Assignment 4
Language & Computers | LINS313

Due: Monday Aug 8, by 11:59pm

The homework is due to Canvas by 11:59pm on its due date. Typeset is preferable, but if you do handwrite it and scan it, please ensure that the writing is clearly legible. If you don't have access to a scanner, there are many free scanner apps for smartphones which will give you a pretty good quality image.

A perfect solution is worth 50 points. **Make sure you show your work**; this will account for the majority of your grade. If any of these instructions or questions do not make sense to you, reach out to me right away.

Problem 1 (4 points) An automatic spelling corrector detects "beeds" as a typo in a document. Generate candidate corrections according to the following edits (two per type of edit):

(a) using one deletion

(b) using one substitution

(c) using one insertion

Also, do the following:

(d) give two candidates that are at least two edits of any kind (and not more than two edits) from beeds.

Problem 2 (15 points) Calculate the minimum edit distance from the string *korekt* to both of the following words:

- *correct*

- *kraken*

Use only insertions, deletions, and substitutions. The costs you should use are as follows:

- insertions: cost 1
- deletions: cost 1
- substitutions
  - no change (e.g. a → a, t→t) : cost 0
  - vowel → vowel: cost 0.5
  - consonant → constant: cost 0.5
  - vowel → consonant: cost 1
  - consonant → vowel: cost 1

For each of the pairs, write down:

- the minimum edit distance, and
- the acyclic graph which you use to calculate the minimum edit distance, with annotated costs for each node.

You do not need to include the string representation for each node in the graph (like "to|nw"), but feel free to include it if it is helpful to you. No need to cram both graphs onto a single page – use a full page for each pair and give yourself plenty of space to write.

Problem 3 (6 points) Consider the following (well-known) text:

would you eat them in a box ? would you eat them with a fox ?
not in a box . not with a fox . not in a house . not with a mouse .

Given this text, and ignoring punctuation, what are the values for the following:

(a) P(not)

(b) P(box)

(c) P(box|a) (the probability of box given that the previous word is a)

(d) P(box|in,a) (the probability of box given that the previous word is a, and that the word two before it is in)

Note: this is a very simple, easy problem, so don't overthink it. The point is that these probabilities really are just coming from counting stuff!

Problem 4 (15 points) You are building a spelling corrector for English and have already built an excellent error detection component. You are given the following sentence:

> Probably no legendary sea monster was as horrifying as the great korekt.

The detector spots "korekt" as a typo. You also have a candidate generator that gives you three candidates as possible corrections: *correct*, *kraken*, and *carrot*. Now you need to rank these candidates by using a Bayesian model trained from text data.

First, you are given a corpus of text which has the following counts:

- 18,933 word tokens
- 126 tokens of correct
- 20 tokens of kraken
- 25 tokens of carrot

Also assume that we know the following from an error model:

- P(korekt | correct) = .025
- P(korekt | kraken) = .008
- P(korekt | carrot) = .034

Your task is the following:

(a) Rank the candidates from most likely to least likely, assuming that we calculate P(candidate=x|typo=korekt) using a unigram language model based on the above training material? Show your work, and the values for each candidate.

(b) The unigram model doesn't use the context of the sentence to choose the best candidate. Let's make it more sensitive to context by using a bigram

language model, which for the above sentence means computing:

$$P(\text{candidate} = x \mid \text{typo} = \textit{korekt},\ \text{previous\_word} = \textit{great})$$

In our training material, we observe the following:

- *great* occurs 1013 times
- *correct* occurs after great 3 times
- *kraken* occurs after great 19 times
- *carrot* occurs after great 2 times

Using the bigram probabilities for each candidate being preceded by *great*, which is the best candidate? Show your work and the values for each candidate.

Note: Use the same error model probabilities for this calculation as you used for (a).

Problem 6 (10 points) Automatic spelling correction can be used in many different contexts, and the balance between precision and recall might be quite different for them. Consider the two following uses:

- A search engine uses automatic spelling correction on web pages before it indexes them so that people searching for terms can find them even when the author of the web page misspells those terms. (So, this doesn't change the web page itself obviously, but it changes the words that the search engine stores in its index as being relevant to that page.)
- A phone doesn't perform interactive real-time spelling correction as the user is creating a text message but can automatically correct the message after it is typed and is being sent. (So, the user cannot correct any errors introduced by the spelling corrector.)

Assume that these spelling correctors can choose not to fix some words that they have detected as typos; that is, they can abstain when they have low confidence in their possible candidate corrections. Discuss whether precision or recall is more important for each case. If precision is more important, discuss which how the kind of error (see the slides) would influence whether the classifier should abstain (not

make a correction) or not.

Note: It will help to first define, for your own benefit, what precision and recall mean with respect to spelling correction. Consider the formula to calculate them. What are false positives in this context? What are true negatives? What are the consequences of different types of results (TP, FP, TN, FN) in these scenarios? You don't have to list everything out, but use these considerations to help justify your answer.