Evaluating XLNet on NLP tasks and interpreting layers using brain recordings

Anonymous Authors[1]

Abstract

Current natural language processing (NLP) models use deep learning to extract representations for modeling language. Deep learned representations are not easily interpreted by humans, so it is difficult to know what the best NLP models are encoding. (Toneva & Wehbe, 2019a) present a new method to interpret network representations from state-of-the-art NLP models using brain data to elucidate similarities and differences between the representations learned by these models. This involves recording fMRI and MEG data from participants reading a set of words and training a classifier to predict these data using only the representations learned by NLP models. A higher performance of this classifier indicates that more information stored in neural representations is also stored in the embeddings of the NLP model being evaluated. The researchers also modify the BERT architecture by removing its attention mechanism at certain layers, and find that the BERT models that better predict brain data after modification also perform better on a common NLP task (masked word prediction). My implementation partially reproduces the findings related to BERT, and extends this paper by visualizing how XLNet, a newer NLP model, performs on the same brain data prediction task. I also evaluate XLNet and BERT for their performance on a binary text classification task and a semantic analysis task. My results followed the trend presented in the paper in that the model that performed better on brain data prediction also performed better on all NLP tasks (BERT performed better than XLNet). However, this result was also unexpected because XLNet is a more recent model than BERT, generally outperforming BERT on most NLP tasks in practice.

1. Paper 1: (Toneva & Wehbe, 2019a)

1.1. Summary

Current natural language processing (NLP) models are implemented without rule encoding, and still perform well on natural language tasks. However, the representations encoded by the best models are not well understood, bottlenecking development of future models. (Toneva & Wehbe, 2019a) present a new method to interpret network representations from state-of-the-art NLP models including ELMo, USE, BERT and Transformer-XL, alongside brain data to elucidate similarities and differences between the representations learned by these models. The proposed approach involves recording fMRI and MEG data from participants reading a set of words and subsequently training a classifier to predict these data using only the representations learned by NLP models. These results imply that brain data can help elucidate powerful representations in NLP models, leading to a more robust understanding of what makes a good model, and can potentially drive development of better models through fine-tuning on brain data.

1.2. Critique

1.2.1. Strengths

The paper develops a strong technique for mapping the results of word embeddings from NLP models to brain data. The approach makes sense. Training a classifier to be able to predict brain recordings solely from layer representations of natural language models poses a strong test to the information contained in these models. If the same information were not truly contained in these embeddings, then the models would be falsified. As for which brain region to consider when taking brain recordings, the researchers utilized priors (in this case, knowledge of various brain areas) that were well-studied, and justified their use of these priors.

1.2.2. Weaknesses

The paper states that the fMRI data was used to show that different network architectures (ELMo, USE, BERT and Transformer-XL) encode information at different context lengths. However, I am curious to know why the fMRI data better supported this conclusion rather than the MEG data, given that fMRI data depends on blood flow in the brain, and can therefore only elucidate events with the precision of hundreds of milliseconds. MEG, on the other hand, is very precise and has a resolution of fewer than 10 milliseconds. In brain research, MEG is primarily used for measuring the time course of events. I would like to ask the authors what difficulties were presented when trying to perform this analysis, as they mention this in their future work section of the paper. Additionally, the researchers utilize four well-known models of natural language processing, however, they fail to analyze some of the state-of-the-art models. I would be curious to see what the results for something like XL-Net, a hybrid between Transformer-XL and BERT, would look like.

2. Paper 2: (Schwartz et al., 2019)

2.1. Summary

Taking the research in (Toneva & Wehbe, 2019a) a step further, (Schwartz et al., 2019) from the same research group fine tune the NLP models using the brain data in (Toneva & Wehbe, 2019a) to create stronger representations of language in these models. After fine-tuning, it is no surprise that BERT was found to better predict neural recordings. They also find that the predictions of brain activity by mappings from BERT are consistent across multiple participants tested in their study. They find that representations learned after fine-tuning the models on fMRI data and MEG data together better predict MRI data, which shows with higher probability that the learned representations capture at least some of the same data as brain representations. These changes to the BERT model also do not harm NLP task performance after fine-tuning.

