Replication of DistilBERT evaluation results

Group M - Nadun Chandrabahu, Tai Ho, Muhammad Umer Bashir, Thanushreyas Appaji

November 1, 2024

1 Introduction

Reproducible research is vital in machine learning for several reasons. Firstly, it enables the validation of results, when research can be consistently reproduced, it enhances the credibility of the original findings. Additionally, reproducibility supports benchmarking for comparing algorithms and models, allowing the machine learning community to identify the best-performing models. It also plays an important role in identifying errors and biases, such as coding mistakes or data handling issues, improving the reliability of the research.

The paper that we selected for this project, (Sanh et al, 2019)[1] "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter", presents a method for training a more compact general-purpose language representation model called DistilBERT, which is a streamlined version of the larger BERT model (Bidirectional Encoder Representations from Transformers). Although BERT is highly effective for Natural Language Processing (NLP) tasks, its size and complexity can lead to slower inference times, particularly in on-the-edge or resource-constrained environments, potentially impacting user experience. This paper aims to pre-train the DistilBERT model to enhance inference performance by reducing both its size and inference time while maintaining accuracy. According to the paper, DistilBERT achieves a 40% reduction in model size, preserves 97% of BERT's language understanding capabilities, and improves processing speed by 60%.

The source files and collaborative work done on this project is available for review on GitHub[2].

2 Project Justification

In this project, we will not replicate the transfer learning or model training processes of DistilBERT. Instead, we aim to reproduce the evaluation metrics shown in Tables 1 and 2 on page 3 of the referenced research paper, across various datasets and downstream

[1](Sanh et al, 2019): https://arxiv.org/pdf/1910.01108

[2]Project on GitHub: https://github.com/nadunchandrabahu/COMP8240-GroupM

tasks. This will allow us to assess DistilBERT's performance both on the datasets from the paper and on some of our own datasets. The DistilBERT model is available for evaluation through Hugging Face's transformers library (Wolf et al., 2019). In the original research, DistilBERT was evaluated using three datasets: the GLUE (General Language Understanding and Evaluation)[3] benchmark, which includes nine sentence-pair language understanding tasks; IMDb (Internet Movie Database)[4], used for sentiment classification of user movie reviews; and SQuAD (Stanford Question Answering Dataset)[5], designed for question-answering based on a provided context.

We will run some python code on Jupyter/Colab notebooks to make predictions on the data with the DistilBERT model and calculate various metrics: the average score across the nine GLUE benchmark tasks, test accuracy on the IMDb dataset, and the Exact Match (EM) and F1 score on the SQuAD task. These metrics are reported in Tables 1 and 2 of the research paper and it is the aim of this project to replicate these results. Each group member will also show the metrics using new data sources to perform similar NLP tasks with DistilBERT.

Table 1 below shows the scoring of DistilBERT evaluation on 9 GLUE benchmark tasks as shown in the research paper.

Table 1: BERT and DistilBERT results on GLUE tasks

	Metrics									
Model.Name	Score	CoLA	MNLI	MRPC	QNLI	QQP	RTE	SST.2	STS.B	WNLI
BERT-base	79.5	56.3	86.7	88.6	91.8	89.6	69.3	92.7	89	53.5
DistilBERT	77	51.3	82.2	87.5	89.2	88.5	59.9	91.3	86.9	56.3

Table 2 below shows the test accuracy of IMDb sentiment analysis tasks, and EM/F1 scores of the SQuAD question answering task as shown on the research paper.

Table 2: IMDb and SQuAD metrics

	IMDb	SQuAD	
Model Name			
	acc.	EM	F1
BERT-base	93.46	81.2	88.5
DistilBERT	92.82	77.7	85.8

[3]GLUE Benchmark: https://gluebenchmark.com/

[4]IMDb dataset: https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews

[5]SQuAD v1.1 dataset: https://rajpurkar.github.io/SQuAD-explorer/

3 Original Datasets

3.1 GLUE Benchmark

Describe about the dataset. Whole section 3 (Original Dataset should be 1/3rd of a page).

3.2 IMDb dataset

Describe about the dataset. Whole section 3 (Original Dataset should be 1/3rd of a page).

3.3 SQuAD dataset

The research paper makes use of SQuAD v1.1 development set available on GitHub[6]. The dataset consists of 17968 question-answer combinations. Each question and answer is based on the provided context. There are multiple ground truth answers to the same question. We only need to extract the context, question and answer (answer_text) for the purpose of calculating Exact Match and F-Score.

```
## Columns of SQuAD devset:
##  ['answer_text', 'answer_start', 'question', 'context', 'subject']
```

4 Replication of Evaluation on Datasets

We are able to access the DistilBERT model from 'transformers' library provided by HuggingFace. It is essential to have pyTorch installed before using the model to make predictions. The amount of time taken to make predictions depends on the size of the dataset, computational power and NLP task it tries to achieve.

