Advanced Deep Learning Homework 4

Abbas Nosrat
810199294

June 12, 2022

Part I

Interpretability & Fairness in DistilBERT

1 Training DistilBERT

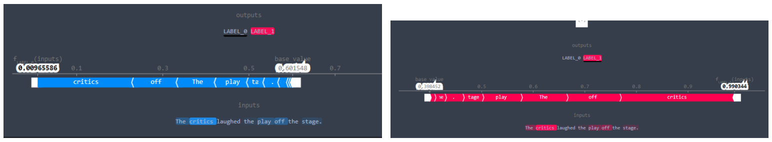
- The model was trained with a notebook provided by Huggingface website. The notebook used the pretrained Bert tokenizer to tokenize the sentences and provide attention masks. This tokenizer took the sentence, maximum length and the option to use padding for each sentence. After tokenazation, the Bert model took the tokenized sentences and their attention mask as input and generated p-values for each word in the sentence. The model has been trained for 5 epochs with crossentropy loss and matthews correlation as metric.

- There is a plethora of text augmentation techniques. These techniques can be divided into three levels. The lowest level is word level. Techniques in word level involve changing random letters in randomly selected words in a text. The second level is sentence level. In this level, words in a sentence can be omitted, substituted with their synonym or antonym or the word order can be scrambled. The third level is text level or paragraph level. One of the techniques in this level is re-translation. This technique, utilizes a translation engine to translate the text to another language and translate it back to the parent language. There are a few repercussions that must be taken for using some of these methods. For example, using antonym substitution, would result in changing the context of the sentence and would turn a positive sentence into a negative sentence. The label must also be changed should this context change occurs.

- The code could not evaluate accuracy so instead, matthews correlation is reported. Training metric is 0.538760 and validation metric is 0.5387602599113016.

2 Using SHAP and LIME

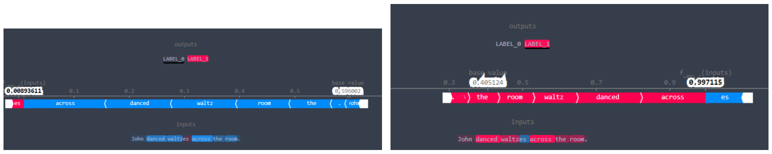
- The following is a brief explanation of SHAP and LIME:
  - SHAP comes from game theory. Basically, SHAP computes the level of contribution of each feature for the given prize. In context of classification, the prize is the decision made by the model. To evaluate the level of contribution (also known as Shaply values), the algorithm evaluates shaply values for a feature in all possible subsets of features involving that feature. As an example, lets choose a subset of three features. all features aside the three chosen features are replaced with random values. This substitution is due to the fact that random values have no prediction power. Assuming the three selected features are called A,B and C and we would like to evaluate Shaply value of feature A. First model predicts with A, B and C and then A is replaced with a random value and the model runs again. Using the p-values provided from both experiments, SHAP evaluates Shaply value of feature A. Shap is repeated for all feature sets involving A and then the total Shaply value for A is an average of shaply values for all subsets. A great limitation of SHAP, is its computational complexity which is 2^n where n is the number of features.
  - LIME stands for Local Interpretatble Model-Agnostic Explanation. As the name suggests, This algorithm fits a locally linear model at the decision boundary which is explainable and uses the linear model to provide contribution of each feature based on p-values provided by the linear model. The problem with LIME is that although the explanations may make sense locally, they may not be sensible globally. One advantage of LIME over SHAP is the lower computational cost.
- The following are outputs of SHAP and LIME for three randomly given examples.



(a) First label

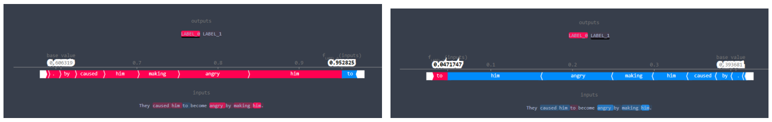
(b) second label

Figure 1: SHAP for the first sentence



(a) First label (b) second label

Figure 2: SHAP for the second sentence



(a) First label (b) second label

Figure 3: SHAP for the third sentence

- It can be interpreted the model has learned grammar from its attention to words and expressions such as **to, es** or pronouns and verbs placement. For example, in Figure 2, the contribution of the expression **es** made to classifying the sentence as incorrect is higher than other words.