2.2. Critique

2.2.1. Strengths

The paper's hypothesis is that while information may not be structured in the ideal format from the encoding of a language model initially, fine-tuning it will restructure the representations learned into a more realistic format with regards to how the brain stores this information. They provide evidence for this hypothesis by actually showing that fine-tuning the model does improve prediction accuracy, indicating that the process of fine-tuning changed something about the model's internal representation. The researchers make use of a "vanilla model" as a control. The technique is flexible because it can be modified to work for prediction tasks of different sizes and resolutions, and the authors essentially demonstrate the feasibility of doing this - in particular, "biasing" language models to learn the relevant relationships. The researchers also show that the model is not simply learning a relationship between the text and fMRI, or text and MEG (modalities of brain recording), but rather the information contained in the brain data itself. NLP tasks not being harmed is a strong conclusion that shows that the models are not changing their internal information.

2.2.2. Weaknesses

Nine participants seems like a small sample size, so extending this study to include data from more participants seems like a direction of future work. Additionally, it is not clear why text from "Harry Potter and the Sorcerer's Stone" was chosen. Perhaps it doesn't matter as long as the text chosen is held constant across all trials. I wonder if changing the text to something more simplistic or more complex would change the experimental results. I would like to ask the authors why they chose BERT as it is not explicitly mentioned. The experimental approach taken was interesting, as the authors presented the chapter to the participants one word at a time, each word presented on the screen for half a second, rather than allowing the participant to read the chapter at their own pace. My conjecture is that the data would not be captured at a high resolution at a faster pace; however, I would like to ask the authors whether this pace would interfere with the participant's ability to easily remember the rest of the earlier sentence and be able comprehend what they were reading rather than losing focus.

3. Paper 3: (Schwartz et al., 2019)

3.1. Summary

Finally, (Wang et al., 2019) utilize convolutional neural networks (CNNs) to predict neural data. Unlike the work done by (Toneva & Wehbe, 2019a) and (Schwartz et al., 2019), this research is focused on visual representations. The researchers' work is based on the following assumption: "If a feature is a good predictor of a specific brain region, information about that feature is likely encoded in that region." They train a model on learned representations of CNNs on 21 vision tasks to predict brain responses on a large scale fMRI dataset, BOLD5000. Based on the task, the researchers predict brain activity in different regions of the brain and elucidate the task-specific architecture of the brain's vision system. Unlike the previous studies, these researchers make use of an open source fMRI dataset which seems to be more expansive, given that observers viewed over 5000 stimuli consisting of natural and object images. The 21 computer vision tasks used represent both 3D and 2D tasks. Finally, using the data from which task predicted which brain area best, the researchers constructed a "task map", which helps provide evidence for the authors' hypothesis that tasks with more transferability will make similar predictions across different brain regions.

3.2. Critique

3.2.1. Strengths

The authors make use of a ridge regression mapping from neuronal activations in computer vision models to brain data rather than a lasso regression or more complicated models to maintain interpretability of the model weights. To choose the regularization parameter which is variable when utilizing a ridge regression model, the researchers make use of a cross-validation strategy. They utilize both Pearson's coefficient and R^2 as statistical measures of model performance, and also shuffled responses 5000 times. Overall, the statistical approach to analysis in this paper is very strong. The findings are very clear as well, and one of the main findings is that the 3D tasks are able to predict a specific part of the visual cortex.

3.2.2. Weaknesses

The researchers use the BOLD5000 dataset which includes data from 3 participants viewing around 5000 stimuli. As compared to (Schwartz et al., 2019), who used 9 participants and perhaps less data (it is not clear the size of the corpus of data between the natural language task and vision task if these can be compared), I would be curious to ask either of the authors which is more important - varying the number of participants for which data is collected or increasing the diversity and quantity of data which is collected for each of the participants. It seems that both approaches would provide a control in one sense. By controlling for participants, this removes some of the noise caused by introducing different participants, but also introduces a doubt about whether the results are generalizable beyond these participants. Having more data and more diverse data is generally favorable, however, perhaps it is not scalable to increase the number of participants while maintaining a large dataset. I would like to ask the authors why they chose a p-value of 0.05 rather than something more strict like 0.01.

