Main NLP tasks

- Token classification
- Masked language modeling (like BERT)
- Summarization
- Translation
- Causal language modeling pretraining (like GPT-2)
- Question answering

Token classification

Some popular token classification methods:

- **Named entity recognition (NER)**: Find the entities (such as persons, locations, or organizations) in a sentence. This can be formulated as attributing a label to each token by having one class per entity and one class for “no entity.”
- **Part-of-speech tagging (POS)**: Mark each word in a sentence as corresponding to a particular part of speech (such as noun, verb, adjective, etc.).
- **Chunking**: Find the tokens that belong to the same entity. This task (which can be combined with POS or NER) can be formulated as attributing one label (usually `B-`) to any tokens that are at the beginning of a chunk, another label (usually `I-`) to tokens that are inside a chunk, and a third label (usually `O`) to tokens that don’t belong to any chunk.

To build a token classifier we are going to be using a dataset named **CoNLL-2003 dataset**
, which contains news stories from Reuters. We will be building a NER classifier.

NER classes: `['O', 'B-PER', 'I-PER', 'B-ORG', 'I-ORG', 'B-LOC', 'I-LOC', 'B-MISC', 'I-`

`MISC']`

O means the word doesn’t correspond to any entity.
B-PER/I-PER means the word corresponds to the beginning of/is inside a person entity.
B-ORG/I-ORG means the word corresponds to the beginning of/is inside an organization entity.
B-LOC/I-LOC means the word corresponds to the beginning of/is inside a location entity.
B-MISC/I-MISC means the word corresponds to the beginning of/is inside a miscellaneous entity.

Note:

To tokenize a pre-tokenized input, we can use our `tokenizer` as usual and just add `is_split_into_words=True`: