Modified Chi-Square Spelling Checker

Michael Wotherspoon

February 22, 2017

This program is a Python implementation of a spell checker based on the spell checker detailed in a post on Peter Norvig's blog. It is identical to Norvig's method in how it identifies the list of candidate words, but it differs in how it selects the correct candidate word.

To see Peter Norvig's original blogpost go to:
`http://norvig.com/spell-correct.html`

1 Identification of Candidate Words

Suppose the program is given "appel" as a misspelled word (call this the given word) and that the typer meant to spell "apple" (call this the target word). Our goal is to identify what words (call them candidate words) the typer could have tried to type, so we will perform some transformations on appel. Define the transformations we can conduct as:

- **Insertion:** We insert a letter somewhere into the given word.
  Ex) appel →appels
- **Deletion:** We delete a letter in the given word.
  Ex) appel →appe
- **Transposition:** We transpose two adjacent letters.
  Ex) appel →apple
- **Swap:** We change one of the letters.
  Ex) appel →arpel

By performing different transformations on the given word we transform it into different strings, some of which are words, some of which are not. We perform all possible transformations of the given word by performing exactly one of the above operations and all possible transformations of the given word by performing exactly two of the above operations. For possible single edits to a given word of length n, we have:

- $26(n+1)$ insertions
- n deletions
- $n-1$ transpositions
- $26n$ swaps

We have quite a bit more for possible dual edits. By performing these transformations, we create a list of strings that were created from transforming the given word. We cross-reference this list with a dictionary to eliminate those strings that are not words and this gives us our list of candidate words.

2 Selection of Correct Spelling for Norvig

From our list of candidate words our goal is to identify the target word from that list (assuming it is actually in there). Norvig's method is to select the candidate word with the highest frequency of occurance in the English language. This frequency of occurance is determined by counting frequency of that word in a large set of english text (a million plus words) and dividing it by the total number of words in the set.

3 Drawbacks of Norvig's Method

The main drawback of this method is that it will systematically fail to identify the correct spelling for many uncommon words. Suppose you were trying to spell "nunnery" (one of the least common words in the english language) and you spell it "nunry". Adding an n inbetween "nr" would produce "nunnry"; however, changing the first n to f and the r to n would produce "funny". Funny is a much more common word than "nunnry" so Norvig's method will never identify the correct spelling of "nunry". A method that identifies the correct target word, independent of its frequency in the English language is arguably preferable.

4 New Method to Identify Target Word

We instead treat identification of the target word as a categorical data problem. Consider each letter as a category, and the occurance of each letter in a word as the counts for each category. For each candidate word, we can compare the frequency of letters to the frequency of letters in the given word using Pearson's Chi Square Test. This is a common statistical hypothesis test for categorical

data that test the hypotheses:

$$H_0 : \text{The distributions are the same}$$
$$H_1 : \text{The distributions are different}$$

By computing the p-value for this test associated with each candidate word, we now have a rough measure of how likely it is that the two letter counts come from the same distribution. Continuing under the assumption that this is a reasonable measure of how likely a given word is to map to a target word, we can identify the target word by taking whichever candidate word has the highest p-value (i.e., the weakest evidence to reject H0). However, for cases where two or more candidate words have the same p-value, we revert to Norvig's method and choose whichever one has the highest frequency.

For a more detailed explanation of Pearson's Chi Square Test see:
`http://www.itl.nist.gov/div898/handbook/eda/section3/eda35f.htm`

5 Why the New Method?

It is an undesirable property to have a higher success rate for correcting common words than uncommon words. It is reasonable to assume that people have a higher success rate spelling common words than uncommon. These are words that they spell frequently as well as words that they see frequently. Uncommon words on the other hand, likely have a much lower spelling success rate. A method that systematically fails to correct words that have a higher probability of being misspelled is not very desirable. The method proposed here is theoretically independent of how frequently a word occurs in the English language, which is a useful property. The existence of this property is discussed empirically below.

6 Comparison of Methods

We test Norvig's method against the modified method in this spell checker using a corpora of 1900 commonly misspelled words provided by wikipedia (it can be found at `http://www.dcs.bbk.ac.uk/~ROGER/corpora.html`). Norvig's method has 1524 successes (80.2%) whereas our modified method has 1633 successes (85.9%).

Below are boxplots for both methods showing the distribution of logged word probabilities for the words the method identified correctly and words the method identified incorrectly. We logged word probabilities to mitigate for substantial skewness in the unlogged probabilities. The unlogged boxplots are included as

well for reference. We can see that there is a difference in the logged word probabilities distribution for Norvig's method. On average, correctly identified words have higher probabilities of occurance than incorrectly identified words indicating that this method does fail in cases when the target word is uncommon.

However, the boxplots for our method shows that there is no real difference in the distribution of the logged word probabilities for words this method identified correctly versus words it identified incorrectly. As predicted, there is strong evidence that this method removes the bias against uncommon words.