CIS 526 Term Project: Domain Adaptation

April 15, 2014

1 Domain Adaptation

Statistical machine translation systems are trained on large corpora of training data. These corpora are often gathered from domains in which parallel text is readily available, such as news sources or parliamentary proceedings. Training with large amounts of parallel text gives the system more information on which to build a language model and translation model, but it also may lead to models that are biased toward the domain of the training data. These models may not be suited for translating sentences that come from vastly different domains. For example, some domains (such as the technological and medical fields) have special terminology that would not be found in the training domain; these words and phrases will be out-of-vocabulary. Other phrases may appear in the translation model, but in the context of the new domain the system should prefer a much less probable translation over the actual highest-ranking translation. For example, "bank" may be translated to French as "banque" in an economic domain, while a geographic context might prefer the translation "rive".

Domain adaptation is important in the field of statistical machine translations because there are many domains for which a large amount of training data would not be available. The ability to translate text from these domains by adapting a system trained on large amounts of out-of-domain data would solve this problem. Furthermore, any machine translation system that intends to be of general use, such as Google Translate, would need to be able to deal with inputs from a wide variety of domains.

2 Getting Started

First, you should download Joshua from http://joshua-decoder.org/ and complete the setup instructions. You might want to try running one of the examples provided by Joshua, to make sure that everything is set up correctly.

In the *models* directory, we have provided a number of translation models and language models for you to work with. The models were trained using the Joshua machine translation system. Feel free to train new models by changing around the parameters in the *train* script. This script runs part of the Joshua machine translation pipeline to train the models. The translation model is saved as grammar.gz and the language model is saved in two formats as lm.gz and lm.kenlm. The default implementation trains models using the *train* script. The trained models are saved in a new directory, *default*. Next, the default implementation uses the parameters in the newly-created file *default/joshua.config* to decode the test set. This configuration file specifies the tuned feature weights for the translation model, along with a number of other parameters for controlling the output of the decoder. To translate the test sentences, run the command:

```
./decode.sh default/joshua.config > output
```

This runs another part of the Joshua pipeline to decode the test set, using the parameters specified in the given configuration file. Normally the entire pipeline can be run at once, but here we separate the two tasks. Because many solutions for domain adaptation involve modifying either the language model or the translation model, or both, it is convenient to be able to run the decoder separately.

The command

```
cat output | grade
```

calculates the BLEU score for half of the test set sentences, for which translations are provided; the other half will be used for scoring on the leaderboard.

3 The Challenge

Your goal is to improve the BLEU score of the translations of the test set. An easy way to find small increases in BLEU is to experiment with different combinations of the old- and new-domain models, and by giving the models different weights in the configuration file. However, there are many approaches to domain adaptation:

- Look for different ways to combine the in-domain and out-of-domain data
- Use techniques from information retrieval
- Implement a linear programming approch to modifying the translation model
- Mine unseen words to decrease OOV ocurrences
- Use word or phrase sense disambiguation to find translations that match the context of the domain
- Try "frustratingly easy" domain adaptation

Or look for another idea! The only restriction: do not simply download the rest of the EMEA corpus to use as training data. The point of the challenge is to see what you can do with a small amount of in-domain data.