This assignment is due **Wednesday, December 15 at 11:59PM**.

Goals

Through this assignment you will:

- Explore issues in shallow discourse parsing.
- Gain familiarity with the Penn Discourse Treebank and CoNLL data.
- Gain some further familiarity with vector-based word embeddings
- Implement a relation sense classification system.

Background

Please review the class slides and readings in Jurafsky and Martin 3rd ed. Chapter 22 on shallow discourse parsing and the Penn Discourse Treebank.

For additional information on the CoNLL16 json file format for shallow discourse parsing data: the CoNLL16 data format tutorial

Implementing Coherence Relation Sense Classification for Shallow Discourse Parsing

Based on the examples in the text, class slides, and other resources, implement a program to perform coherence relation sense classification, one of the steps in shallow discourse parsing. Specifically, your program should:

1. Read in Glove embedding vectors from the provided file.
2. Load training and test coherence relation classification data from a provided subset of CoNLL resource files
3. Create training and test classification vectors from the data. For each shallow discourse parsing instance:
    1. For the Arg1 and Arg2:
        1. Tokenize the raw text, ideally using NTLK.word_tokenize()
        2. Using the corresponding Glove embeddings of the tokens, create averaged vector representation of the Arg
    2. Concatenate the Arg1 and Arg2 representation to make the classification vector
4. Write the training and test instances to respective files in comma separated value format, with the sense of the instance as the last element in each line
5. Train a classifier on the training instances. You can use whatever method for classification you'd like, including any of the classifiers in scikit-learn
6. Test on the test instances. Writing to the output file
    1. The overall per-class F-measure
    2. For each test instance: true_label\tpredicted_label

Programming

Create a program **hw9_coherence.sh** that implements the coherence relation sense classification as specified as above invoked as:

hw9_coherence.sh <glove_embedding_file> <relation_training_data_file> <relation_testing_data_file> <training_vector_file><testing_vector_file> <output_result_file>

- <glove_embedding_file>
  - This string should specify the location of the Glove Embedding file.
- <relation_training_data_file>
  - This is the name of the file that contains the json formatted training data for the coherence relation sense classification. The format is the same as the CoNLL16 relations,json file. Please see the file here.
- <relation_testing_data_file>
  - The name of the input file holding the json formatted testing data for the coherence relation sense classification, in the same format as above
- <training_vector_file>
  - The name of the output file for your training vector representations. These should be one instance per line, comma separated values, with the coherence relation sense last.
- <testing_vector_file>
  - The name of the output file for your training vector representations, similar to the file above.
- <output_filename>
  - This is the name of the file to which you should write your classification results.

Files

All files are found in /dropbox/21-22/571/hw9/ on patas:

Test, Gold Standard, and Example

- **glove.6B.50d.txt**
  - 50 dimensional Glove embeddings trained on Wikipedia and Gigaword: a small Glove model
- **relations_train.json**
  - File of shallow discourse parsing instances to be used for training the classifier.
- **relations_test.json**
  - File of shallow discourse parsing instances to be used for testing the classifier.
- **train_examples.txt**
  - Example train instances; this shows the format that your training_vector_file should take.
- **example_output.txt**
  - Example output file; this shows the format that your output_filename should take. (NB: you do not need to match the predicted labels here, since you can use any reasonable classification method.)

Submission Files

- **hw9.tar.gz**: Tarball containing the following:
  - **hw9_coherence.sh**: Program which implements and evaluates the embedding based coherence relation sense classification task.
  - **hw9_training_vectors.txt:** Vector representation of training instances
  - **hw9_test_vectors.txt:** Vector representation of testing instances
  - **hw9_output.txt**: Classification scores and paired gold-predicted outputs.
- **readme.{txt|pdf}**: Write-up file
  - This file should describe and discuss your work on this assignment. Include problems you came across and how (or if) you were able to solve them, any insights, special features, and what you learned. Give examples if possible. If you were not able to complete parts of the project, discuss what you tried and/or what did not work. This will allow you to receive maximum credit for partial work.

[Back to Top]