109550198 卜銳凱

Intro to Artificial Intelligence

Report

HW4

**1. Describe your understanding and findings about the attention mechanism by exBERT**

As stated in the paper, neural networks, such as BERT, are based on the Transformer architecture. And the transformer is based on subsequent application of “multi-head attention” to route the model reasoning. By using exBERT, we can have both the attention and the token embeddings knowledge for the user-defined model and corpus by probing whether the representations capture metadata such as linguistic features or positional information.

In this section, I will use ExBERT, a visualization tool for several variants of BERT, such as bert-base-cased and distilbert-based-uncased, to better understand BERT's attention mechanisms.

**Using BERT as Masked Language Model**

Masked Language Model (MLM) is a task whose input is a sentence with part of words being masked. The language model's goal is to predict the masked words using the context of sentence.

In this section, I will use [bert-base-cased]

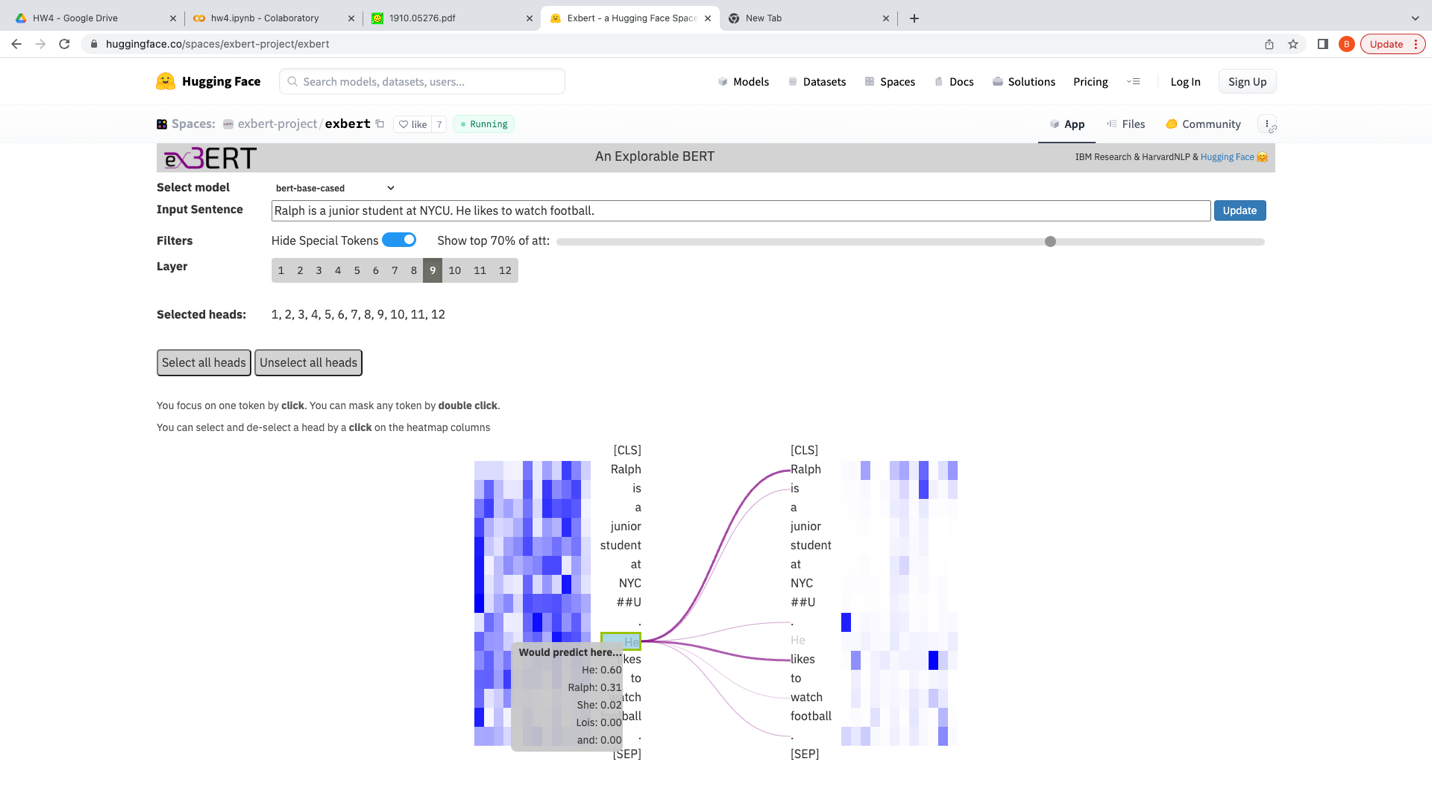
Behind the mask

A screenshot of a computer

Description automatically generated with medium confidence

As we can see from the above figure, I masked “escape” in the sentence: “The girl ran to a local pub to escape the din of her city.” I observe that BERT can precisely find the token “ran”, “to”, and “din” related to it at layer 9, the matched word summary is verb, and it can predict the masked word to be “escape” by 70% chance, which can prove the attention mechanism applied in BERT.

Behind the heads



Using the sentence “Ralph is a junior student at NYCU. He likes to watch football.” and mask the word “He”.

Another example is shown above. I mask the subject and see how BERT know what word should be filled in. As a normal human would do. BERT pay its attention about the first sentence, which is my name “Ralph”. We can also observe that “He” has the highest chance to appear in the masked position. This also shows that BERT knows Ralph is usually the name of a male.

More than position

A screenshot of a computer

Description automatically generated with medium confidence

In this example above, I wanted to see if BERT can determine whether a certain token should focus on the preceding or succeeding token. Therefore, I used the following sentence: "Ralph ran to a local pub and met his friend whose name is Sandra.” And masked “his ” at the same time. From the above figure, we can find that BERT won’t determine that the masked token is related to “Sandra”, which is incorrect. Instead, BERT can successfully predict that the masked token is related to "Ralph", indicating that it is likely to be "his" in layer 10. To summarize, we can confirm that BERT can determine whether it should focus on the preceding or subsequent token.

