Huggingface

Huggingface

Introduction

- ML and data science platform
- helps users to build, deploy, and train ML models
- provides infrastructure to run and deploy AI in live applications
- largest community of ML-models

Huggingface

Ecosystem

- Models
  - Multimodal
  - Computer Vision
  - Natural Language Processing
    - Text Classification
    - Translation
    - Summarisation
    - Text Generation
    - ...
    - Fill-Mask
  - Audio
  - Tabular
  - Reinforcement Learning
- Datasets
- Metrics
- Documentation

Huggingface

Pipelines: Task-specific

- simple API to model inference
- two categories of pipeline abstractions
  - pipeline(task=…) general encapsulation for specific task

```
pipe = pipeline(task="text-classification")
```

```
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.
```

```
pipe("I like it very much.")
✓ 0.1s
[{'label': 'POSITIVE', 'score': 0.9998759031295776}]
```

Huggingface

Pipelines: Model-specific

- simple API to model inference
- two categories of pipeline abstractions
  - pipeline(model=…) model specific pipeline

Set up Pipeline:

```
pipe = pipeline(task="text-classification",
                model="nlptown/bert-base-multilingual-uncased-sentiment")
```

Run Pipeline:

```
pipe("I like it very much.")
```

✓ 0.2s

```
[{'label': '5 stars', 'score': 0.5000860095024109}]
```

Huggingface

Pipelines: Further details

- Transformers Pipeline

```python
def pipeline(
    task: str = None,
    model: Optional[Union[str, "PreTrainedModel", "TFPreTrainedModel"]] = None,
    config: Optional[Union[str, PretrainedConfig]] = None,
    tokenizer: Optional[Union[str, PreTrainedTokenizer, "PreTrainedTokenizerFast"]] = None,
    feature_extractor: Optional[Union[str, PreTrainedFeatureExtractor]] = None,
    image_processor: Optional[Union[str, BaseImageProcessor]] = None,
    framework: Optional[str] = None,
    revision: Optional[str] = None,
    use_fast: bool = True,
    token: Optional[Union[str, bool]] = None,
    device: Optional[Union[int, str, "torch.device"]] = None,
    device_map=None,
    torch_dtype=None,
    trust_remote_code: Optional[bool] = None,
    model_kwargs: Dict[str, Any] = None,
    pipeline_class: Optional[Any] = None,
    **kwargs,
) -> Pipeline:
```

default tokenizer of model used if not specified

Huggingface

Pipelines: Input

- flexible use
- can handle Strings or List<Strings>

```
# consume just a string
pipe("I like it very much.")
```

```
# %% consume a list
pipe(["I like it very much.",
      "I hate it."])
```

Huggingface

Task: Named Entity Recognition

- process structured and unstructured data
- classify named entities into predefined categories

Pipeline: `"ner"`

Apple Inc. was founded by Steve Jobs, Steve Wozniak, and Ronald Wayne on April 1, 1976, in Cupertino, California.

Named Entities:

Organisation | People | Date | Location

Huggingface

Task: Text Classification

- assign pre-defined categories or labels to a given text
- relevant for spam detection, topic categorization, sentiment analysis

Pipeline: `"text-classificiation"`

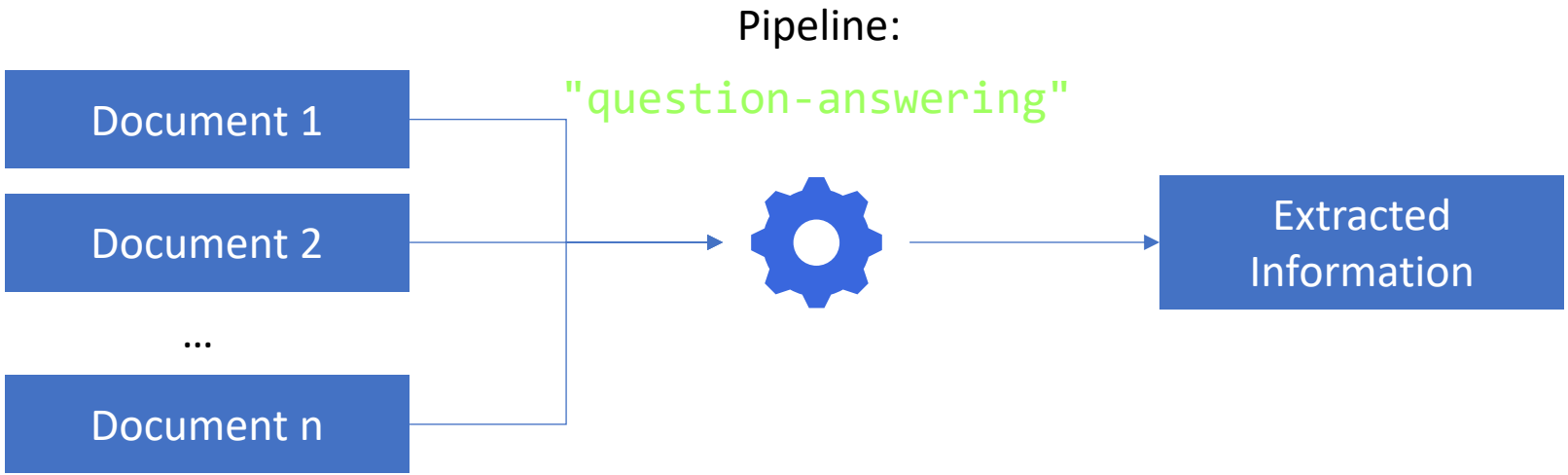
Sample Text: I like it very much.

