VIETNAM GENERAL CONFEDERATION OF LABOUR
TON DUC THANG UNIVERSITY
FACULTY OF INFORMATION TECHNOLOGY

DEEP LEARNING

FINAL PROJECT II

Supervisor: **Le Anh Cuong**

Authors: **Tran Vu Ky Anh – 520K0323**

HO CHI MINH CITY, 2023

VIETNAM GENERAL CONFEDERATION OF LABOUR
TON DUC THANG UNIVERSITY
FACULTY OF INFORMATION TECHNOLOGY

DEEP LEARNING

FINAL PROJECT II

Supervisor: **Le Anh Cuong**

Authors: **Tran Vu Ky Anh – 520K0323**

HO CHI MINH CITY, 2023

PROJECT IS COMPLETED
AT TON DUC THANG UNIVERTY

I commit that this project is my / our own project and is supervised by Mr. Le Anh Cuong. The contents, results in this topic are honest and unpublished in any form before. The data in the tables for analyzing, commenting and evaluating are collected from various sources and by the authors and the citations are specified in References section.

In addition, a number of comments and assessments as well as data from other authors and organizations are also used in the project are referenced and annotated clearly.

If this project has any cheating or plagiarism, I take full responsibility for the content of my project. Ton Duc Thang University is not related to infringement of copyright caused by me during the process of this project implementation.

Ho Chi Minh City, May 5th , 2023

Authors

(signature and full name)

Anh

Trần Vũ Kỳ Anh

Contents

I. TEXT CLASSIFICATION

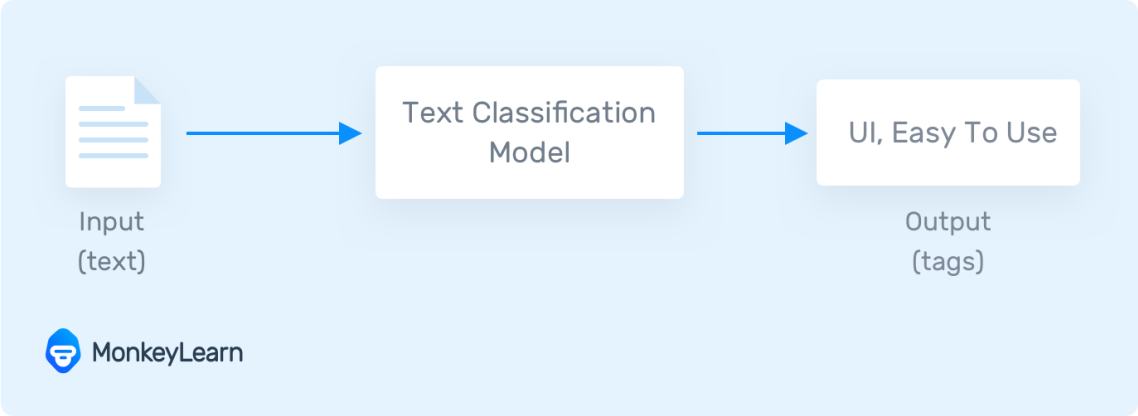
What is text classification

Text classification is a machine learning technique that assigns a set of predefined categories to open-ended text. Text classifiers can be used to organize, structure, and categorize pretty much any kind of text – from documents, medical studies and files, and all over the web.

Text classification can be used for a wide range of applications such as spam filtering, sentiment analysis, topic classification, language identification, and many others. The process of text classification typically involves several steps, including data preprocessing, feature extraction, model training, and evaluation.

For example, new articles can be organized by topics; support tickets can be organized by urgency; chat conversations can be organized by language; brand mentions can be organized by sentiment; and so on.

Here is an example figure to show how it work:
"The user interface is quite straightforward and easy to use."
A text classifier can take this phrase as an input, analyze its content, and then automatically assign relevant tags, such as UI and Easy To Use.



