length: the length of the input sequence

Batch size:the number of sequences processed in parallel.

Input size:the size of each input vector at each time step.

Output dimension:

Hidden size: the number of units in the hidden state of the RNN.

In LSTM model,

Input dimension:

Sequence length: the length of the input sequence.

Batch size:the number of sequences processed in parallel.

Input size: the size of each input vector at each time step.

Output dimension:

Hidden size: the number of units in the hidden state of the LSTM.

DISCUSSION:

Transformer model , ELMO, and attention mechanism are some of the innovations of NLP field.These new technique are all used in this HW. Transformer models are a NN-type architecture that have proven highly effective in NLP.It uses self-attention mechanism to process input sequences.It's a SOTA method on a wide range of NLP tasks.

ELMO is a deep contextualized word representation model that

uses a bi-directional LSTM to generate word embeddings that are sensitive to the context in which they are used.
Attention Mechanisms allow models to focus on specific parts of an input sequence.It is used to improve performance of machine translation and text classification.

n-gram models are a relatively easy model.It only considers sequences of n consecutive words as features. It is limited when we are dealing with long-term dependencies
Then LSTM can handle sequential data and are designed to overcome the limitations of ngram models.Long-term dependencies can be captured.
BERT is a SOTA NLP model based on transformer architecture.It uses a large corpus of text to learn contextualized representations of words.It can capture a wird range of linguistic phenomena. From n-grams to BERT , The AI researchers have conquered many problem with better solution. The appearance of ChatGPT tells everybody who is even not in the field of Computer Science that AI is going to

change our world.

DIFFICULTIES:

I met difficulties of understanding the theory of some mathematics using on bi-gram, I solved it by watching videos talking about the methods to solve it and reference some information on the Internet, it helped me a lot.

It’s quite hard to deal with models like BERT and LSTM without that much of knowledge of Machine Learning.Few classes in Machine Learning topic way more like an introduction. I am not knowing very well about the deep concept in RNN and other ways like word embedding. I am not familiar with Pytorch, either. So I spent a lot of time researching information and the definition of some professional noun and trying to construct and fine-tune BERT and LSTM models.But It can not work on colab after many trials of debugging, and also with other problems that I don’t mention. I’ll keep spending time to deal with it even after the submission of the HW until I fully understand how it works.