CSCI 699: Assignment #2
Relation Extraction

Last updated: October 4, 2018

1 Overview

In this assignment, you will implement a model for Relation Extraction (RE) using deep learning techniques. To start your assignment, please download the data and starter code from dropbox.

You should complete this assignment individually and submit the assignment to TA (Hongtao Lin) by email (`lin498@usc.edu`) before **10/24 11:59 PM PST**. For detailed instructions on submission, please refer to Sec. 3.

1.1 Introduction to RE

According to [1], Relation Extraction is usually referred to as either global-level RE or mention-level RE. We will perform mention-level RE in this assignment: given an entity pair and a sentence that contains it, the model identifies the relation for this entity pair (in a predefined relation set). Mention-level RE is context-dependent: The sentence "`Tim Cook joined Apple in March 1998`" does not indicate the relation `CEO(Tim Cook, Apple)`, although this relation may be expressed by other sentences.

1.2 Task

You are asked to implement a model for SemEval 2010 Task 8 [2]. Given a sentence and two tagged nominals, your task is to predict the relation between those nominals and the direction of the relation. By "direction", it means that we treat the relation `Rel(e1,e2)` and `Rel(e2,e1)` as two different labels. One additional label `Other` is assigned to an instance if the entity pair expresses no relation of our concern. This label is not regarded as a valid relation. In total, there are 9 relations of our concern (e.g. `Cause-Effect`, `Component-Whole`), as documented in [2]. These 9 relations correspond to 18 labels when directionality is taken into consideration.

1.3 Dataset

The whole dataset is divided into `train` and `test` set, with 8,000 and 2,717 instances respectively. We remove the labels from `test` for final evaluation. An example of training instance is as follows:

```
42  "The <e1>transmitter</e1> was discovered inside a bed settee <e2>suite</e2> on
which he had been sitting."
Content-Container(e1,e2)
Comment:  prototypical example
```

The first line consists of an instance id and a sentence, with two entities explicitly marked in it. This is followed by a label for this instance and a line of comment.

1.4 Evaluation

The metric for our final evaluation is **macro-averaged** F_1-score for (9+1)-way classification, taking directionality into account.

For example, suppose you have a list of ground-truth labels y and predicted labels $\hat{y}$ as follows (we abbreviate labels from `Cause-Effect(e2,e1)` to `C-E2` for simplicity):

y	C-E2	C-E1	P-P2	O	E-D1	P-P1	E-D2
$\hat{y}$	C-E1	C-E1	O	O	P-P1	P-P1	C-E2

To calculate the final score, we first recognize all relations in ground-truth labels. There are three of them: `C-E`, `P-P` and `E-D` (with direction is ignored). Then we calculate precision, recall and F_1 for these labels. Take `C-E` for example, we have one instance with both relation and direction predicted match. So its precision, recall is $\frac{1}{3}$ and $\frac{1}{2}$, respectively.

Proceeding this way, we know F_1 score for `C-E`, `P-P` and `E-D` are $\frac{2}{5}$, $\frac{1}{2}$ and 0. The macro-averaged result would be $\frac{3}{10}$.

2 Methodology

You should implement a relation extraction model using deep learning techniques (of your own choice) and any external resource that is helpful for the task. Here we provide some initial ideas for you to explore more upon:

1. **Model architecture**. CNN [3], RNNs [4], Attention mechanism [4].
2. **Additional Information**. Position embeddings [5], Entity masking [4].

Note that there are relevant works that focus on distantly-supervised RE, which is outside the scope of our concern. But you can definitely borrow some ideas from them (e.g, data preprocessing, additional information introduced, model architecture).

2.1 Development Pipeline

For reference, here's a pipeline for students to kick off this assignment:

1. Load raw data into desired format (e.g. transform into a triple: (entity1, entity2, sentence)). Remember to spare some data in `train` set as `dev` set, or perform cross validation in later steps.
2. Perform data preprocessing and feature engineering. Since our provided data only contains texts, you should consider getting additional information from Part-of-Speech (POS) tagging, Dependency parsing. You are free to use any open source tools such as spaCy [1] and Stanford CoreNLP Toolkit [2].
3. Implement a baseline model. You can choose either CNN or RNN as a baseline model to start with.
4. Modify the baseline model. Refer to relevant works and try to implement their model.
5. Perform error analysis and explore more on model construction.

3 Submission Guideline

3.1 Deliverable

The deliverable for this assignment consists of three parts. Each part will be evaluated and taken into consideration for the final score:

[1] `https://github.com/explosion/spaCy`
[2] `https://stanfordnlp.github.io/CoreNLP/`

1. **Code repository (30 points)**. The code to replicate your results (put into a directory called `code/`). There are several things to note about code repository:
   (a) Please document the instructions on how to reproduce the results in `README.md`.
   (b) You should include the external resources that are necessary to reproduce the result. If it's too large (e.g. pretrained word vector), you can provide specific instructions on how to access it in `README.md`.
   (c) The model binary file and checkpoints shall be removed from code.

   We will evaluate your code based on the completeness (all necessary code and data should be included) and the quality (good coding style, enough comments and well-organized structure).

2. **Prediction result (30 points)**. A prediction result for `test` set (named as `test_output.txt`).

   Each line in your result file should be in format: "<SENT_ID> <RELATION>" with a tab as separator. For example, the first few lines of your result can be:

   ```
   1  Component-Whole(e2,e1)
   2  Other
   3  Instrument-Agency(e2,e1)
   ...
   ```

   You will be evaluated based on macro-averaged F_1 score in `test` set. Basically, there is a positive correlation between F_1 score and points you get.

3. **Report (40 points)**. A report named as `report.pdf`.

   In this assignment, you are encouraged to explore and analyze different techniques in RE. The report is a place to write down your investigation process and interesting findings. Besides, you are required to include the following information in your report: (1) Open source libraries used. (2) References.

   We will evaluate your report based on the amount of work, the depth of analysis and clarity of writing.

The above deliverable should be included into a single zip file named as `FirstName_LastName_HW2.zip`.

3.2 Late Policy

According to our syllabus, we allow 4 days in total for the assignments and project proposal/survey. So if the student already used up all 4 days in the first assignment, the late submission in this assignment will be penalized. 10 points will be deducted as penalty.

References

[1] Sachin Pawar, Girish K Palshikar, and Pushpak Bhattacharyya. Relation extraction: A survey. *arXiv preprint arXiv:1712.05191*, 2017.

[2] Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, pages 94–99. Association for Computational Linguistics, 2009.

[3] Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2335–2344, 2014.

[4] Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D Manning. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45, 2017.

[5] Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762, 2015.