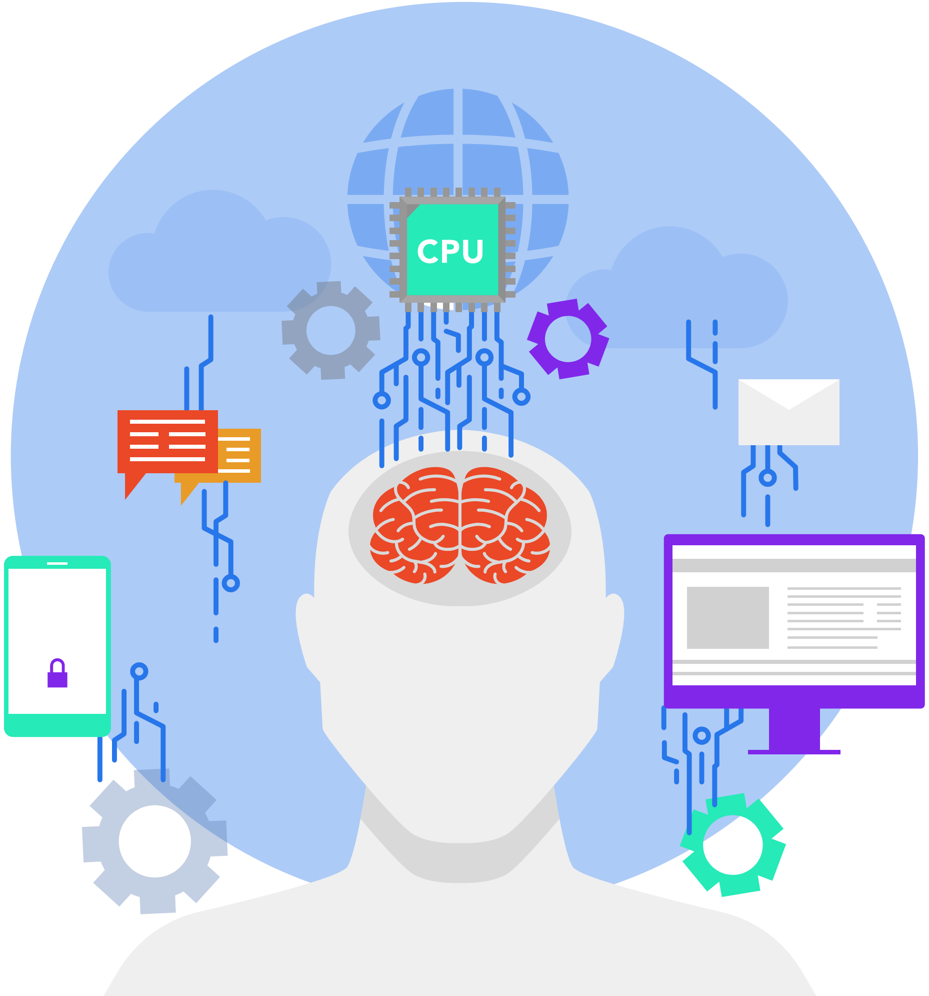
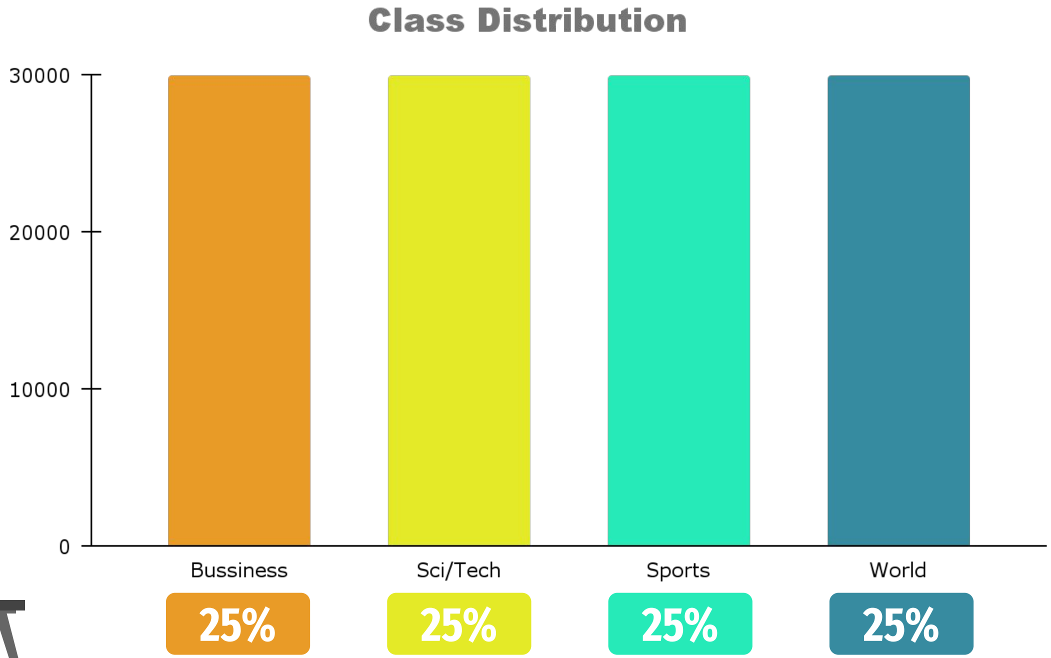



