Applied Language Technology - Assignment 3

Tushar Nimbhorkar 11394110
Diede Rusticus 10909486

October 2017

1 Introduction

The task is to calculate translation costs for 500 sentences; German to English. Of each sentence the decoding trace is given. This enables us to calculate the following cost per phrase transition:

1. translation model cost
2. language model cost
3. reordering (distortion) cost
4. linear distortion cost
5. phrase penalty

The phrase penalty is just a constant of -1 per phrase transition. We use two types of distortion costs because they represent different things. Namely, the reordering cost takes the actual alignments into accounts of every phrase-pair and uses the probabilities of the alignment orientations (monotone, discontinuities , etc.). The linear distortion however, only looks whether the translated phrases are ordered as in the source language. In the end, we sum all the costs per phrase transition to get one translation cost per sentence.

2 Implementation

2.1 Lamdas

We first define our lambda values.

These lambdas are the weights per cost. In this case, we set them all to '1', as fine-tuning these weights is beyond the scope of this assignment.

2.2 Read data

We read the data into dictionaries.

- The phrase table is captured in a dictionary where the keys are tuples of phrase-pairs and the values the probabilities ($p(f|e)$, $lex(f|e)$, $p(e|f)$, $lex(e|f), wordpenalty$).
- The dictionary of the English language model has as keys the English phrases and as values the tuples of the probability and the backoff. Also the minimum value of the probabilities is saved. We will use this value later on.
- The keys of the dictionary of the reordering model are the phrase-pairs and the values the probabilities of the monotone, swap, and discontinuous (LR and RL).

2.3 Costs

We then calculate the individual costs per phrase transition. This section explains how to get all the individual costs per phrase of the trace. These (weighted) costs are summed for every phrase and the phrase costs are summed for the entire sentence.

- The **translation cost** is calculated by summing the probabilities and the word penalty. In order to do so, we took the log (base 10) of the probabilities. The elements of this sum all have a weight. But again, in this case all set to '1'. If the phrase-pair is not found in the translation model we set the translation cost to '-1'.
- The **language model** cost is calculated by taking n-grams. The language model of a phrase is the sum of $p(w_n|w_1, ..., w_{n-1})$ for every w_n.

  The unigram of the first word is not taken into account, because there is no preceding word. We loop over the words and keep the history. The start tag <s> and end tag < \s> are added respectively in front of the first phrase and at the end of the last phrase. Based on the history, the language model and the current word, we calculate a word cost which is summed to the total phrase cost.

  The word cost is calculated by getting the n-gram (consisting of the history and the current word) and see if it exists in the language model.

  If it does, the probability (not the backoff) is taken as the word cost. If it does not, the word cost is checked recursively by shortening the history by one word every time and check if the n-gram now does exist in the language model. If an n-gram is found, now the backoff probability is taken as the word cost.

  If no n-gram was found in the language model, the minimum language model probability is taken as the word cost.

- The **reordering (distortion)** cost is calculated by multiplying the R-L costs with the L-R costs. Of this product the log (base 10) is taken, in order to sum it with the other costs that are also in log space.

  If the current phrase is the first phrase of the trace, the R-L cost is set to the monotone r-l probability (taken from the reordering dictionary). In the other cases, the R-L cost is either the monotone r-l, swap r-l or discontinuous r-l probability by checking the alignments of the previous phrase.

  The same holds for the L-R cost but then taking the next phrase instead of the previous.

  If the phrase pair is not found in the reordering model, the cost is set to -1.

- The **linear distortion** cost is defined by:

$$h_{LD}(s_i, s_j) = -1 \times |first\_pos(s_j) - last\_pos(s_i) - 1|$$

  This means that when phrase B in the source sentence comes after A, and the translations $\widetilde{B}$ also comes after $\widetilde{A}$ in the target sentence, the linear distortion is 0. However, when there is one phrase in between $\widetilde{A}$ and $\widetilde{B}$, the linear distortion is -1.

3 Examples

das war keinesfalls einfach ! |||
this was no mean feat ! |||
lm: -12.516103
tm: -5.71333300592
rm: -1.07130694366
lin dist: 0.0
penalty: -4.0 |||
Line cost: -23.3007429496

als folge all dessen bestehen beim scr derzeit noch mittelbindungen in höhe von mehr als 21 mrd . euro , die ausgegeben werden müssen . |||
a consequence of all this is that the scr today has over eur 21 billion in outstanding commitments awaiting payments . |||
lm: -153.743731
tm: -32.3960292619
rm: -7.32708758403
lin dist: -6.0
penalty: -16.0 |||
Line cost: -215.466847846