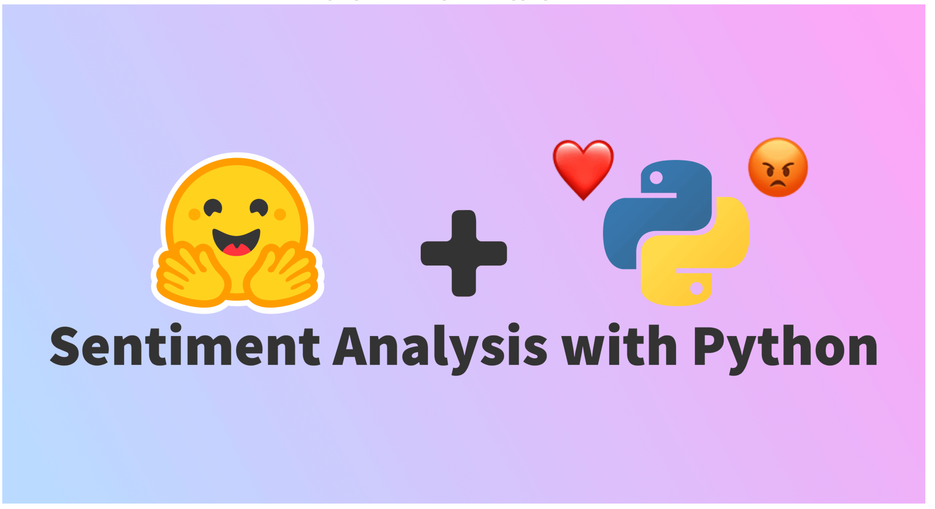
Natural Language Processing with Hugging Face Transformers



Project goals:

- Text Classification.
  - Sentiment Analysis. Classifies the polarity of a given text.
- Topic Classification. Classifies sequences into specified class names.
- Text Generator. Generates text from a given input.
- Token Classification.
  - Name Entity Recognition (NER). Labels each word with the entity it represents.
- Question answering. Extracts the answer from the context.
- Text Summarization. Generates a summary of a long sequence of text or document.
- Translation. Translates text into another language.

Importing Required Libraries

In [ ]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [ ]:
```
from transformers import pipeline
from transformers import AutoTokenizer
from transformers import AutoModel
```

```
2023-05-22 22:49:43.189604: I tensorflow/core/platform/cpu_feature_guard.cc:182] This TensorFlow binary is optimized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
```

In [ ]:
```
classifier = pipeline("sentiment-analysis", model="distilbert-base-uncased-finetuned-sst-2-english")
```

```
Xformers is not installed correctly. If you want to use memory_efficient_attention to accelerate training use the following command to install Xformers
pip install xformers.
```

In [ ]:
```
classifier("Having three long haired, heavy shedding dogs at home, I was pretty skeptical that this could hold up to all the hair and dirt they trek in, but this wonderful piece of tech has been nothing short of a godsend for me!
```

Out [ ]:
```
[{'label': 'POSITIVE', 'score': 0.9982457160949707}]
```

As we see, the sentiment is classified as Positive with 99.8% accuracy score.

In [ ]:
```
classifier = pipeline("zero-shot-classification", model="facebook/bart-large-mnli")
classifier(
    "Exploratory Data Analysis is the first course in Machine Learning Program that introduces learners to the broad range of Machine Learning concepts, applications, challenges, and solutions, while utilizing interesting real-li
    candidate_labels=["art", "natural science", "data analysis"],
)
```

Out [ ]:
```
{'sequence': 'Exploratory Data Analysis is the first course in Machine Learning Program that introduces learners to the broad range of Machine Learning concepts, applications, challenges, and solutions, while utilizing interestin
g real-life datasets',
 'labels': ['data analysis', 'art', 'natural science'],
 'scores': [0.9957792758941650.0026982498820871115, 0.0015224901726469398]}
```

As we see, 'data analysis' is the most successful candidate for the topic of this input, having 99.6% score.

In [ ]:
```
generator = pipeline("text-generation", model="gpt2")
generator("This course will teach you")
```

```
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
```

Out [ ]:
```
[{'generated_text': 'This course will teach you what you need to know to develop and become successful as a developer.\n\nThis course will cover:\n\nthe tools that can really make a big difference in the community\n\nthe tools fr
om which you learn (eg'}]
```

Alternatively, we can also use "distilgpt2" model, as well as some parameters, such length and number of the sentences needed. Distilled GPT-2 model is an English-language model pre-trained with the supervision of the smallest version of GPT-2. Like GPT-2, DistilGPT2 can be used to generate text. For more information about this model, please visit this link.