4. Implementation, Evaluation, and Discussion

4.1. Overview

My implementation has two parts: first, I seek to reproduce the findings of (Toneva & Wehbe, 2019a) related to BERT, and extend this paper by visualizing how XLNet, a newer NLP model, predicts the same brain recordings. Second, I also investigate how XLNet and BERT fare on common NLP tasks such as binary text classification and sentiment analysis. According to the paper's results, the model that performs better on predicting brain data should also perform well on natural language processing tasks.

Existing code contains methods to extract NLP features from the 4 given models (ELMo, USE, BERT and Transformer-XL), build an encoding model that takes these NLP representations and predicts brain recordings of 8 subjects reading the same text, and finally, evaluate the predictions using a classification task presented in the paper. I modified the existing source code to do the same for XLNet, which involved a few small modifications.

For the NLP task experiments, an existing script trains and tests BERT for text classification (Cheng, 2020). I modify the script to perform the same for XLNet, and also extend this script to do semantic analysis by integrating a new dataset. I perfom experiments for semantic analysis for both BERT and XLNet. I find that BERT performs better for predicting brain recordings for all layers. XLNet's middle layers(approximately layers 7-12), however, begin to match performance of BERT's early layers (layers 1-4). For NLP task peformance, BERT has an accuracy of 94% on the binary classification task, while XLNet has an 82% accuracy on the same task. For semantic analysis, BERT performs with an accuracy of 74% and XLNet 69%.

4.2. NLP model descriptions

4.2.1. BERT: Bidirectional Encoder Representations from Transformers

BERT is a popular NLP model developed by Google in 2018 that makes use of Transformers. Transformers include an encoder and decoder to produce outputs for input text. BERT's unique contribution to the world of NLP is that it applies bidirectional Transformer training to NLP tasks. Before BERT, models looked at sequences of text from left-to-right, or vice versa, or combined both of these. HuggingFace (cite) provides a pretrained model of BERT containing 12 layers, a hidden layer size of 768, and trained on Wikipedia.

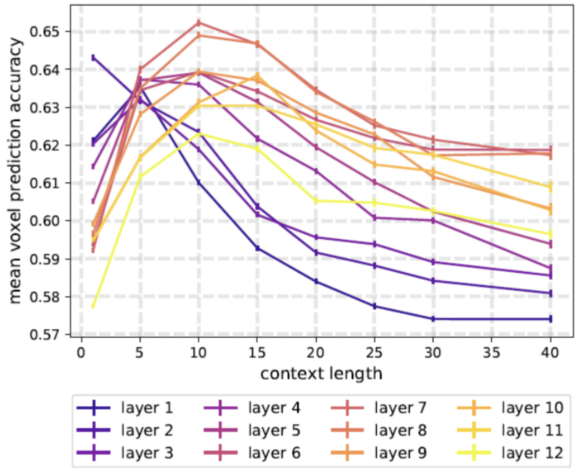
4.2.2. XLNet

XLNet is an autoregressive model, meaning that it predicts future behavior using past behavior. XLNet is more recent than BERT, co-developed by Google and CMU researchers in 2019, and building upon ideas from both BERT and Transformer-XL, another popular NLP model. Given its recency and state-of-the-art status on many NLP tasks, XLNet would be expected to perform better than BERT on NLP tasks if not predicting brain data. XLNet also performs well on long-context tasks, so I hypothesize a higher accuracy in predicting brain data of long sequences (30+ words), as compared to short sequences of 1, 5 or 10 words. XLNet has 25 layers and a hidden layer size of 1024.

4.3. Existing code and changes

4.3.1. Predicting brain data

(Toneva & Wehbe, 2019a) provide a repository of code to perform the method proposed in their paper on 4 NLP models: Transformer-XL, BERT, ELMO, and USE. This method to evaluate performance of a given NLP model on predicting brain recordings with respect to the same text (a chapter of Harry Potter text) is composed of 3 steps (Toneva & We-



(b) BERT

Figure 1. BERT results for predicting fMRI data, Figure 4(b) in (Toneva & Wehbe, 2019a)

hbe, 2019b): 1) Representations of the Harry Potter text are extracted from the given NLP model. 2) An encoding model is built to relate these extracted representations to the brain recordings of people reading the same text. Ridge regression is used as the linear mapping model, which is chosen for its computational efficiency and previous results showing that different regularization techniques (such as lasso regression) lead to similar results for predicting fMRI data. 3) Finally, the method evaluates the predictions of the encoding model using a complex classification task using 3D brain folding data.

