# Torchtext +Dataloaders Tutorial

## Version

* Python >=3.7, <=3.10
* PyTorch >=1.13.0
* Torchtext >= 0.13.0
* Torchdata >= 0.5
* SpaCy's [en\_core\_web\_sm](https://spacy.io/models/en) and [de\_core\_news\_sm](https://spacy.io/models/de) models

## torchtext.datasets classes

from torchtext.datasets import Multi30k

* IMDB
* AG\_NEWS
* Multi30k
* TranslationDataset
* LanguageModelingDataset

We can iterate, downloadm preprocess and conjunctino with pytorch for creating training data.

## Load Tokenizer

We can load and customize the get tokenizer according to our need:

------------------------------\

from torchtext.data.utils import get\_tokenizer

tokenizer = get\_tokenizer("basic\_english")

------------------------------\

spacy\_en\_tokenizer = get\_tokenizer("spacy", language="en\_core\_web\_sm")

tokens = spacy\_en\_tokenizer(eng\_sent)

------------------------------\

def mytokenizer(sent):

return sent.upper().split()

my\_tokenizer = get\_tokenizer(mytokenizer)

## vocab

en\_vocab = vocab(

ordered\_en,

min\_freq=1,

specials=('<BOS>', '<EOS>', '<PAD>', '<unk>')

)

en\_stoi = en\_vocab.get\_stoi()

en\_itos = en\_vocab.get\_itos()

## Sequential

Sequential class to define a text processing pipeline that performs the following steps:

* Tokenize the text into individual tokens
* Convert the tokens to their numerical representation using a vocabulary
* Truncate the sequence to a maximum length
* Add a special token (such as <BOS> or <EOS>) to the beginning or end of the sequence