In [ ]:
```
generator = pipeline("text-generation", model="distilgpt2")
generator(
    "This course will teach you",
    max_length=30,
    num_return_sequences=2,
)
```

```
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
```

Out [ ]:
```
[{'generated_text': 'This course will teach you the basics of how to apply a special set of principles to an app that comes with a custom app.\n\n\n\n\n'},
 {'generated_text': 'This course will teach you the fundamental idea of using a keyboard in any given environment. It will teach you the tools and mechanisms necessary to create a modern'}]
```

In [ ]:
```
unmasker = pipeline("fill-mask", "distilroberta-base")
unmasker("This course will teach you all about <mask> models.", top_k=4)
```

Out [ ]:
```
[{'score': 0.19619765877723694,
  'token': 30412,
  'token_str': ' mathematical',
  'sequence': 'This course will teach you all about mathematical models.'},
 {'score': 0.04052731022238314,
  'token': 38163,
  'token_str': ' computational',
  'sequence': 'This course will teach you all about computational models.'},
 {'score': 0.03301792219281966,
  'token': 27930,
  'token_str': ' predictive',
  'sequence': 'This course will teach you all about predictive models.'},
 {'score': 0.031941514462232594,
  'token': 745,
  'token_str': ' building',
  'sequence': 'This course will teach you all about building models.'}]
```

In [ ]:
```
ner = pipeline("ner", model="dbmdz/bert-large-cased-finetuned-conll03-english", grouped_entities=True)
ner("My name is Roberta and I work with IBM Skills Network in Toronto")
```

Out [ ]:
```
[{'entity_group': 'PER',
  'score': 0.9993105,
  'word': 'Roberta',
  'start': 11,
  'end': 18},
 {'entity_group': 'ORG',
  'score': 0.9976597,
  'word': 'IBM Skills Network',
  'start': 35,
  'end': 53},
 {'entity_group': 'LOC',
  'score': 0.99702173,
  'word': 'Toronto',
  'start': 57,
  'end': 64}]
```

In [ ]:
```
del ner
```

As we see, the model properly identifies all entities in the sentence with highest confidence scores.

In [ ]:
```
qa_model = pipeline("question-answering", model="distilbert-base-cased-distilled-squad")
question = "Which name is also used to describe the Amazon rainforest in English?"
context = "The Amazon rainforest, also known in English as Amazonia or the Amazon Jungle."
qa_model(question = question, context = context)
```

Out [ ]:
```
{'score': 0.8247056603431702, 'start': 48, 'end': 56, 'answer': 'Amazonia'}
```

As we see, the correct answer has been extracted with 82% confidence score.

In [ ]:
```
summarizer = pipeline("summarization", model="sshleifer/distilbart-cnn-12-6")
summarizer(
    """
Exploratory Data Analysis is the first course in Machine Learning Program that introduces learners to the broad range of Machine Learning concepts, applications, challenges, and solutions, while utilizing interesting real-life da
EDA is a visual and statistical process that allows us to take a glimpse into the data before the analysis. It lets us test the assumptions that we might have about the data, proving or disproving our prior believes and biases. I
As any data scientist would agree, the most challenging part in any data analysis is to obtain a good quality data to work with. Nothing is served to us on a silver plate, data comes in different shapes and formats. It can be str
This course will teach you to 'see' and to 'feel' the data as well as to transform it into analysis-ready format. It is introductory level course, so no prior knowledge is required, and it is a good starting point if you are inte
The course contains videos and reading materials, as well as well as a lot of interactive practice labs that learners can explore and apply the skills learned. It will allow you to use Python language in Jupyter Notebook, a cloud

"""
)
```

Out [ ]:
```
[{'summary_text': ' Exploratory Data Analysis is the first course in Machine Learning Program that introduces learners to the broad range of Machine Learning concepts, applications, challenges, and solutions . EDA is a visual and
statistical process that allows us to take a glimpse into the data before the analysis . It lays foundation for the analysis so our results go along with our expectations .'}]
```

In [ ]:
```
del summarizer
```

In [ ]:
```
en_fr_translator = pipeline("translation_en_to_fr", model="t5-small")
en_fr_translator("How old are you?")
```

Out [ ]:
```
[{'translation_text': 'Quel est votre âge ?'}]
```

If you would like to use a specific model that is from one specific language to another, you can also directly use the translation pipeline without specifying the model under the hood.