4.1 GLUE Benchmark

Write about how you did the replication. Include link to notebook

4.2 IMDb dataset

Write about how you did the replication. Include link to notebook

4.3 SQuAD dataset

Evaluation of SQuAD (Stanford QUestion Answering Dataset) question answering task was performed by Nadun Chandrabahu and the Jupyter-notebook (SQuad-v1.1.ipynb) is available in the github repository.

[6]SQuAD v1.1 Dev set: https://github.com/rajpurkar/SQuAD-explorer/tree/master/dataset

Using the DistilBERT model, predicted answers were obtained based on the context and question. The inference time was 44 minutes on an AMD Ryzen 5 CPU with 16GB RAM. The model prediction returns a score, start and answer. I made a new column called Exact match that is either 1 or 0 if the predicted answer is the exact same as the answer_text column. And I calculated the average score when the model predicted an exact match. I obtained a result of 77.6%, while the research paper reported an EM of 77.7%.

The F-Score was calculated by using the following formula.

The F-score is given by $F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$.

I used the f1_score function from Python's scikit-learn library to calculate the F-Score, as well as precision and recall, which rely on the counts of True Positives, False Positives, and False Negatives. True Positives occur when the predicted values match the ground truth exactly. False Positives arise when the prediction overlaps with the ground truth but is not fully correct, while False Negatives are predictions that do not align with any part of the ground truth.

I obtained an F-Score of 75.2%, which is 10.6% lower than the 85.8% reported in the paper. This difference may stem from the inherent variability in calculating Precision and Recall, as the model's predictions can differ slightly with each run, significantly influencing Precision and Recall values, and hence the F-Score.

5 Evaluation on New Datasets

5.1 Nadun Chandrabahu

Evaluation of my new dataset McTest (Machine Comprehension Test)[7] (Richardson et al, 2013) with question answering task was performed and the Jupyter-notebook (own-dataset.ipynb) is available in the github repository.

The dataset includes 600 records of the following columns:

```
## Columns of McTest devset:
##  idx, question, story,
## properties, answer_options, answer, question_is_multiple
```

The dataset can be easily loaded into Python using the `load_datasets` method from the HuggingFace datasets library. I had to combine the above two columns answer_options, which is a dictionary of all the possible multiple choice answers and answer, which is the correct choice out of A, B, C, or D, into a new column called answer_text which would act as the ground truth label. The model predictions once again included a score, start and answer (predicted answer) and the rest of the processing & evaluating of EM and F1 scores was carried out similar to how SQuAD task was evaluated.

[7]McTest Dataset: https://huggingface.co/datasets/sagnikrayc/mctest

I selected the McTest dataset due to it being another question-answering NLP task that is possible to be conducted using the DistilBERT model. Previously, we achieved an Exact Match (EM) score of 77.6% and an F1 score of 75.2% on the SQuAD question-answering task. I aimed to achieve similar results with this dataset. However, while I achieved an EM score of 76.1%, my F-Score was only 31.2%, this could be due to the ground truth labels in this dataset being formatted much differently to how DistilBERT makes predictions. As a solution, I could have manually curated the ground truth anwers to be similar to the predictions made by DistilBERT.

5.2 Tai Ho

Write about your own dataset and metrics

5.3 Thanushreyas Appaji

Write about your own dataset and metrics

6 Reflections

Each of us can write a paragraph about our reflections on the project.

I, Nadun Chandrabahu, successfully replicated the evaluation results of DistilBERT on the SQuAD task, and explored the same metrics on McTest NLP question answering task. Although my EM scores were satisfactory, my F1 scores did not match up to expectations on both the SQuAD and MCTest question-answering tasks. This discrepancy may stem from an error in my method to F1 score calculation or could be due to the random errors in model predictions, which can notably impact the F1 score as True Positives, False Positives and False Negatives may be incorrectly counted.

7 References

1. Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. https://doi.org/10.48550/arXiv.1910.01108

2. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., Platen, P. V., M, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., … Drame, M. (2021). HuggingFace's Transformers: State-of-the-art Natural Language Processing. Webology. https://doi.org/10.48550/arXiv.1910.03771

3. Richardson, M., Burges, C. J., & Renshaw, E. (2013). MCTest: A Challenge Dataset for the Open-Domain Machine Comprehension of Text. EMNLP. https://mattr1.github.io/mctest/MCTest_EMNLP2013.pdf