# Fine-Tuning 🤗 Hugging Face Transformers for News Text Classification

Bárbara Rodrigues - *up202007163*
Guilherme Pereira - *up202007375*
Lucas Sousa - *up202004682*

# Dataset

Dataset for **News Topic Classification** and perfectly balanced, as each label has the same amount of samples, exactly 30000 news per type.



# 🤗Model Selection

dstefa / **roberta-base_topic_classification_nyt_news** like 2

This model is a fine-tuned version of roberta-base on the NYT News dataset, which contains 256,000 news titles from articles published from 2000 to the present (https://www.kaggle.com/datasets/aryansingh0909/nyt-articles-21m-2000-present).

- it has been **fine-tuned on New York Times news articles**, making it well-suited for your task.
- it may **better understand the language, topics, and structure** commonly found in news content.
- we may **achieve good results with minimal effort** compared to training a model from scratch or using a generic pre-trained model.

| class | Description |
|---|---|
| 0 | Sports |
| 1 | Arts, Culture, and Entertainment |
| 2 | Business and Finance |
| 3 | Health and Wellness |
| 4 | Lifestyle and Fashion |
| 5 | Science and Technology |
| 6 | Politics |
| 7 | Crime |

# 🤗Model Selection

google-bert / **bert-base-uncased** like 1.56k

Model size | 110M params

Pretrained model on English language using a masked language modeling (MLM) objective. It was introduced in this paper and first released in this repository. This model is uncased: it does not make a difference between english and English.

distilbert / **distilbert-base-uncased** like 432

Model size | 67M params

DistilBERT is a transformers model, smaller and faster than BERT, which was pretrained on the same corpus in a self-supervised fashion, using the BERT base model as a teacher. This means it was pretrained on the raw texts only, with no humans labelling them in any way (which is why it can use lots of publicly available data) with an automatic process to generate inputs and labels from those texts using the BERT base model.

- Pretrained on **BookCorpus**, a dataset consisting of 11,038 unpublished books and **English Wikipedia** (excluding lists, tables and headers).
- Pre Trained with **Masked Language Modeling** and **Next Sentence Prediction** as main goals
- Multiple versions available, chosen version trained on **uncased** text.