In [ ]:
```
translator = pipeline("translation", model="Helsinki-NLP/opus-mt-fr-en")
translator("La science des données est la meilleure.")
```

Out [ ]:
```
[{'translation_text': 'Data science is the best.'}]
```

In [ ]:
```
specific_model = pipeline(model="cardiffnlp/twitter-roberta-base-sentiment")
data = "Artificial intelligence and automation are already causing friction in the workforce. Should schools revamp existing programs for topics like #AI, or are new research areas required?"
specific_model(data)
```

Out [ ]:
```
[{'label': 'LABEL_1', 'score': 0.5272256731987}]
```

In [ ]:
```
original_model = pipeline("sentiment-analysis")
data = "Artificial intelligence and automation are already causing friction in the workforce. Should schools revamp existing programs for topics like #AI, or are new research areas required?"
original_model(data)
```

```
No model was supplied, defaulted to distilbert-base-uncased-finetuned-sst-2-english and revision af0f99b (https://huggingface.co/distilbert-base-uncased-finetuned-sst-2-english).
Using a pipeline without specifying a model name and revision in production is not recommended.
```

Out [ ]:
```
[{'label': 'NEGATIVE', 'score': 0.9989722367147827}]
```

In [ ]:
```
classifier = pipeline("zero-shot-classification", model="facebook/bart-large-mnli")
classifier(
    "I love travelling and learning new cultures",
    candidate_labels=["art", "education", "travel"],
)
```

Out [ ]:
```
{'sequence': 'I love travelling and learning new cultures',
 'labels': ['travel', 'education', 'art'],
 'scores': [0.9902299642562866, 0.005778191145509481, 0.003991852048784494]}
```

In [ ]:
```
generator = pipeline('text-generation', model = 'gpt2')
generator("Hello, I'm a language model", max_length = 30, num_return_sequences=3)
```

```
Setting `pad_token_id` to `eos_token_id`:50256 for open-end generation.
```

Out [ ]:
```
[{'generated_text': "Hello, I'm a language modeler. I wrote a program to help you build your first language, called C++ Standard C++ Standard, as"},
 {'generated_text': "Hello, I'm a language model that allows you to build apps with many different types of objects. This is the only set of objects I've ever"},
 {'generated_text': "Hello, I'm a language model.\n\nLet me give you an example…\n\nYou can create your own templates for the next step."}]
```

In [ ]:
```
nlp = pipeline("ner", model="Jean-Baptiste/camembert-ner", grouped_entities=True)
example = "Her name is Anjela and she lives in Seoul."

ner_results = nlp(example)
print(ner_results)
```

```
[{'entity_group': 'PER', 'score': 0.94814444, 'word': 'Anjela', 'start': 11, 'end': 18}, {'entity_group': 'LOC', 'score': 0.9986114, 'word': 'Seoul', 'start': 35, 'end': 41}]
```

In [ ]:
```
question_answerer = pipeline("question-answering", model="distilbert-base-cased-distilled-squad")
question_answerer(
    question="Which lake is one of the five Great Lakes of North America?",
    context="Lake Ontario is one of the five Great Lakes of North America. It is surrounded on the north, west, and southwest by the Canadian province of Ontario, and on the south and east by the U.S. state of New York, whose wat
)
```

Out [ ]:
```
{'score': 0.9834363460540771, 'start': 0, 'end': 12, 'answer': 'Lake Ontario'}
```

In [ ]:
```
summarizer = pipeline("summarization", model="sshleifer/distilbart-cnn-12-6",  max_length=59)
summarizer(
    """
Lake Superior in central North America is the largest freshwater lake in the world by surface area and the third-largest by volume, holding 10% of the world's surface fresh water. The northern and westernmost of the Great Lakes o
"""
)
```

Out [ ]:
```
[{'summary_text': ' Lake Superior is the largest freshwater lake in the world by surface area . It holds 10% of the world's surface fresh water . It straddles the Canada-U.S. border with the province of Ontario to the north . It
drains into Lake Huron via St.'}]
```

In [ ]:
```
translator = pipeline("translation_en_to_de", model="t5-small")
print(translator("New York is my favourite city", max_length=40))
```

```
[{'translation_text': 'New York ist meine Lieblingsstadt'}]
```