LING572 Hw6: Beam search
Due: 11:45pm on Feb 18, 2010

The example files are under dropbox/09-10/572/hw6/examples/.

Q1 (75 points): Write a script, **beamsearch_maxent.sh**, that implements the beam search for POS tagging.

- The format is: beamsearch_maxent.sh test_data boundary_file model_file sys_output beam_size topN topK
- test_data has the same format as before (e.g., **ex/test.txt**): "instanceName goldClass f1 v1 f2 v2 ..." (goldClass is the POS tag according to the gold standard)
- boundary_file: the format of boundary_file is one number per line, which is the length of a sentence (e.g., **ex/boundary.txt**).
- model_file is a MaxEnt model in text format (e.g., **m1.txt**).
- sys_output has the same format as before except that for each instance you need to output only the best class (not all the classes) and its probability (e.g., **ex/sys**); that is, the format is "instanceName goldClass sysClass prob", where *sysClass* is the POS tag y chosen by the system, and *prob* is $P(y \mid x)$.
- topN: When expanding a hypothesis h, create the new hypotheses only for the *topN* labels of the current instance.
- beam_size is the max ratio between the prob of the best hypothesis and the prob of kept hypotheses: that is, a kept hypothesis should satisfy $lg(prob) + beam\_size \geq lg(max\_prob)$, where *max_prob* is the prob of the best hypothesis for the current position.
- topK is the max number of hypotheses kept alive at each position after pruning.

Note:

- A *hypothesis* in the beam search corresponds to a node in the beam search tree. And for more about how beam search works and the meaning of beam_size, topN and topK, see slide #41-42 in 2_9_MaxEnt_p2.pdf.
- Remember that the feature vectors in the test_data do not include t_{i-1}=tag (e.g., **prevT=NN**) and t_{i-2} t_{i-1}=$tag_{i-2} + tag_{i-1}$ features (e.g., **prevTwoTags=JJ+NN**), because the tags of the previous words are not supposed to be known for the test data beforehand. You need to add those features to the feature vectors before calling the model to classify the current instance based on the current hypothesis.
  - For instance, suppose the current instance is "instanceName goldTag f1 v1 f2 v2 ...", and in the current hypothesis the system tags the previous word as NN and the word before the previous word as JJ. You need to add "prevT=NN" and "prevTwoTags=JJ+NN" to the feature vector in order to determine the top tags of the current instance under the current hypothesis.

- When you add these two types of features, only add the ones that appear in the model file. If a feature (e.g., prevTwoTags=NN+RB) does not appear in the model file, that means that the tag bigram does not appear in the training data. In that case, do not add the feature to the feature vector, as the model does not contain the weights for the corresponding feature functions.
- For your convenience, the list of these two types of features in the **m1.txt** is stored in **feats_to_add**. Your code should not read in a file like **feats_to_add** as it comes from the model file. This file is there just to show you what POS tags are used in the training data.

Q2 (20 points): Run beamsearch_maxent.sh with sec19_21.txt as the test data, m1.txt as model_file, sec19_21.boundary as the boundary file. Fill out Table 1.

Table 1: Beam search

beam_size	topN	topK	Test accuracy	Running time
0	1	1		
1	3	5		
2	5	10		
3	10	100		

Q3 (5 points): What conclusions can you draw from the experiments?

Note: Please test your code thoroughly using ex/test.txt as the test file, ex/boundary.txt as boundary file, m1.txt as the model file. Then run the code on the real data set with the (0, 1, 1) setting, and record the time it takes. The running time for other settings could be much longer.

Submission: Submit a tar file via CollectIt. The tar file should include the following.

- If your team has two people, please submit only one copy. In your note file, please list the names of team members.
- In your note file hw6.*, include your answers to the questions, and any notes that you want the TA to read.
- The source code and shell script for Q1.