**[distilbert-base-uncased]**

DistilBERT is a distil version of BERT that can accomplish 97% of the original BERT's language understanding capability while also being 60% faster and 40% less in size. Following that, I will conduct three observations in the same manner as in the previous section.

Behind the mask

A screenshot of a computer

Description automatically generated with medium confidence

In this section, I masked "escape" while using the words "The girl ran to a local pub to escape the din of her city." The result is similarly comparable to BERT's in that it could additionally find that "ran", "to", and "din" have a relationship with the masked token. What's more, it can properly predict the masked token to be "escape" with a 37% possibility, which is significantly lower than BERT's. Based on the information presented above, we are certain that distilBERT may perform self-attention, although at a slightly lower percentage than BERT.

Behind the heads

A screenshot of a computer

Description automatically generated with medium confidence

The same with the last part, with the sentence” Ralph is a junior student at NYCU. He likes to watch football.” and masking "He," distilBERT will predict whether the masked token is "Ralph" or "he," revealing that it understands the context due to the self-attention mechanism. Furthermore, based on this example, distilBERT appears to perform as well as BERT.

More than position

A screenshot of a computer

Description automatically generated

In this part, I masked “his” in “Ralph ran to a local pub and met his classmate whose name is Sandra.” it turns out to be a little misunderstand the context since it predicts the masked token is “her” with only 15% possibility, while “his” has higher possibility. In conclusion, both BERT and distilBERT are provided with an attention mechanism via exBERT, and BERT clearly outperforms distilBERT when the sentence structure is difficult to understand.

**2.Compare at least 2 sentiment classification models**

In this section, I begin with two sentences:

*“It was a fantastic performance!”*

*“That is a terrible movie.”*

And I will compare two sentiment classification models given by TA by LIME.

A picture containing text, screenshot

Description automatically generated

Model 1(distilBERT: It was a fantastic performance!)

A picture containing text, screenshot

Description automatically generated

A picture containing text, screenshot, font

Description automatically generatedModel 2(smallBERT: It was a fantastic performance!)

Model 1(distilBERT: That is a terrible movie.)

A picture containing text, screenshot, font

Description automatically generated

Model 2(smallBERT: That is a terrible movie.)

According to the figure above, both distilBERT and smallBERT can successfully detect whether a sentence is positive or negative, despite the fact that the sentences are relatively simple, therefore there is no obvious distinction between these two models.

