Statistical AnnCorra Tagger

Abhigyan Ghosh

Theory

We are using a statistical model to calculate the count of all unigrams, bigrams and trigrams. The the tag for each word is then calculated based on trigram count. If a trigram doesn't exist, then it searches for bigrams and then unigrams.

Requirements

- `python 3.0^`

Files

- `statistical.py` - main tagger file
- `split.py` - split tags and words into separate files
- `train.txt` - data file formed by concatenation for training
- `test.txt` - data file formed by concatenation for testing
- `src-train.txt` - source file for training
- `tgt-train.txt` - target file for training
- `src-test.txt` - source file for testing
- `tgt-test.txt` - target file for testing

Steps

1. Concatenate data files into a single file using from the `cat train/*.dat > train.txt`, `cat test/*.dat > test.txt` and `cat dev/*.dat > dev.txt`
2. Run the training file using `python train_stat.py`. The file takes a few minutes to run. The output is stored in `stat_out.txt`.
3. Run `python test_stat.py` to evaluate the output for `stat_out.txt` with `tgt-test.py`

Accuracy

Out of `40759`, `10412` tags were inaccurately tagged in `tgt-test.txt` which brings the total accuracy is `74.45472165656665`