Token classification

Fine-tuning the library models for token classification task such as Named Entity Recognition (NER), Parts-of-speech tagging (POS) or phrase extraction (CHUNKS). The main script `run_ner.py` leverages the 🤗 Datasets library. You can easily customize it to your needs if you need extra processing on your datasets.

It will either run on a datasets hosted on our hub or with your own text files for training and validation, you might just need to add some tweaks in the data preprocessing.

The following example fine-tunes BERT on CoNLL-2003:

```
python run_ner.py \
  --model_name_or_path bert-base-uncased \
  --dataset_name conll2003 \
  --output_dir /tmp/test-ner
```

To run on your own training and validation files, use the following command:

```
python run_ner.py \
  --model_name_or_path bert-base-uncased \
  --train_file path_to_train_file \
  --validation_file path_to_validation_file \
  --output_dir /tmp/test-ner
```

Note: This script only works with models that have a fast tokenizer (backed by the 🤗 Tokenizers library) as it uses special features of those tokenizers. You can check if your favorite model has a fast tokenizer in this table.