The graph shown in Figure 1 is what I attempted to reproduce. This figure shows the classification accuracies of each of BERT's layers for context lengths ranging from 1 to 40, averaged over all participant data. It is important to note that there were data of 8 participants utilized to make the graph shown in the original paper, but due to computational constraints, I reproduced the graph using only one subject's data (subject F), which is shown in 6.

The method is generalizable to additional NLP models with a few modifications. Each model's features is extracted and averaged in a slightly different way to accomplish step 1. The models are obtained from PyTorch, AllenNLP and other sources, pretrained on a corpus of text such as Wikipedia. To introduce XLNet into this repository and create the functionality for it, I followed the steps to extract features from Transformer-XL. The changes made were the model loading, the feature extraction step, the number of layers to extract features for, and syntax changes throughout the script used to extract features from Transformer-XL. Overall, since the paper's method is generalizible and modular, there were minimal changes made to this script, and steps 2 and 3 were treated as a black box to extract the final predictions.

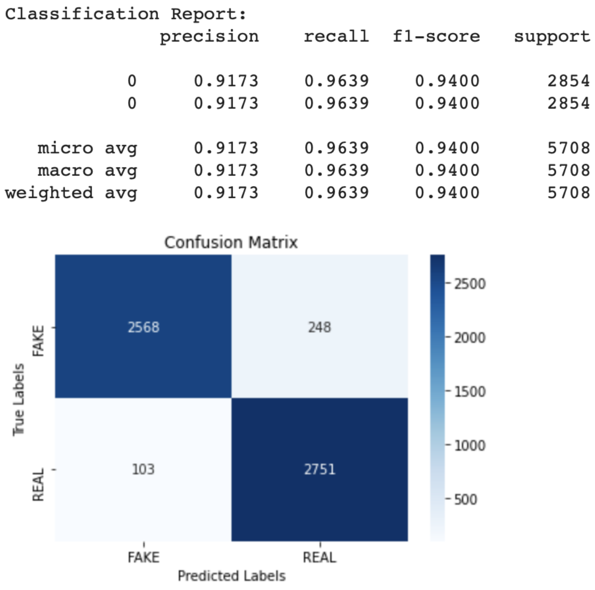
To plot the figures, I wrote additional scripts to run steps 1, 2 and 3 for subject F's data, for each NLP model (BERT and XLNet) on a remote server, which ran for around 2 days. Finally, I wrote one more script to average the results for each layer for each sequence length, to plot the data show in in 6. None of the plotting logic was included in the repository. I assumed that for each layer and each sequence length, the average prediction accuracies were averaged together to produce a single number on the plot shown.

4.3.2. Text Classification

To see how BERT and XLNet perform on text classification, a text classification task was performed separately. I ran an existing script to fine-tune BERT on a binary classification task on a fake news data set (Cheng, 2020) to obtain the accuracy of BERT for the binary classification task of classifying real news vs. fake news. The news dataset contains large sequences of text as the training input, and a label, either "FAKE", or "REAL" as the output. I then introduced XLNet into the script in order to compare the accuracy of XLNet on the same binary classification task. This involved changing the tokenizer involved to tokenize the raw text, as XLNet and BERT both have different tokenization schemes.

Then, I modified the script to run a multiclass classification task. This time, the input data was the content of a tweet, and the output was one of 5 categories: Extremely Positive, Positive, Neutral, Negative, and Extremely Negative, to classify the sentiment of the tweet. These tweets were obtained from a Kaggle data set of COVID-19 related tweets (Miglani, 2020). Since tweets include special characters such as the "@" symbol for Twitter handles, hashtags, and emoticons, and irrelevant data such as links that are not helpful when doing sentiment analysis, I cleaned the data through a series of cleaning steps to make the text more meaningful before performing sentiment analysis.

