OpenNRE Documentation

Tianyu Gao

May 20, 2021

Get Started

CHAPTER 1

Introduction

OpenNRE (https://github.com/thunlp/OpenNRE) is an open-source and extensible toolkit that provides a unified framework to implement relation extraction models. This package is designed for the following groups:

- **New to relation extraction**: We have hand-by-hand tutorials and detailed documents that can not only enable you to use relation extraction tools, but also help you better understand the research progress in this field.
- **Developers**: Our easy-to-use interface and high-performance implementation can acclerate your deployment in the real-world applications. Besides, we provide several pretrained models which can be put into production without any training.
- **Researchers**: With our modular design, various task settings and metric tools, you can easily carry out experiments on your own models with only minor modification. We have also provided several most-used benchmarks for different settings of relation extraction.
- **Anyone who need to submit an NLP homework to impress their professors**: With state-of-the-art models, our package can definitely help you stand out among your classmates!

1.1 What is Relation Extraction

Relation extraction is a natural language processing (NLP) task aiming at extracting relations (e.g., *founder of*) between entities (e.g., **Bill Gates** and **Microsoft**). For example, from the sentence *Bill Gates founded Microsoft*, we can extract the relation triple (**Bill Gates**, *founder of*, **Microsoft**).

Relation extraction is a crucial technique in automatic knowledge graph construction. By using relation extraction, we can accumulatively extract new relation facts and expand the knowledge graph, which, as a way for machines to understand the human world, has many downstream applications like question answering, recommender system and search engine.

1.2 How to Cite

A good research work is always accompanied by a thorough and faithful reference. If you use or extend our work, please cite the following paper:

```
@inproceedings{han-etal-2019-opennre,
    title = "{O}pen{NRE}: An Open and Extensible Toolkit for Neural Relation␣
↪Extraction",
    author = "Han, Xu and Gao, Tianyu and Yao, Yuan and Ye, Deming and Liu, Zhiyuan␣
↪and Sun, Maosong",
    booktitle = "Proceedings of EMNLP-IJCNLP: System Demonstrations",
    year = "2019",
    url = "https://www.aclweb.org/anthology/D19-3029",
    doi = "10.18653/v1/D19-3029",
    pages = "169--174"
}
```

It's our honor to help you better explore relation extraction with our OpenNRE toolkit!

CHAPTER 2

Install

2.1 Install as A Python Package

We are now working on deploy OpenNRE as a Python package. Coming soon!

2.2 Using Git Repository

Clone the repository from our github page (don't forget to star us!)

```
git clone https://github.com/thunlp/OpenNRE.git
```

Then install all the requirements:

```
pip install -r requirements.txt
```

Then install the package with

```
python setup.py install
```

If you also want to modify the code, run this:

```
python setup.py develop
```

Note that we have excluded all data and pretrain files for the fast deployment. You can manually download them by running scripts in the `benchmark` and `pretrain` folders. For example, if you want to download FewRel dataset, you can run

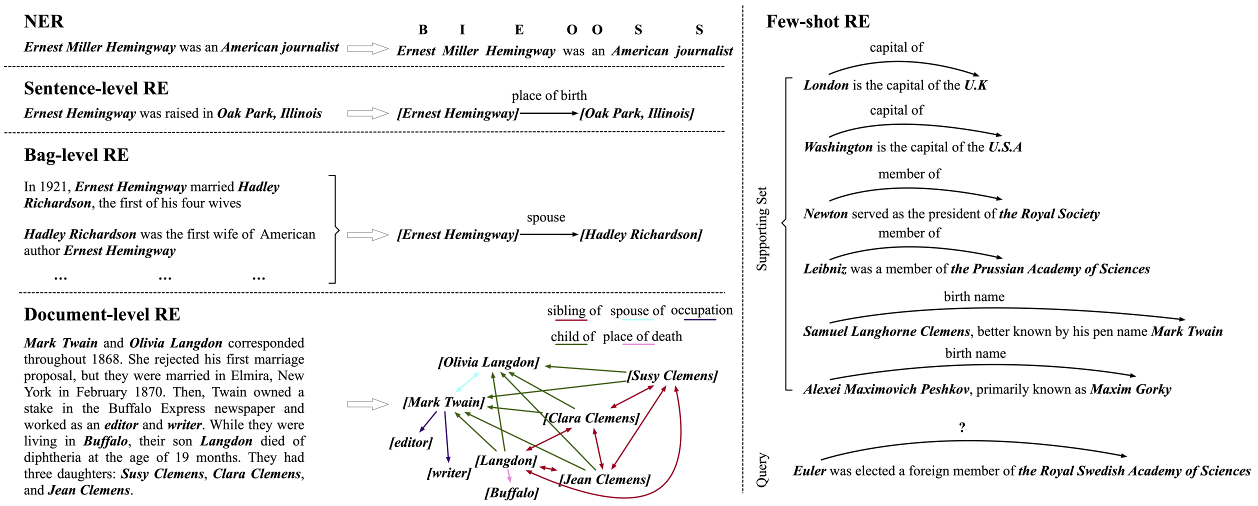
```
bash benchmark/download_fewrel.sh
```

Note: When running our example codes or using a specific pretrain model, the toolkit will automatically download the files it needs.

CHAPTER 3

Settings and Benchmark

There are many settings in relation extraction, targeting different challenges in this area. The following figure gives a vivid example of the formats of these settings.