In addition, I chose three IMDB movie reviews. Two of them are positive, while one is negative.

A screenshot of a computer

Description automatically generated with low confidenceExample 1: Model 1(positive)

A screenshot of a computer

Description automatically generated with low confidence

Example 1: Model 2 (positive)

A screenshot of a computer

Description automatically generated with low confidence

Example 2: model 1(negative)

A picture containing screenshot, text

Description automatically generated

Example 2: model 2(negative)

A screenshot of a computer

Description automatically generated with low confidence

Example 3: model 1(positive)

A picture containing screenshot, text, font

Description automatically generated

Example 3: model 2(positive)

When I compared the results of the two models, I discovered that the first model outperformed the second. Since the second model must judge the more confused review, it will misinterpret the meaning of the sentence, whereas the first can understand it. Look at Example 3. We can see that model 1 correctly predicts that this review is 99% positive. Model 2 not only fails to assess if the review is positive, but also tells us that the review is 86% negative. Model 1 will, in my honest view, outperform model 2 because the latter employs prajjwal1/bert-small to tokenize and pretrain the model. prajjwal1/bert-small was created by converting a TensorFlow checkpoint from the official Google BERT repository, which was pretrained with different data, resulting in lower accuracy. As a result, the pure distilBERT (model 1) will outperform the smallBERT (model 2) on those IMDB reviews, as shown in example 3.

3.Compare the explanation of **LIME and SHAP**.

First, I'll use pure distilBERT to compare LIME and SHAP. In the final section, I'll start with basic sentences:

*“It was a fantastic performance!”*

*“That is a terrible movie.”*

A picture containing text, screenshot

Description automatically generated

LIME (It was a fantastic performance!)

A screenshot of a computer

Description automatically generated with low confidence

SHAP (It was a fantastic performance!)

A picture containing text, screenshot, font

Description automatically generated

LIME (That is a terrible movie.)

A screenshot of a computer

Description automatically generated with low confidence

SHAP (That is a terrible movie.)

We can see from the figures above that there is a slight difference between LIME and SHAP. Nonetheless, both methods might provide a credible explanation for these simple and short sentences as a whole.

A screenshot of a computer

Description automatically generated with low confidenceIn addition, I compare LIME and SHAP to IMDB reviews.

LIME - Example 1: Model 1(positive)

A picture containing screenshot, text, line

Description automatically generated

SHAP - Example 1: Model 1(positive)

A screenshot of a computer

Description automatically generated with low confidence

LIME - Example 2: Model 1(negative)A screenshot of a computer

Description automatically generated with medium confidence

SHAP - Example 2: Model 1(negative)

A screenshot of a computer

Description automatically generated with low confidence

LIME - Example 3: Model 1(positive)

A screenshot of a computer

Description automatically generated with low confidence

SHAP - Example 3: Model 1(positive)

In the case of multiple sentences, such as these IMDB reviews, we observe that SHAP can explain the entire sentence at once, whereas LIME can only explain a specific and single word each time, resulting in some errors. In example 3, which is a positive review, we can see that "not" contribute 15% to the positive prediction, whereas the whole sentence is "Some films simply should not be remade.", which appears to be a negative statement. SHAP, on the other hand, can explain the entire sentence and predict "simply should not be remade" as a negative comment, which is the second most important contribution to a negative comment. As a result, I feel SHAP can provide a better explanation than LIME.

When it comes for several sentences, SHAP outperforms LIME for two reasons. On the other hand, as previously stated, LIME can only perceive one word as a feature, whereas SHAP can select a variable number of words to be a feature, which can greatly improve its ability to explain when faced with multiple sentences that understand the meaning of paragraph by paragraph will be beneficial to its effectiveness.

On the other hand, because SHAP possesses three features: local accuracy, missingness, and consistency that LIME does not, allowing SHAP to be more consistent with human intuition and hence perform better than LIME. To confirm my argument, I obtained another explanation of example 3 from LIME, this time using smallBERT as the model.

A screenshot of a computer

Description automatically generated with low confidence

In the figure above, we observe that LIME cannot satisfy local accuracy because the prediction value changes as the model is became different. For example, "bad" accounts for 21% of negative comments in model 2, but it accounts for negative in model 1.