3 Fairness

- The model associates certain jobs based on gender with a high bias which quit sexist and unfair.
- Performing sentence level text augmentations such as replacing every male name or pronoun with a female, could remedy the problem. If the model learns that a man can work as a nurse or a woman as a carpenter for example, it would no longer associate those jobs with a certain gender. However, statistically speaking, some jobs are indeed more associated with certain genders and the aforementioned augmentation would reduce model power. For example, the number of women working as carpenters is insignificant and this fact is reflected in the dataset. The model can exploit this fact to preform with more accuracy. All in all, such augmentations must not be preformed with a heavy hand.
- Some metrics used to quantify model fairness include:
  - **Disparate Impact**: This is the ratio of probability of favorable outcomes between the unprivileged and privileged groups.
  - **Average odds difference**: This is the average of difference in false positive rates and true positive rates between unprivileged and privileged groups. A value of 0 implies both groups have equal benefit.

- **Average odds difference**: This is the average of difference in false positive rates and true positive rates between unprivileged and privileged groups. A value of 0 implies both groups have equal benefit.

Using such metrics along with a statistical analysis of the dataset regarding the distribution of minority groups and awareness of the model towards the unprivileged groups would provide a good understanding of model fairness.

Part II

Adversarial Attacks

- The following figure, demonstrates the predicted values and gradients of images:

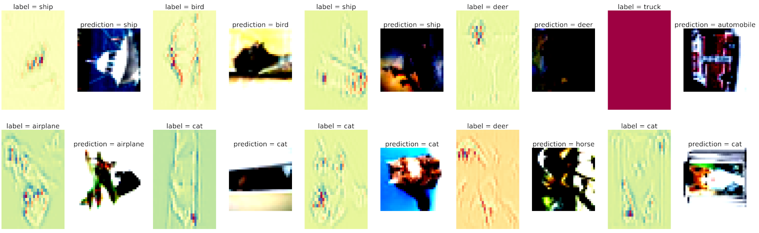
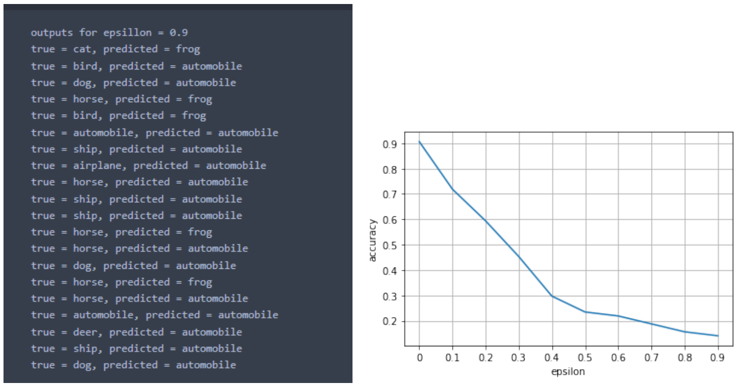


Figure 4: predicted classes with image gradients



```
outputs for epsillon = 0.9
true = cat, predicted = frog
true = bird, predicted = automobile
true = dog, predicted = automobile
true = horse, predicted = frog
true = bird, predicted = frog
true = automobile, predicted = automobile
true = ship, predicted = automobile
true = airplane, predicted = automobile
true = horse, predicted = automobile
true = ship, predicted = automobile
true = ship, predicted = automobile
true = horse, predicted = frog
true = horse, predicted = automobile
true = dog, predicted = automobile
true = horse, predicted = frog
true = horse, predicted = automobile
true = automobile, predicted = automobile
true = deer, predicted = automobile
true = ship, predicted = automobile
true = dog, predicted = automobile
```

(a) preditions

(b) accuracy per epsillon

Figure 5: Results of the attack

- From the figure above, it can be inferred that the attack pushes the images towards automobile class more than others. (Attacks were preformed with torchattack library)

- There are three main methods of defense against this attack. Data augmentation with adversarial examples, Attack detection with PCA and loss modification. Due to ease of implementation, the first method was chosen. For each batch, random samples were chosen and tainted with attack and the model trained as normal with the modified data.

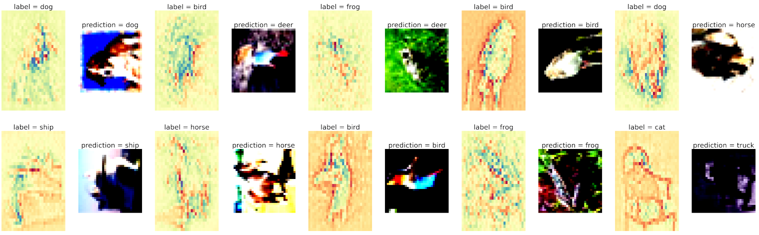


Figure 6: Gradients post attack

-

- As demonstrated above, the gradients have more clarity than their previous visualization. This means that the model has trained to pay more attention to the gradients.