NER
Ernest Miller Hemingway was an American journalist
B I E O O S S
Ernest Miller Hemingway was an American journalist
Sentence-level RE
Ernest Hemingway was raised in Oak Park, Illinois
place of birth
[Ernest Hemingway] [Oak Park, Illinois]
Bag-level RE
In 1921, Ernest Hemingway married Hadley Richardson, the first of his four wives
Hadley Richardson was the first wife of American author Ernest Hemingway
... ...
spouse
[Ernest Hemingway] [Hadley Richardson]
Document-level RE
Mark Twain and Olivia Langdon corresponded throughout 1868. She rejected his first marriage proposal, but they were married in Elmira, New York in February 1870. Then, Twain owned a stake in the Buffalo Express newspaper and worked as an editor and writer. While they were living in Buffalo, their son Langdon died of diphtheria at the age of 19 months. They had three daughters: Susy Clemens, Clara Clemens, and Jean Clemens.
sibling of
spouse of
occupation
child of
place of death
[Olivia Langdon]
[Susy Clemens]
[Mark Twain]
[Clara Clemens]
[editor]
[Langdon]
[writer]
[Jean Clemens]
[Buffalo]
Few-shot RE
Supporting Set
capital of
London is the capital of the U.K
capital of
Washington is the capital of the U.S.A
member of
Newton served as the president of the Royal Society
member of
Leibniz was a member of the Prussian Academy of Sciences
birth name
Samuel Langhorne Clemens, better known by his pen name Mark Twain
birth name
Alexei Maximovich Peshkov, primarily known as Maxim Gorky
Query
?
Euler was elected a foreign member of the Royal Swedish Academy of Sciences

3.1 Sentence-Level Relation Extraction

It is the most classical relation extraction task. Given a sentence and two tagged entities in the sentence, models neet to classify which relation these two entites have from a predefined relation set.

Two commonly used datasets for sentence-level relation extraction are **SemEval 2010 Task-8** (paper / website) and **TACRED** (paper / website). We also provide a new dataset **Wiki80**, which is derived from **FewRel**.

The statistics of the three datasets are as follows:

Dataset	#class	#instance
SemEval-2010 Task 8	9	6,647
TACRED	42	21,784 (exclude *no_relation*)
Wiki80	80	56,000

Note: Due to the copyright reason, we did not release the TACRED dataset. Please refer to the official site for details.

3.2 Bag-Level Relation Extraction

This setting comes with **distant supervision**. Annotating large-scale relation extraction dataset is labor-intensive and money-consuming. On the other hand, there exist some knowledge graphs (KGs), like FreeBase and WikiData, already including relation triples. By linking entities mentioned in the text to these in the KGs, we can utilize the existing annotations in KGs to automatically label text, which is called distant supervision.

Though making large-scale training data available, distant supervision inevitably brings noise. For the denoising purpose, multi-instance multi-label setting was proposed. Instead of predicting relations for each sentence, in the multi-instance multi-label setting, models need to predict labels for each entity pair (which may have many sentences). Sentences share the same entity pair are called a “bag”.

The most used dataset for distant supervision is **NYT10** (paper / website). We also provide **NYT10m** and **Wiki20m**, two distant supervised datasets with **manually** annotated test sets (paper: TBD).

Dataset	#class	#instance
NYT10	53	694,491
NYT10m	25	474,059
Wiki20m	81	901,314

3.3 Document-Level Relation Extraction

You can refer to DocRED for details.

3.4 Few-Shot Relation Extraction

Inspired by the facts that human can grasp new knowledge with only a handful of training instances, few-shot learning was proposed to explore how models can fast adapt to new tasks. **FewRel** is a large-scale few-shot relation dataset (paper / website). The way to sample data and evaluate models in the few-shot setting is quite different from others, and you can refer to the paper for details. Statistics of the dataset are shown in the following table:

Dataset	#class	#instance
FewRel	100	70,000

CHAPTER 4

Pretrain

We now have the following released models:

- **wiki80_cnn_softmax**: trained on **wiki80** dataset with a CNN encoder.
- **wiki80_bert_softmax**: trained on **wiki80** dataset with a BERT encoder.

4.1 How to Use Them

After installing OpenNRE, open Python and load OpenNRE:

```
>>> import opennre
```

Then load the model with its corresponding name:

```
>>> model = opennre.get_model('wiki80_cnn_softmax')
```

It may take a little while to download the model. After loading it, you can do relation extraction with the following format:

```
>>> model.infer({'text': 'He was the son of Máel Dúin mac Máele Fithrich, and␣
↪grandson of the high king Áed Uaridnach (died 612).', 'h': {'pos': (18, 46)}, 't': {
↪'pos': (78, 91)}})
```

The `infer` function takes one dict as input. The *text* key represents the sentence and *h* / *t* keys represent head and tail entities, in which *pos* (position) should be specified.

The model will return the predicted result as:

```
('father', 0.5108704566955566)
```

CHAPTER 5

User Guide

5.1 Use our pre-trained models

You can use our pre-trained models:

- **wiki80_cnn_softmax**: trained on **wiki80** dataset with a CNN encoder.
- **wiki80_bert_softmax**: trained on **wiki80** dataset with a BERT encoder.

In the following way:

1. Load the model

```
>>> import opennre
>>> model = opennre.get_model('wiki80_cnn_softmax')
```

2. Use `infer` to do relation extraction

```
>>> model.infer({'text': 'He was the son of Máel Dúin mac Máele Fithrich, and␣
↪grandson of the high king Áed Uaridnach (died 612).', 'h': {'pos': (18, 46)}, 't': {
↪'pos': (78, 91)}})
```

The model will return the predicted result as:

```
('father', 0.5108704566955566)
```

5.2 Train your own models

We have examples to train relation extraction models in the `example` folder:

- `train_supervised_cnn.py`: train a supervised CNN model.
- `train_supervised_bert.py`: train a supervised BERT model.
- `train_bag_pcnn_att.py`: train a bag-level PCNN-ATT model.

CHAPTER 6

FAQ

Coming soon . . .