I changed the architecture of the BERT and XLNet models to have an output size of 5 classes to accomodate this new task, and I also modified the loss function from Binary Cross Entropy Loss to Cross Entropy loss for multiclass classification. Since this data set is much bigger than the News dataset for the binary classification task, consisting of more than 50,000 instances of data as compared to just over 600, I fine-tuned both models on a small subset of this dataset (12,225 instances of data) due to computational constraints.



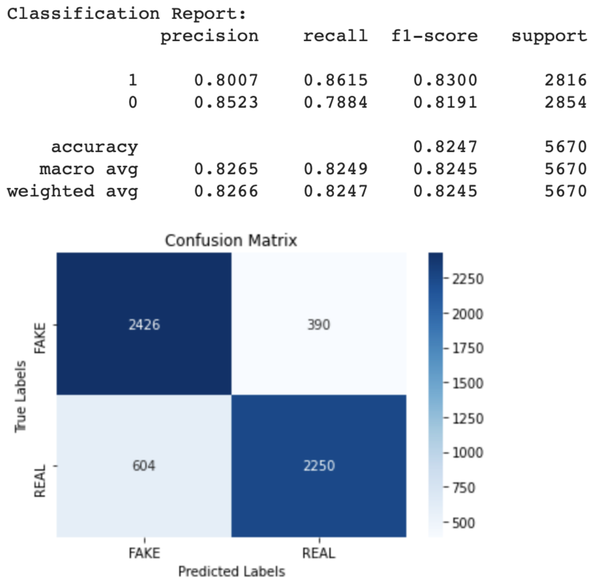
```
Classification Report:
              precision    recall  f1-score   support

           0     0.9173    0.9639    0.9400      2854
           0     0.9173    0.9639    0.9400      2854

   micro avg     0.9173    0.9639    0.9400      5708
   macro avg     0.9173    0.9639    0.9400      5708
weighted avg     0.9173    0.9639    0.9400      5708
```

Figure 2. Results of BERT on binary classification task.

4.4. Results

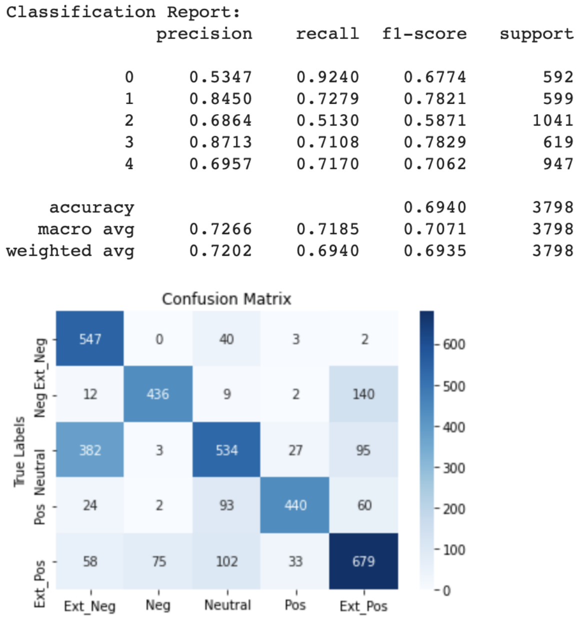
Figure 2 shows the results of BERT's performance on the binary classification task, and figure 3 shows the results of XLNet's performance on the same. Figure 4 shows the results of XLNet's performance on the semantic analysis task, and figure 5 shows the results of BERT's performance on the semantic analysis task. On the binary classification task, it can be seen that BERT outperforms XLNet (94% accuracy compared to 82% accuracy). On the multiclass classification task, it is clear that BERT outperforms XLNet as well (74% accuracy compared to 69% accuracy). The precision, recall, and f1-scores are also shown in the figures.

Figures 6 and 7 show the results of using one subject's data to produce the layer prediction accuracies for BERT and XLNet, respectively. Figure 6 looks different than figure 4(b) from the original paper, which is likely due to lack of averaging over data from all subjects. It is interesting to note that prediction accuracy increases significantly with context length for later layers in my graph, which could be a characteristic of subject F's brain data. BERT's accuracy for predicting fMRI data does not go below 54%, while XLNet's accuracy in the later layers drops to below 50%. However, in the middle layers (approximately layers 7-12), XLNet shows an increase in predictivity. The trend for XLNet seems to be low predictivity in earlier layers, higher predictivity in middle layers, and a steep drop in predictivity in later layers. Context length does not seem to affect this accuracy, although I do notice there is a oscillating pattern in accuracy as context length increases.



