SemEval-2015 Task 1: Paraphrase and Semantic Similarity in Twitter (PIT)

Updated: Jan 15, 2015 (packed up after the official evaluation)

= ORGANIZERS =

- Wei Xu, University of Pennsylvania
- Chris Callison-Burch, University of Pennsylvania
- Bill Dolan, Microsoft Research

= REFERENCES =

Please cite the following papers if you use the data or code accordingly:

all papers are included in this package:

- paper about the dataset, baselines, and the MultiP model (multiple-instance learning paraphrase):

  @article{Xu-EtAl-2014:TACL, author = {Wei Xu and Alan Ritter and Chris Callison-Burch and William B. Dolan and Yangfeng Ji}, title = {Extracting Lexically Divergent Paraphrases from {Twitter}}, journal = {Transactions of the Association for Computational Linguistics}, volume = {}, number = {}, year = {2014}, pages = {}, publisher = {Association for Computational Linguistics}, url = {http://www.cis.upenn.edu/~xwe/files/tacl2014-extracting-paraphrases-from-twitter.pdf} }

- overview paper of the shared task:

```
@inproceedings{xu2015semeval,
  author    = {Wei Xu and Chris Callison-Burch and William B. Dolan},
  title     = {{SemEval-2015 Task} 1: Paraphrase and Semantic Similarity in {Twitt
  booktitle = {Proceedings of the 9th International Workshop on Semantic Evaluatic
  year      = {2015}
}
```

- paper about the dataset:

```
@phdthesis{xu2014data,
  author = {Xu, Wei},
  title = {Data-Drive Approaches for Paraphrasing Across Language Variations},
  school = {Department of Computer Science, New York University},
  year = {2014},
  url = {http://www.cis.upenn.edu/~xwe/files/thesis-wei.pdf}
}
```

= TRAIN/DEV/TEST DATA =

The dataset contains the following files:

```
./data/dev.data   (4727 sentence pairs)
./data/test.data  (972 sentences pairs)
./data/test.label (a separate file of labels only, used by evaluation scripts)
```

Both data files come in the tab-separated format. Each line contains 7 columns:

```
Topic_Id | Topic_Name | Sent_1 | Sent_2 | Label | Sent_1_tag | Sent_2_tag |
```

The "Trending_Topic_Name" are the names of trends provided by Twitter, which are not hashtags.

The "Sent_1" and "Sent_2" are the two sentences, which are not necessarily full tweets. Tweets were tokenized by Brendan O'Connor et al.'s toolkit (ICWSM 2010) and split into sentences.

The "Sent_1_tag" and "Sent_2_tag" are the two sentences with part-of-speech and named entity tags by Alan Ritter et al.'s toolkit (RANLP 2013, EMNLP 2011).

The "Label" column for *dev/train data* is in a format like "(1, 4)", which means among 5 votes from Amazon Mechanical turkers only 1 is positive and 4 are negative. We would suggest map them to binary labels as follows:

```
paraphrases: (3, 2) (4, 1) (5, 0)
non-paraphrases: (1, 4) (0, 5)
debatable: (2, 3)  which you may discard if training binary classifier
```

The "Label" column for *test data* is in a format of a single digit between between 0 (no relation) and 5 (semantic equivalence), annotated by expert.
We would suggest map them to binary labels as follows:

```
paraphrases: 4 or 5
non-paraphrases: 0 or 1 or 2
debatable: 3   which we discarded in Paraphrase Identification evaluation
```

We discarded the debatable cases in the evaluation of Paraphrase Identification task, but kept them in the evaluation of Semantic Similarity task.

= EVALUATION =

There are two scripts for the official evaluation:

```
./scripts/pit2015_checkformat.py (checks the format or the system output file)
./scripts/pit2015_eval_single.py (evaluation metrics)
```

The participants are required to produce a binary label (paraphrase) for each sentence pair, and optionally a real number between 0 (no relation) and 1 (semantic equivalence) for measuring semantic similarity.

The system output file should match the lines of the test data. Each line has 2 columns and separated by a tab in between, like this: | Binary Label (true/false) | Degreed Score (between 0 and 1, in the 4 decimal format) | if your system only gives binary labels, put "0.0000" in all second columns.

The output file names look like this: PIT2015_TEAMNAME_01_nameofthisrun.output PIT2015_TEAMNAME_02_nameofthisrun.output

= BASELINES & STATE-OF-THE-ART SYSTEMS =

There are scripts for two baselines:

```
./scripts/baseline_logisticregression.py
```

and their outputs on the test data, plus outputs from two state-of-the-art systems:

```
./systemoutputs/PIT2015_BASELINE_02_LG.output
./systemoutputs/PIT2015_BASELINE_03_WTMF.output
./systemoutputs/PIT2015_BASELINE_04_MultiP.output
```

(1) The logistic regression (LG) model using simple lexical overlap features:

It is our reimplementation in Python. This is a baseline originally used by Dipanjan Das and Noah A. Smith (ACL 2009): "Paraphrase Identification as Probabilistic Quasi-Synchronous Recognition".

To run the script, you will need to install NLTK and Megam packages: http://www.nltk.org/_modules/nltk/classify/megam.html http://www.umiacs.umd.edu/~hal/megam/index.html

If you have troubles with Megam, you may need to rebuild it from source code: http://stackoverflow.com/questions/11071901/stuck-in-using-megam-in-python-nltk-classify-maxentclassifier

Example output, if training on train.data and test on dev.data will look like:

```
Read in 11513 training data ...  (after discarding the data with debatable cases)
Read in 4139 test data ...       (see details in TRAIN/DEV DATA section)
PRECISION: 0.704069050555
RECALL:    0.389229720518
F1:        0.501316944688
ACCURACY:  0.725537569461
```

The script will provide the numbers for plotting precision/recall curves, or a single precision/recall/F1 score with 0.5 cutoff of predicated probability.

(2) The Weighted Matrix Factorization (WTMF) model is a unsupervised approach developed by Weiwei Guo and Mona Diab (ACL 2012): "Modeling Sentences in the Latent Space" Its code is available at: http://www.cs.columbia.edu/~weiwei/code.html

(3) The Multiple-instance Learning Paraphrase model (MultiP) is a supervised approach developed by Wei Xu et al. (TACL 2014): "Extracting Lexically Divergent Paraphrases from Twitter" Its code is available at: http://www.cis.upenn.edu/~xwe/multip/