Spell Checker (IR 2020)

Nethra Gunti
S20180010061

Aim

The aim of this challenge is to create a Spell Checker which detects, corrects and suggests more corrections for an error in a word.

Approach

In order to solve create one such spell checker, we will use the Noisy Channel Model.

The dataset we will use for creating the dictionary is Peter Norvig's List of Words (*https://norvig.com/big.txt*).

Method

A spell checker has to be done in the following steps:

1. Error Detetection
2. Error Correction
3. Suggested Words

Each of the steps have been further described in the following slides.

Error Detection

We check if the given word s in the dictionary

If it is, we return the same word as the correct words/

If it is not in the dictionary, we will use kgrams, edit distances of 1 and 2 and edit types delete, insert, substitute and transpose.

This way we will get the list of all the possible kgrams at edit distance=1.

Then, we remove any such words from the list, that are not in the dictionary.

If the list is empty for edit distance 1, we generate candidates for edit distance 2 using the first set of candidates.

Error Correction

Noisy Channel Model

We apply the noisy channel model on the generated list of candidates.

We do this to associate every candidate (c) with a probability of being wrongly spelled as w. This probability is P(c|w)

We apply Bayes Theorem to acquire this probability.

$$P(c|w) = P(w|c) * P(c)$$

Error Correction

Language Model (P(c))

P(c) = probability of a candidate (c) occurring

This probability is the term factor which is calculated by taking the ratio of the number of times *c* occurs and the total number of unique terms in the dataset.

P(c) = Tf / N

Where Tf is the term frequency of c and N is the the total number of terms

Error Correction

Channel Model (P(w|c))

P(w|c)= The likelihood of w being misspelled as (c)

In our model, every generated candidate at an equal distance, has the same likelihood. That is, say there are m candidates generated at a distance of k, then all those m candidates have a likelihood of 1/m.

Since we are having only edit distances 1 and 2, the candidates at edit distance 1 are always prioritized over those at edit distance 2. This is ofcourse, if the word is not already in the dictionary.

Error Correction

Choosing the correct term

As discussed initially, we will assign posterior probabilities to all the candidates of them being wrongly spelled as w.

After calculating these probabilities, we choose the candidate with the highest posterior(P(c|w)). This will give us the best possible correct for a misspelled word.

Suggested Words

In continuation to the previous step, after we calculate the posterior probabilities to each candidate we choose (W) candidates with the highest probabilities.

Out of these W candidates, the first one is the Best Possible Correct word while the remaining words are the Suggested Words.

Thank you!

Nethra Gunti
S20180010061