In text classification: Model receive a text as input and return class labels and their associated probabilities

Dataset

COLA DATASET

The Corpus of Linguistic Acceptability (COLA) consists of English acceptability judgments drawn from books and journal articles on linguistic theory. Each example is a sequence of words annotated with whether it is a grammatical English sentence. The public version contains 9594 sentences belonging to training and development sets, and excludes 1063 sentences belonging to a held out test set

For example:

“I love the atmosphere of the morning. Nice coffee brew”

Valid: 0.949

Invalid: 0.051

II. MODEL

RoBERTa

RoBERTa (short for "Robustly optimized BERT approach") is a large pre-trained language model developed by Facebook AI Research in 2019. It is based on the same architecture as BERT, but it uses a larger pre-training dataset and some modifications to the training process that make it perform better on a range of natural language processing tasks.

To use RoBERTa for text classification, you would typically fine-tune the pre-trained model on a specific task using a labeled dataset. The fine-tuning process involves adding a classification layer on top of the pre-trained RoBERTa model, and then training the entire model on the labeled dataset.

During fine-tuning, the RoBERTa model learns to map input text to the correct output label, based on the patterns it has learned from the pre-training data and the labeled dataset. The fine-tuned model can then be used to classify new text inputs.

Code

BERT

BERT (Bidirectional Encoder Representations from Transformers) is a pre-trained language model developed by Google researchers in 2018 that has achieved state-of-the-art results on a wide range of natural language processing tasks, including text classification.

To use BERT for text classification. The fine-tuning process involves adding a classification layer on top of the pre-trained BERT model, and then training the entire model on the labeled dataset.

During fine-tuning, the BERT model learns to map input text to the correct output label, based on the patterns it has learned from the pre-training data and the labeled dataset. The fine-tuned model can then be used to classify new text inputs.

Additionally, BERT uses a combination of masked language modeling and next sentence prediction during pre-training, which allows it to learn a deeper understanding of the relationships between words and sentences in natural language.

CODE

DISTILBERT

DistilBERT is a smaller and faster version of the BERT model, developed by Hugging Face in 2019. It is designed to be a more computationally efficient model that can be used on devices with limited computational resources, such as mobile phones or embedded systems.

Like BERT, DistilBERT is a pre-trained language model that can be fine-tuned on a specific task, such as text classification. The fine-tuning process involves adding a classification layer on top of the pre-trained DistilBERT model, and then training the entire model on the labeled dataset.

One of the key differences between DistilBERT and BERT is that DistilBERT uses a knowledge distillation technique during training, which involves training a smaller model (DistilBERT) to mimic the behavior of a larger model (BERT). This allows DistilBERT to achieve similar performance to BERT while using fewer computational resources.

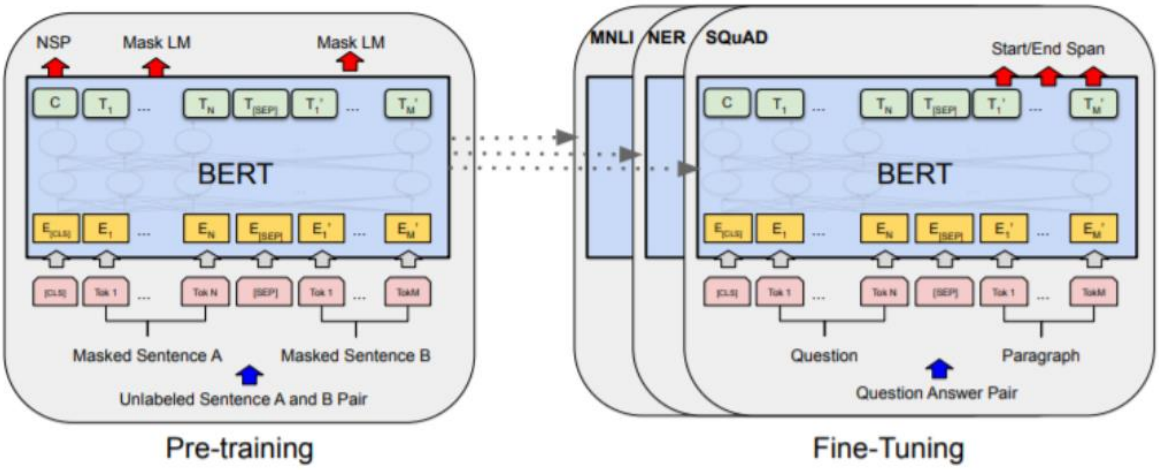
In addition to being more computationally efficient than BERT, DistilBERT also has a smaller memory footprint, which makes it easier to deploy in production systems. Despite its smaller size, DistilBERT has been shown to achieve similar performance to BERT on a wide range of natural language processing tasks, including text classification.