- The defense method was chosen for its ease of implementation. However, the newly added perturbation made training more difficult and unstable for the model. with incomplete training, it can be claimed that this method has success to some degree.

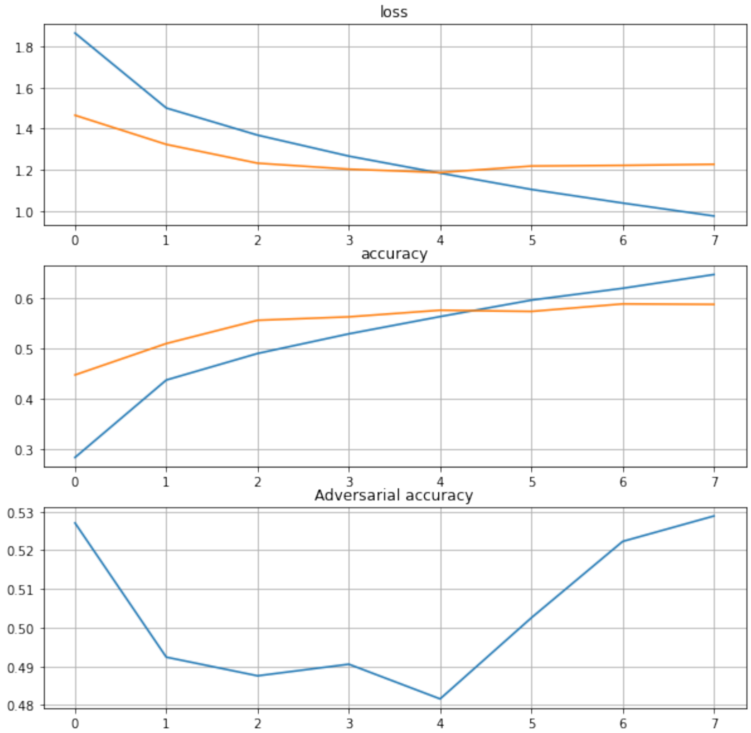
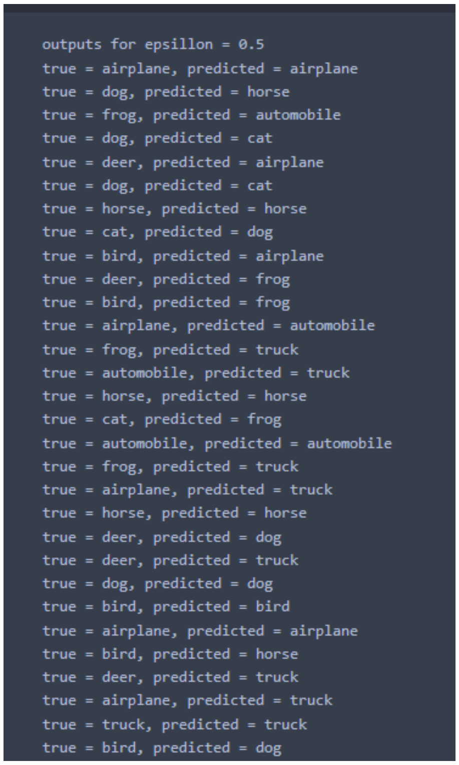


Figure 7: Training logs for adversarial training



```
outputs for epsillon = 0.5
true = airplane, predicted = airplane
true = dog, predicted = horse
true = frog, predicted = automobile
true = dog, predicted = cat
true = deer, predicted = airplane
true = dog, predicted = cat
true = horse, predicted = horse
true = cat, predicted = dog
true = bird, predicted = airplane
true = deer, predicted = frog
true = bird, predicted = frog
true = airplane, predicted = automobile
true = frog, predicted = truck
true = automobile, predicted = truck
true = horse, predicted = horse
true = cat, predicted = frog
true = automobile, predicted = automobile
true = frog, predicted = truck
true = airplane, predicted = truck
true = horse, predicted = horse
true = deer, predicted = dog
true = deer, predicted = truck
true = dog, predicted = dog
true = bird, predicted = bird
true = airplane, predicted = airplane
true = bird, predicted = horse
true = deer, predicted = truck
true = airplane, predicted = truck
true = truck, predicted = truck
true = bird, predicted = dog
```

Figure 8: predictions for tainted data after adversarial training

- The image was attached to the computational graph, fed to the model, the model gradients were set to zero and the gradients were evaluated with respect to images compered to labels. The gradients were converted to gray-scale and visualized with a heatmap. This method was the easiest to use hence it was chosen.