Implemented by sayef .

Overview

The FSNER model was proposed in Example-Based Named Entity Recognition by Morteza Ziyadi, Yuting Sun, Abhishek Goswami, Jade Huang, Weizhu Chen. To identify entity spans in a new domain, it uses a train-free few-shot learning approach inspired by question-answering.

Abstract

> *We present a novel approach to named entity recognition (NER) in the presence of scarce data that we call example-based NER. Our train-free few-shot learning approach takes inspiration from question-answering to identify entity spans in a new and unseen domain. In comparison with the current state-of-the-art, the proposed method performs significantly better, especially when using a low number of support examples.*

Model Training Details

identifier	epochs	datasets
sayef/fsner-bert-base-uncased	10	ontonotes5, conll2003, wnut2017, and fin (Alvarado et al.).

Installation and Example Usage

You can use the FSNER model in 3 ways:

1. Install directly from PyPI: `pip install fsner` and import the model as shown in the code example below

   or

2. Install from source: `python setup.py install` and import the model as shown in the code example below

   or

3. Clone repo and change directory to `src` and import the model as shown in the code example below

```
from fsner import FSNERModel, FSNERTokenizerUtils

model = FSNERModel("sayef/fsner-bert-base-uncased")

tokenizer = FSNERTokenizerUtils("sayef/fsner-bert-base-uncased")

# size of query and supports must be the same. If you want to find all the entitites
in one particular query, just repeat the same query n times where n is equal to the
number of supports (or entities).


query = [
    'KWE 4000 can reach with a maximum speed from up to 450 P/min an accuracy from
50 mg',
    'I would like to order a computer from eBay.',
]

# each list in supports are the examples of one entity type
# wrap entities around with [E] and [/E] in the examples

supports = [
        [
           'Horizontal flow wrapper [E] Pack 403 [/E] features the new retrofit-kit
„paper-ON-form“',
           '[E] Paloma Pick-and-Place-Roboter [/E] arranges the bakery products for
the downstream tray-forming equipment',
           'Finally, the new [E] Kliklok ACE [/E] carton former forms cartons and
trays without the use of glue',
           'We set up our pilot plant with the right [E] FibreForm® [/E]
configuration to make prototypes for your marketing tests and package validation',
           'The [E] CAR-T5 [/E] is a reliable, purely mechanically driven cartoning
machine for versatile application fields'
        ],
        [
            "[E] Walmart [/E] is a leading e-commerce company",
            "I recently ordered a book from [E] Amazon [/E]",
            "I ordered this from [E] ShopClues [/E]",
            "[E] Flipkart [/E] started it's journey from zero"
        ]
   ]

device = 'cpu'

W_query = tokenizer.tokenize(query).to(device)
W_supports = tokenizer.tokenize(supports).to(device)

start_prob, end_prob = model(W_query, W_supports)
```

```
output = tokenizer.extract_entity_from_scores(query, W_query, start_prob, end_prob,
thresh=0.50)

print(output)
```