CODE

III. ARCHITECTURE

BERT:

BERT reuses the encoder block from the Transformer. The BERT_{BASE} uses 12 encoder blocks, while this number for the BERT_{LARGE} version is 24. Note that, while these two versions also have some other differences in layer sizes, the fundamental architectures are the same.



Architecture of BERT

	$BERT_{BASE}$	$BERT_{LARGE}$
No. encoders	12	24
Embedding dim	768	1024
Attention heads	12	16
No. parameters	110M	340M

Comparison of size in 2 versions.

RoBERTa

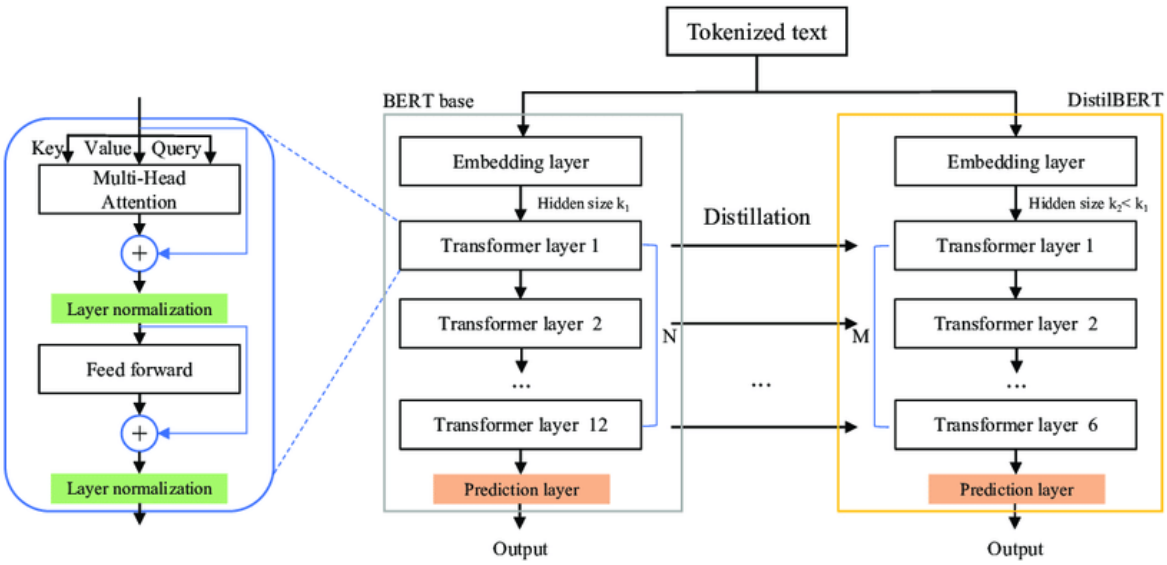
RoBERTa uses the same architecture as BERT.

To improve the performance of the BERT architecture, the authors of RoBERTa made some changes to the architecture and training procedure. They **removed the Next Sentence Prediction (NSP) objective**, which improved downstream task performance. They also **trained the model with larger batch sizes and longer sequences**, which improved perplexity on the masked language modeling objective and end-task accuracy. Additionally, they dynamically **changed the masking pattern during training**, which improved performance compared to a single static mask. These changes resulted in a pre-trained model, RoBERTa, that outperformed BERT on a range of natural language processing tasks.