Last but not least, I employ smallBERT to tell the difference between LIME and SHAP. Furthermore, I pay great attention to example 3, the most confusing review.

A picture containing screenshot, text, font

Description automatically generated

LIME - Example 3: Model 2(positive)

A screenshot of a computer

Description automatically generated with low confidence

SHAP - Example 3: Model 2(positive)

SHAP, it turns out, also makes incorrect predictions. Nonetheless, SHAP can successfully determine if a sentence is positive or negative. For example, consider the first sentence. LIME will misunderstand the first word “Some” to be positive word. In the light of SHAP, it can explain the sentence “Some films just simply should not be remade.” to be negative comment, which is the right explanation.

**4.Describe how you implement other explanation techniques. And discuss with the explanation result.**

**[integrated gradients]**

A screenshot of a computer program

Description automatically generated with medium confidence

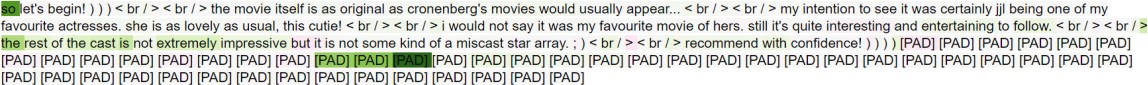
A screenshot of a computer program

Description automatically generated with low confidence

The above screenshot shows how to use integrated gradients to do sentiment analysis. Integrated gradients begin by using a baseline that consists of only the start and end tokens and no emotional words. As the step increase, it adds up the sentence by each token. In this way, it can calculate each token’s contribution to the positive or negative meaning to the sentence. Following are some examples used in Integrated gradients.



It was a fantastic performance!

IMDB review (example1)

Regarding the first, integrated gradients predicts that "fantastic" and "performance" contribute equally to positive meanings, which is worse than LIME, not to mention SHAP. In terms of larger sentences, it can successfully explain that this paragraph is overall positive, but it believes that "So" is the most significant aspect of making the sentence positive, which is incorrect and performs worse than LIME and SHAP.

**5.Try 3 different input sentences for attacks. Also, describe your findings and**

**how to prevent the attack if you retrain the model in the future**

In this part, I will apply pure distilBERT to perform some attack.

[**misspelling**]

The original sentence: “The movie is bad.”

A screenshot of a computer

Description automatically generated with low confidenceA picture containing text, screenshot

Description automatically generated

Both LIME and SHAP can explain precisely why this sentence has a 100% negative meaning, due to the word "bad." Then, I try to launch an attack by misspelling "bad" with "baaaaaaaad".

A picture containing screenshot, text, line, diagram

Description automatically generatedA picture containing text, screenshot

Description automatically generated

There is 57% confidence that the sentence has positive meaning, implying that the misspelling attack was successful.

**[double negation]**

To confuse the explainer, I launch an attack using the double negative sentence "It was a fantastic performance, which was not true."

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with low confidence

Because of the presence of "not," both explainers interpret the sentence as negative. LIME believes that the sentence has a negative sense, signifying a completely successful attack.

**[Word substitution by synonym]**

On the Internet, I found this sentence: "connoisseurs of Chinese film will be pleased to discover that Tian's meticulous talent has not withered."

A screenshot of a computer

Description automatically generated with low confidence

A screenshot of a computer

Description automatically generated with low confidence

SHAP is 90% certain that it is a positive sentence. Then I try to attack by replacing "film" with "footage".

A screenshot of a computer

Description automatically generated with low confidence

A screenshot of a phone

Description automatically generated with low confidence

the explainer then predicts the sentence to be 54% chance to positive comment. It reveals that I attack successfully by substituting the synonym.

**[prevention]**

To prevent such an attack, I can pretrain the model with more data containing double sentences, misspellings, and word substitutions with synonyms. Furthermore, I may add or remove words at random while preserving the model to resist other types of attacks as well.

**6.Describe problems you meet and how you solve them.**

I believe that the most difficult aspect of this assignment is finding an appropriate sentence to test our model and explainer, without which we will be unable to see the desired situation. For example, I intended to find a semantically equivalent sentence to give an effective attack, but I tried and failed to discover a meaningful sentence. I eventually had no choice but use the Internet example.