Possible Categories: Positive Negative

Huggingface

Task: Question Answering

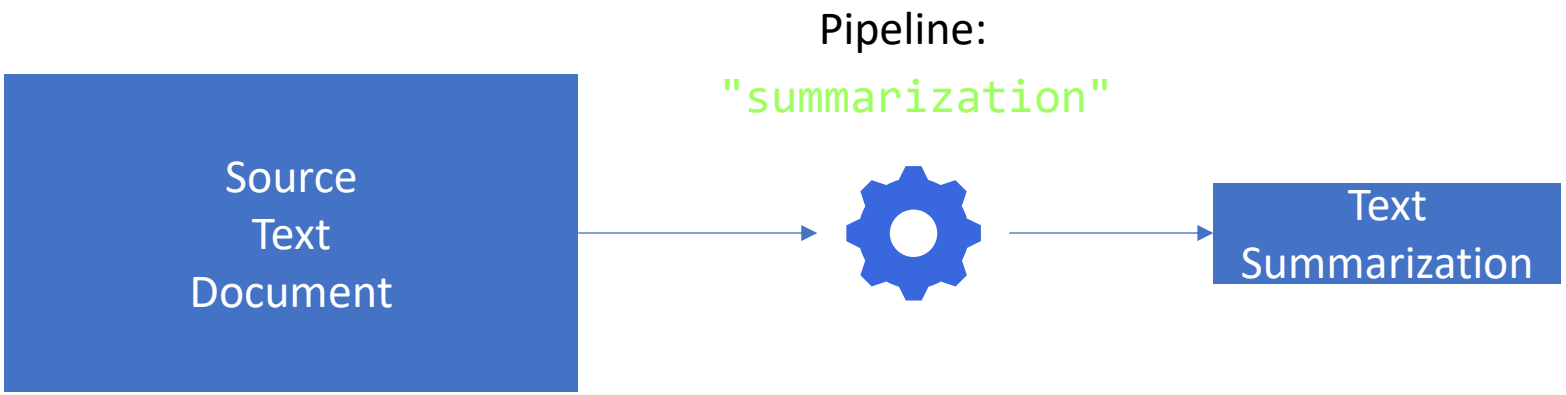
- understand and respond to questions
- extract relevant information



Huggingface

Task: Summarization

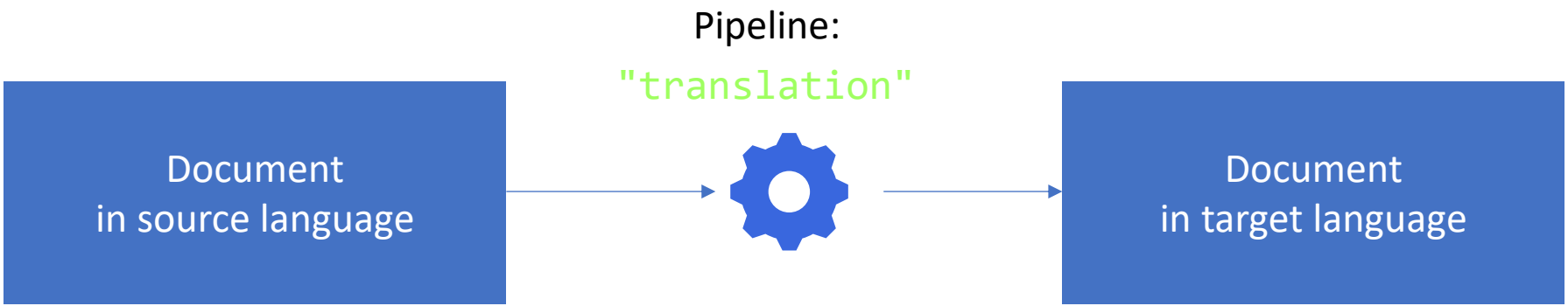
- distill essential information from source text
- preserve meaning and key details
- Applications
  - Document summarization
  - social media post summarization
  - news summarisation



Huggingface

Task: Translation

- translate a text from a source language into one or more target languages



Huggingface

Task: Fill-Mask

- predict masked / missing word within given sentence
- model are able to fill blank with suitable word or token based on context