DistilBERT:

DistilBERT uses a similar general architecture as BERT, but with fewer encoder blocks (6 blocks, compared to 12 blocks of BERT_{BASE} and 24 blocks of BERT_{LARGE}).

Also, the token-type embeddings and the pooler are removed.



IV. CHARACTERISTICS

BERT:

- Transformer-based pre-trained language model developed by Google.
- Large model with 110 million parameters in the base version and 340 million parameters in the large version.
- Uses a combination of masked language modeling and next sentence prediction during pre-training.
- Can model long-range dependencies in input text using a self-attention mechanism.
- Achieves state-of-the-art performance on a wide range of natural language processing tasks, including text classification.
- Can be computationally expensive to train and deploy, especially in the larger versions.

RoBERTa:

- Transformer-based pre-trained language model developed by Facebook AI Research.

- Similar in architecture to BERT but uses a larger and more diverse set of pre-training data and a modified training process.
- Has a similar number of parameters to the large version of BERT.
- Uses a modified training process that removes certain biases and encourages the model to focus on longer-term dependencies in the input text.
- Has been shown to outperform BERT on certain tasks.
- Achieves state-of-the-art performance on a wide range of natural language processing tasks, including text classification.
- Can be computationally expensive to train and deploy, similar to the large version of BERT.

DistilBERT:

- Smaller and faster version of BERT developed by Hugging Face.
- Has 66 million parameters, which is much smaller than BERT and RoBERTa.
- Uses a knowledge distillation technique during training to achieve similar performance to BERT while using fewer computational resources.
- Achieves similar performance to BERT and RoBERTa on a wide range of natural language processing tasks, including text classification.
- Has a smaller memory footprint, which makes it easier to deploy in production systems.
- Is faster to train and deploy than larger models.
- Is a good choice for use cases where computational resources are limited.

V. Pros and Cons

BERT:

Pros:

- Achieves state-of-the-art performance on a wide range of natural language processing tasks, including text classification.
- Can model long-range dependencies in input text using a self-attention mechanism.
- Has been widely adopted and has a large community of users and resources available.
- Has a large pre-training dataset and provides a strong foundation for fine-tuning on specific tasks.

Cons:
- Is computationally expensive, especially in the larger versions.
- May require a large amount of labeled data for fine-tuning on certain tasks.
- May be slower to train and deploy than smaller models.

RoBERTa:

Pros:
- Uses a larger and more diverse set of pre-training data than BERT, which can lead to better performance on a wider range of tasks.
- Uses a modified training process that removes certain biases and encourages the model to focus on longer-term dependencies in the input text, which can improve performance on certain tasks.
- Has been shown to outperform BERT on certain tasks.
- Has a large community of users and resources available.

Cons:
- Is computationally expensive, similar to the large version of BERT.
- May require a large amount of labeled data for fine-tuning on certain tasks.
- May be slower to train and deploy than smaller models.

DistilBERT:

Pros:

- Achieves similar performance to BERT and RoBERTa while using fewer computational resources.
- Has a smaller memory footprint, which makes it easier to deploy in production systems.
- Is faster to train and deploy than larger models.
- Is a good choice for use cases where computational resources are limited.

Cons:

- May not perform as well as BERT or RoBERTa on certain tasks.
- Has a smaller capacity for modeling long-range dependencies than larger models.
- May require a larger amount of labeled data for fine-tuning on certain tasks than larger models.