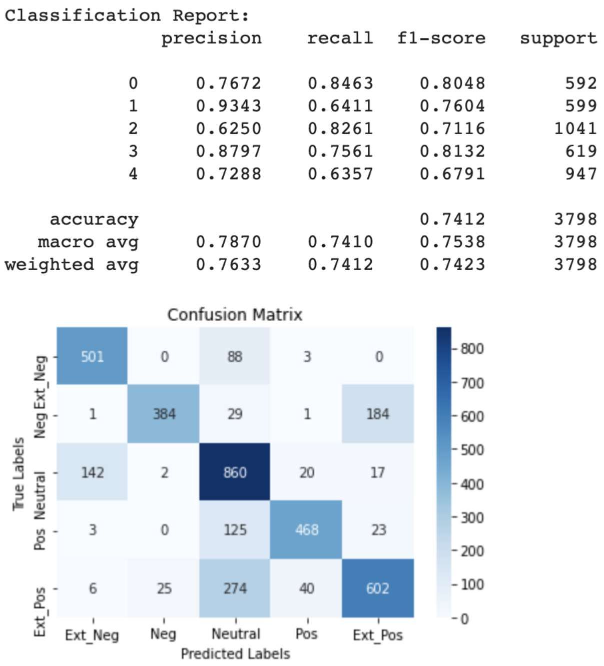
```
Classification Report:
              precision    recall  f1-score   support

           1     0.8007    0.8615    0.8300      2816
           0     0.8523    0.7884    0.8191      2854

    accuracy                         0.8247      5670
   macro avg     0.8265    0.8249    0.8245      5670
weighted avg     0.8266    0.8247    0.8245      5670
```

Figure 3. Results of XLNet on binary classification task.



```
Classification Report:
              precision    recall  f1-score   support

           0     0.5347    0.9240    0.6774       592
           1     0.8450    0.7279    0.7821       599
           2     0.6864    0.5130    0.5871      1041
           3     0.8713    0.7108    0.7829       619
           4     0.6957    0.7170    0.7062       947

    accuracy                         0.6940      3798
   macro avg     0.7266    0.7185    0.7071      3798
weighted avg     0.7202    0.6940    0.6935      3798
```

Figure 4. Results of XLNet on semantic analysis task.



```
Classification Report:
              precision    recall  f1-score   support

           0     0.7672    0.8463    0.8048       592
           1     0.9343    0.6411    0.7604       599
           2     0.6250    0.8261    0.7116      1041
           3     0.8797    0.7561    0.8132       619
           4     0.7288    0.6357    0.6791       947

    accuracy                         0.7412      3798
   macro avg     0.7870    0.7410    0.7538      3798
weighted avg     0.7633    0.7412    0.7423      3798
```

Confusion Matrix

True Labels \ Predicted Labels	Ext_Neg	Neg	Neutral	Pos	Ext_Pos
Ext_Neg	501	0	88	3	0
Neg	1	384	29	1	184
Neutral	142	2	860	20	17
Pos	3	0	125	468	23
Ext_Pos	6	25	274	40	602

Figure 5. Results of BERT on semantic analysis task.

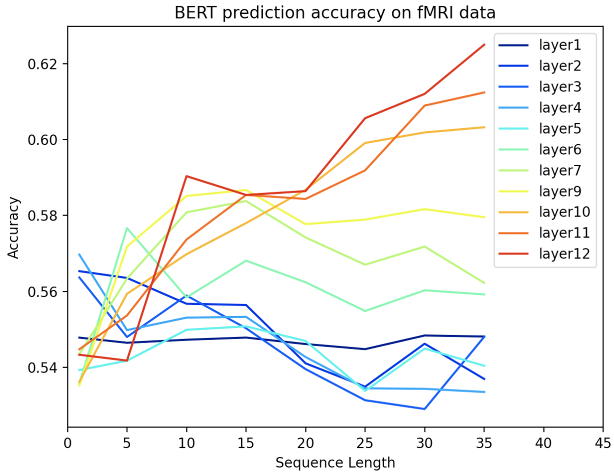


Figure 6. My attempt to reproduce the results shown in figure 1, but with one subject's data.

