I. Motivation and Goal:

Spelling mistakes can arise, for example, from incorrect encoding of text which ruins special character (a legacy problem). Another source is OCR (Optical Character Recognition).

Our project’s goal is to improve OCR-Output.

II. Resources:
- SRLM (Stanford Research Institute Language Modeling Toolkit)
- JSBD-1.6 (Sentence Splitter Tool)
- ACL ARC (Anthology Reference Corpus)

III. The Method:

Step 1: Create a substitution list using an unsupervised algorithm

Step 2: Correct OCH-Tools by using the created substitution list

IV. Evaluation:

\[ n = \text{Number of the letters of each word} \]
\[ a = \text{Number of the possible substitutions for a letter} \]
\[ b = \text{Number of the letters to be corrected} \]

Formula for the number of the possible corrected words after substitution:

\[ \frac{n!}{(n-a)!} \times \left( a \right)^{b-a} \times \left( a \right)^{b-a} = \frac{n!}{(n-a)!} \times a^{b-a} \times a^{b-a} \]

For example: for the word “smitten”, \( n = 7 \)

Assume there are 4 substitutions for each letter, then the number of the possible correct words is:

\[ 7 \times 4^{7} = 314,928 \]

In order to generate just one correct word, the computer has to generate 314,928 possibilities between those letters. Further more, since the final result is depend on the value of the possibilities between the words after filter, it is quite possible to generate a word that has nothing to do with the word to be corrected. Therefore, the method is insufficient.

1. The substitution list is too long for one letter, there are 50 candidates for substitution.
2. There is always a Memory Exception.
3. In order to generate just one correct word, there are 50 candidates for substitution. This means that all words that are generated are random. The program will be corrected, but all words that are not correct will be submitted. It is too much unnecessary work.
4. The substitution list is reduced to be only 5 candidates for one letter. The program will run endlessly in case that the length of a word is about 14.

V. Optimization Suggestions:

1. Reduce the substitution list by using a real dictionary when the mostSimilarWords are generated.
2. Define a method to filter out the part of the examples from the article.
3. Not to consider the words that contain more than 10 letters.
4. Not to generate the substitution list, but directly correct the word after the mostSimilarWords are generated.

VI. References:

- http://www.free-program.com/