CS6320: Natural Language Processing

Project: Semantic Relation Classification (100 points)

Due date: 05/05/2023 2:00pm

For this project, you will design and implement a semantic relation classifier on the SemEval-2010 Task 8 dataset using either a model of your choice or from one of the models listed in this document.

Data

To train and evaluate your semantic relation classifier, you will use the SemEval 2010 - Task 8 dataset. To simplify the problem, we will use a relation set that contains 5 relations, i.e. **Cause-Effect**, **Component-Whole**, **Product-Producer**, **Instrument-Agency**, and Other-Relation. The dataset contains a total of 10 relations. Kindly convert the dataset to a 5 relation dataset by tagging all relations other than **Cause-Effect**, **Component-Whole**, **Product-Producer**, and **Instrument-Agency** as Other-Relation. The raw dataset is provided as **dataset.zip** via elearning. We advise you to kindly read the dataset paper linked before and here to understand the dataset thoroughly.

Tasks

1. Preprocess the dataset to contain only 5 relations. Print the dataset statistics post-preprocessing.
2. Train a model on the preprocessed dataset. You can either use your own model or one of the models listed below. Please keep in mind the memory available for training a model when selecting your model. NOTE, you need to train the model and not use a model pre-trained on the SemEval dataset.
3. Report the training and validation statistics of the trained model. Although you are free to use any method to create a validation set, a simple 80:20 or 90:10 split of the training set should suffice.
4. Evaluate the trained model on the test set. Report the evaluation statistics and also carry out manual error analysis on 50 random test sentences.

You will write all your code in Google Colab: a free notebook server provided by Google to run your Python code. Colab comes bundled with a GPU so you can use it to make your code run fast (faster than a CPU or even the csgrads1 server).

Sample models

1. Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification. paper code
2. Simple Relation Extraction with a Bi-LSTM Model. code
3. Matching the Blanks: Distributional Similarity for Relation Learning. paper code

Submission

Submit the following bundled into a single zip file via eLearning:

1. Your code file(s)
2. A Readme giving clear and precise instructions on how to run the code.
3. A plaintext file showing preprocessed dataset statistics, training and validation statistics, and testing statistics.
4. A short report describing your results, observations drawn from the manual error analysis of 50 test sentences and what lessons you learned.

Useful Links

Check out these additional links to help you with your project:

1. https://nlpprogress.com/english/relationship_extraction.html
2. https://paperswithcode.com/sota/relation-extraction-on-semeval-2010-task-8