VI. COMPARISON

Comparison	BERT October 11, 2018	RoBERTa July 26, 2019	DistilBERT October 2, 2019
Parameters	**Base:** 110M **Large:** 340M	**Base:** 125 **Large:** 355	**Base:** 66
Layers / Hidden Dimensions / Self-Attention Heads	**Base:** 12 / 768 / 12 **Large:** 24 / 1024 / 16	**Base:** 12 / 768 / 12 **Large:** 24 / 1024 / 16	**Base:** 6 / 768 / 12
Training Time	**Base:** 8 x V100 x 12d **Large:** 280 x V100 x 1d	1024 x V100 x 1 day (4-5x more than BERT)	**Base:** 8 x V100 x 3.5d (4 times less than BERT)
Performance	Outperforming SOTA in Oct 2018	88.5 on GLUE	97% of BERT-base's performance on GLUE
Pre-Training Data	BooksCorpus + English Wikipedia = 16 GB	BERT + CCNews + OpenWebText + Stories = 160 GB	BooksCorpus + English Wikipedia = 16 GB
Method	Bidirectional Trans-former, MLM & NSP	BERT without NSP, Using Dynamic Masking	BERT Distillation

Source

Model size: RoBERT is the largest of the three models. RoBERTa has a similar number of parameters to the large version of BERT, while DistilBERT is much smaller, with only 66 million parameters.

Pre-training data: RoBERTa uses a larger and more diverse set of pre-training data than BERT, which can lead to better performance on a wider range of tasks. BERT and DistilBERT use similar pre-training data.

Training process: RoBERTa uses a modified training process that removes certain biases and encourages the model to focus on longer-term dependencies in the input text, which can improve performance on certain tasks. DistilBERT uses a knowledge distillation technique during training to achieve similar performance to BERT while using fewer computational resources.

Performance: BERT, RoBERTa, and DistilBERT have all achieved state-of-the-art performance on a wide range of natural language processing tasks, including text classification. However, RoBERTa has been shown to outperform BERT on certain tasks, while DistilBERT is a more computationally efficient model that performs similarly to BERT and RoBERTa.

Computational resources: BERT is the most computationally expensive of the three models, while DistilBERT is the most computationally efficient. RoBERTa is similar in computational requirements to the large version of BERT.

Training time: RoBERTa has a longer training time compared to BERT due to its larger pre-training data and modified training process. DistilBERT has a shorter training time compared to BERT and RoBERTa due to its smaller size.

Inference time: DistilBERT has the shortest inference time as it has fewer parameters and requires less computational resources during inference

BERT and RoBERTa are highly effective models for text classification, especially when computational resources are not a constraint. DistilBERT, on the other hand, is a more computationally efficient model that performs similarly to BERT and RoBERTa, making it a good choice for use cases where computational resources are limited.

VII. Metric and evaluation

The Matthews correlation coefficient (MCC), instead, is a more reliable statistical rate which produces a high score only if the prediction obtained good results in all of the four confusion matrix categories (true positives, false negatives, true negatives, and false positives), proportionally both to the size of positive elements and the size of negative elements in the dataset.

Its job is to gauge or measure the difference between the predicted values and actual values

MCC is a best single-value classification metric which helps to summarize the confusion matrix or an error matrix. A confusion matrix has four entities:

A score of 1 represents a perfect prediction, 0 represents a random prediction, and -1 represents a completely wrong prediction.

True positives (TP)

True negatives (TN)

False positives (FP)