# Approach

1. Divided the dataset into **train** (72%), **validation** (8%) and **test** (20%)
2. Tested the *roberta-base_topic_classification_nyt_news* **without fine tuning** for comparison purposes
3. Fined tuned the *roberta-base_topic_classification_nyt_news* model using a **sample of the dataset**, and evaluate results
4. Fined tuned the *roberta-base_topic_classification_nyt_news* model using the **full dataset**, and evaluate results
5. Repeat steps **3** and **4** but for the *bert -base-uncased* model (without trained classification)
6. **Domain Adaptation** using *distilbert/distilbert-base-uncased* model, end evaluate results

# Training Parameters

```
training_args = TrainingArguments(
        output_dir="./results",
        learning_rate=2e-5,
        per_device_train_batch_size=6,
        per_device_eval_batch_size=6,
        num_train_epochs=10,
        weight_decay=0.01,
        evaluation_strategy="epoch",
        save_strategy="epoch",
        load_best_model_at_end=True,
        warmup_steps=500
        gradient_accumulation_steps=10
)
```

```
data_collator
=DataCollatorWithPadding(tokenizer=tokenizer)

trainer = Trainer(
   model=model,
   args=training_args,
   train_dataset=tokenized_dataset["train"],
   eval_dataset=tokenized_dataset["validation"],
   tokenizer=tokenizer,
   data_collator=data_collator,
   compute_metrics=compute_metrics
)
```

# Domain Adaptation

1. Trained the *distilbert-base-uncased* model with Masked Language Modeling.
    a. Choose Distilbert instead of Bert due to VRAM limitations
    b. Used the **DistilbertForMaskedLM** model and **DistilbertTokenizer**
    c. And **DataCollatorForLanguageModeling**
    d. Used the full dataset text as input, labels were ignored
    e. Saved the model
2. Fine-tuned that model for Sequence Classification
    a. Loaded the model
    b. Used the **DistilbertForSequenceClassification** and **DistilbertTokenizer**
    c. And **DataCollatorWithPadding**
    d. Standard fine-tune with our smaller sampled dataset

# Previous Results

Traditional ML Models

| Evaluate the models | Accuracy | F1-Score | Precision | Recall |
|---|---|---|---|---|
| Naîve Bayes | 0.895 | 0.894 | 0.894 | 0.895 |
| Logistic Regression | 0.897 | 0.896 | 0.896 | 0.897 |
| Decision Tree | 0.772 | 0.772 | 0.772 | 0.772 |
| Random Forest | 0.86 | 0.86 | 0.86 | 0.86 |
| Support Vector Machine | 0.896 | 0.895 | 0.895 | 0.896 |
| Extreme Gradient Boost | 0.88 | 0.88 | 0.88 | 0.88 |
| Gradient Boosting | 0.89 | 0.89 | 0.89 | 0.89 |
| Nearest Neighbor | 0.763 | 0.767 | 0.787 | 0.763 |

# Results

| Evaluate the models | Accuracy | F1-Score | Precision | Recall |
|---|---|---|---|---|
| Roberta NYT Classifier | 0.75 | 0.72 | 0.80 | 0.75 |
| Roberta NYT Classifier FT Small Dataset | 0.92 | 0.92 | 0.92 | 0.92 |
| Roberta NYT Classifier FT Full Dataset | 0.95 | 0.95 | 0.95 | 0.95 |
| Bert Base Small Dataset | 0.93 | 0.93 | 0.93 | 0.93 |
| Bert Base Full Dataset | 0.96 | 0.96 | 0.96 | 0.96 |
| Distilbert Base Domain Adaptation | 0.93 | 0.93 | 0.93 | 0.93 |

# Conclusions

- We were able to implement a wide array of different models in different contexts.
- As expected, all the transformers performed better than the traditional models, with the exception of the *dstefa/roberta-base_topic_classification_nyt_news* without fine tuning.
  - This is where our lack of fine-tuning shows its effect, even though the model had previously been fine-tuned with the *nyt_news* dataset
- Even though the process was tricky, **domain adaptation** was implemented; however the results weren't as good as expected.
- As expected, models fine-tuned with larger datasets produced better results
- In our project, the best model was *bert-base-uncased* with fine tuning with all of the data.


Q&A