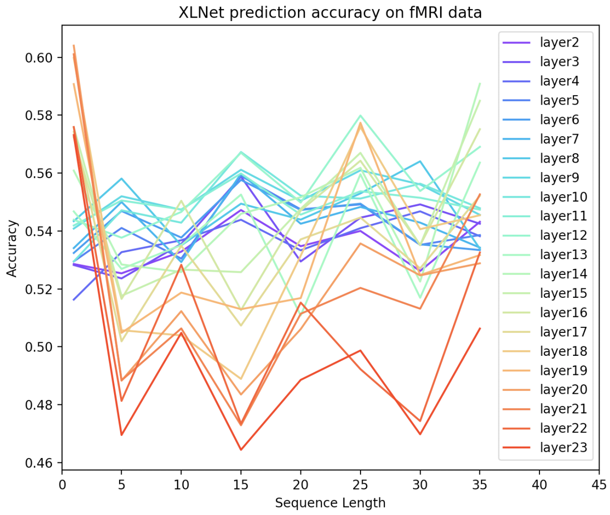


Figure 7. My attempt to reproduce the results shown in figure 1, but using XLNet, and on one subject's data.

4.5. Discussion and Future Work

It is not clear why XLNet does not outperform BERT on the NLP tasks tested here. It is expected that XLNet should perform better because XLNet has been shown to outperform BERT on many NLP benchmarks. One possibility is that I neglected some data processing steps that are commonly done to prepare XLNet for downstream training on the data, or used a loss function that is not commonly used. However, I am currently not aware of any particular protocols involved when training XLNet. Perhaps the text sequences used for training were too short (tweets are limited to 280 characters). As for predicting brain data, it is interesting that XLNet does not perform significantly better on predicting brain data for longer contexts, as XLNet is trained on long sequences of data as compared to BERT. Later layers seem to perform quite well on predicting sequences of length 1 in XLNet.

Future work should investigate the same binary classification and semantic analysis tasks on significantly longer sequences of data for both BERT and XLNet to falsify these results. Additionally, future work should average results of predictions for all 8 subjects to generate more accurate versions of figures 6 and 7. Future work should also investigate the apparent oscillatory pattern observed in XLNet prediction accuracy on brain data as context length increases.

References

Cheng, R. Bert text classification using pytorch. https://towardsdatascience.com/

bert-text-classification-using-pytorch-723dfb8b6b5b/, 2020.

Miglani, A. Coronavirus tweets nlp - text classification. https://www.kaggle.com/datatattle/covid-19-nlp-text-classification/, 2020.

Schwartz, D., Toneva, M., and Wehbe, L. Inducing brain-relevant bias in natural language processing models. In Wallach, H., Larochelle, H., Beygelzimer, A., dAlché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 14123–14133. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/9559-inducing-brain-relevant-bias-in-natural-language-processing-models.pdf.

Toneva, M. and Wehbe, L. Interpreting and improving natural-language processing (in machines) with natural language-processing (in the brain). In Wallach, H., Larochelle, H., Beygelzimer, A., dAlché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 14954–14964. Curran Associates, Inc., 2019a. URL http://papers.nips.cc/paper/9633-interpreting-and-improving-natural-language-processing-in-machines-with-natural-langua pdf.

Toneva, M. and Wehbe, L. Interpreting and improving natural-language processing (in machines) with natural language-processing (in the brain). In *Advances in Neural Information Processing Systems*, pp. 14954–14964, 2019b.

Wang, A., Tarr, M., and Wehbe, L. Neural taskonomy: Inferring the similarity of task-derived representations from brain activity. In Wallach, H., Larochelle, H., Beygelzimer, A., dAlché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 15501–15511. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/9683-neural-taskonomy-inferring-the-/similarity-of-task-derived/-representations-from-brain-activity.pdf.