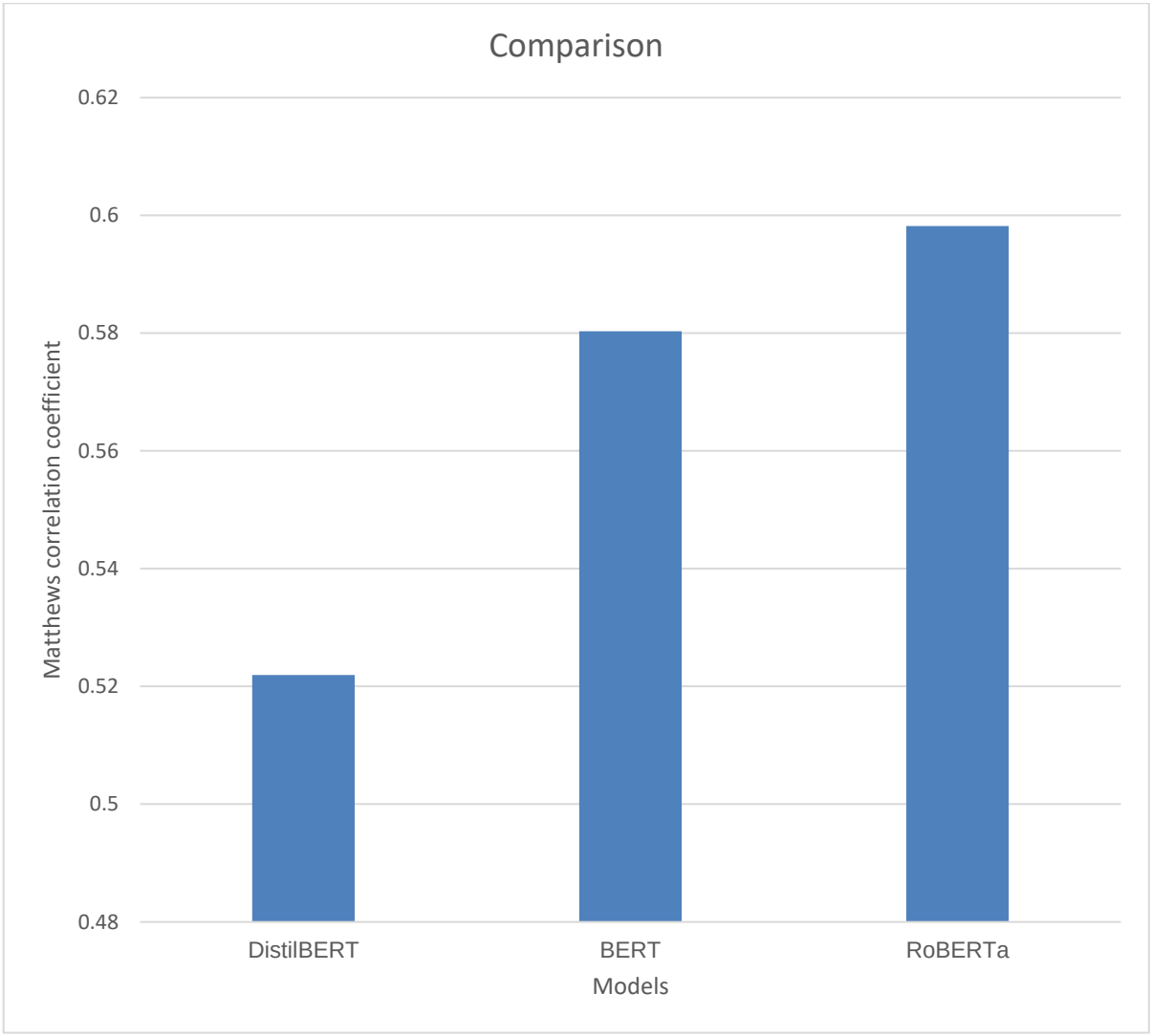
False negatives (FN)

And is calculated by the formula:

$$MCC = \frac{TN \times TP - FN \times FP}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

MCC produces a more informative and truthful score in evaluating binary classifications **than accuracy and F1 score**

RESULT


Comparison
0.62
0.6
0.58
0.56
0.54
0.52
0.5
0.48
Matthews correlation coefficient
DistilBERT
BERT
RoBERTa
Models

REFERENCES

English Documents

1. https://monkeylearn.com/text-classification/#:~:text=Tutorial-,What%20is%20Text%20Classification%3F,and%20all%20over%20the%20web.
2. https://paperswithcode.com/dataset/cola
3. https://www.geeksforgeeks.org/overview-of-roberta-model/
4. https://tungmphung.com/a-review-of-pre-trained-language-models-from-bert-roberta-to-electra-deberta-bigbird-and-more/#bert
5. https://www.analyticsvidhya.com/blog/2022/10/a-gentle-introduction-to-roberta/
6. https://medium.com/dataseries/bert-distilbert-roberta-and-xlnet-simplified-explanation-and-implementation-5a9580242c70
7.