Sarika Halarnakar - shalarn1
Sindhuula Selvaraju - sselvar4
Machine Translation Assignment 6: Inflection

Contents of the Repository:

1. data : This directory contains the data files given to us from CLS Grid
2. scripts: This directory contains the script files used to compute inflection
3. inflect_original: This has not been modified by us
4. inflect_bigram: This file tries to find the best inflection based on the bigram model without backoff
5. inflect_bigram_backoff: This file tries to find the best inflection based on the bigram model with backoff
6. inflect_trigram: This file tries to find the best inflection based on the trigram model
7. inflect_pos: This file adds to the trigram model by also checking tags at the unigram level
8. results : The results for the various models we tried

Part 1: Bigram Model

Usage: (In inflect)

```
cat data/dtest.lemma | ./scripts/inflect_bigram | ./scripts/grade
```

Part 2: Bigram Model with BackOff

Usage: (In inflect)

```
cat data/dtest.lemma | ./scripts/inflect_bigram_backoff | ./scripts/grade
```

Part 3: Trigram Model

Usage: (In inflect)

```
cat data/dtest.lemma | ./scripts/inflect_trigram | ./scripts/grade
```

Part 4: Trigram Model with POS tagging

Usage: (In inflect)

```
cat data/dtest.lemma | ./scripts/inflect_pos -d data/dtest | ./scripts/grade
```

Note : For dev + test

Usage: (In inflect)

```
cat data/dtest.lemma | ./scripts/inflect_<file> > result_<file>
```

```
cat data/etest.lemma | ./scripts/inflect_<file> -d data/etest >>
result_<file>
```

Description:

We tried different ways to improve the accuracy in finding inflected forms of words.
The bigram model simply found if the word along with the previous word was also found in the training data and if it was found it returned it otherwise it returned the word as it is.

The bigram model with backoff not only checked if the bigram exists in the training data but if it does not exist it checks if the unigram exists in the training data. If the unigram exists then the most frequently seen inflected form is returned otherwise the word is returned as such.

The trigram model is similar to the bigram model with backoff but for trigrams. It checks if the trigram exists in the training data, if not it backsoff to check if the bigram exists or else backsoff to check for the unigram.

The POS model is similar to trigram model but if the model backs off to the unigram form it checks for the most frequent inflected form for the (word, tag) pair. If the tag does not match then it returns the word without inflection.

Results:

1. Inflected Bigram

Dev

```
macintosh-3:inflect chibb9$ cat data/dtest.lemma | ./scripts/inflect_bigram | ./scripts/grade
38737 / 70974 = 0.55
```

Dev + Test

```
cat data/dtest.lemma | ./scripts/inflect_bigram > result_bigram
cat data/etest.lemma | ./scripts/inflect_bigram >> result_bigram
```

NOTE: File size 1.1MB so unable to upload.
NOTE 2: Tried File Size workaround, unable to see test scores.

2. Inflected Bigram with Backoff

Dev

```
cat data/dtest.lemma | ./scripts/inflect_bigram | ./scripts/grade
46614 / 70974 = 0.66
```

Dev + Test

```
cat data/dtest.lemma | ./scripts/inflect_bigram > result_bigram_backoff
cat data/etest.lemma | ./scripts/inflect_bigram >> result_bigram_backoff
0.585
```

3. Inflected Trigram with Backoff

Dev

```
cat data/dtest.lemma | ./scripts/inflect_trigram | ./scripts/grade
46729 / 70974 = 0.66
```

Dev + Test

```
cat data/dtest.lemma | ./scripts/inflect_trigram > result_trigram
cat data/etest.lemma | ./scripts/inflect_trigram >> result_trigram
0.586
```

4. Inflected Trigram with POS

Dev

```
cat data/dtest.lemma | ./scripts/inflect_pos | ./scripts/grade
46719 / 70974 = 0.66
```

Dev + Test

```
cat data/dtest.lemma | ./scripts/inflect_pos > result_trigram
cat data/etest.lemma | ./scripts/inflect_pos -d data/etest >> result_trigram
0.586
```

Conclusion:

As we can see the back-off Bigram model performs much better than using just bigram model. The Trigram model further increases this accuracy. Using POS tagging was not as successful as the Trigram model but it was still better than Bigram.