Typographical error and spelling correction methods analysis

1. Introduction

Writing, as one of the most important skills, is a predominant way for entertainment, communication, and work in people's lives. With the arrival of the information age, more and more people are beginning to type on computing devices instead of hand-writing, consequently, spelling errors are ineluctable due to negligence in typing. Term "Spelling Correction", also known as "Spelling Checker", is widely used in word processing software, input method software, and search engines. The spelling correction process can be split into two parts [1]:

- Spelling Error Detection: According to the type of spelling error, it is divided into Real-word Errors and Non-word Errors. The former means a misspelled word is legal itself but are not intended by the user (beyond our scope), such as "their" is mistakenly spelled as "there"; Non-word Errors are words not in the dictionary, such as "their" is written as "ther".
- Spelling Error Correction: Automatically corrects errors or gives an expected value or even a list of spelling suggestions, for instance, for non-word error "ther", given a list "there, their, other, the" for user.

This article mainly discussed the common causes of typographical errors based on different spelling correction algorithms over some given data and improves corresponding algorithms to enhance performance.

2. Error classification and Methodology

2.1 Error type hypothesis

For the purpose of the assignment, as well as the applied algorithms, typographical errors can be divided into four categories [2]:

- Adding extra letter (e.g., "knowledge" to "knowlledge")
- Deleting correct letter (e.g., "technology" to "technoogy")
- Replacing correct letter (e.g., "assignment" to "assigmment")
- Transporting letters' relative position (e.g. "difficult" to "diffciult")

Among these four types, deletion and substitution are the most common error types in practice (we will discuss later).

2.2 Algorithm overview

There is a variety of spelling correcting algorithms applied in different area. One of the most famous algorithms is Trigram Algorithm [3], the principle of which is that each word extracts a set of three consecutive characters and computes the similarity of two different words by comparing each of their patterns. Another important method is the Bayes approach [4]. The theory is based on inverting the probability of word pairs. In other words, P(c|m) is the probability of the correct word "c" given a misspelled word "m" and P(m|c) is the probability of mistaking the correct word "c" to misspelled word "m". According to Bayes formula,

$$P(c|m) = \frac{P(m|c) \times P(c)}{P(m)}$$

Where P(m) – constant, P(c) is the probability of the intended word "c" which need data training (beyond our scope). In addition, other improved algorithms on the basis of these two algorithms are popular as well, such as Tribayes Algorithm [5]. In this study, three basic spelling correction algorithms are implemented which are Global Edit Distance, N-gram, and Soundex [6], respectively. To compute the global edit distance of two words, I choose Levenshtein approach which the four parameters are 0 (match), +1 (insert), +1 (delete), +1 (replace).

3. Experiments

3.1 Dataset

The dataset manipulated in experiments are "wiki_misspell.txt" [7], "wiki_correct.txt" and "dict.txt". In "wiki_misspell.txt" contains 4453 typographical errors and every error has a correct word in "wiki_correct.txt" corresponding to it. In experiments, these 4453 misspelled words were revised according to "dict.txt" by different methods and then compared the predicted word with the intended word in "wiki_correct.txt".

3.2 Evaluation metrics

The experimental results are evaluated by two major parameters throughout this paper: precision and recall. Precision is defined as follows:

$$\text{Precision} = \frac{\text{correct prediction count}}{\text{total prediction count}}$$

Recall is:

$$\text{Recall} = \frac{\text{correct prediction count}}{\text{totally misspelled word count}}$$

Another secondary factor is running time. Even if an algorithm has satisfying precision and recall, it cannot be applied in practice in case of unacceptable running time.

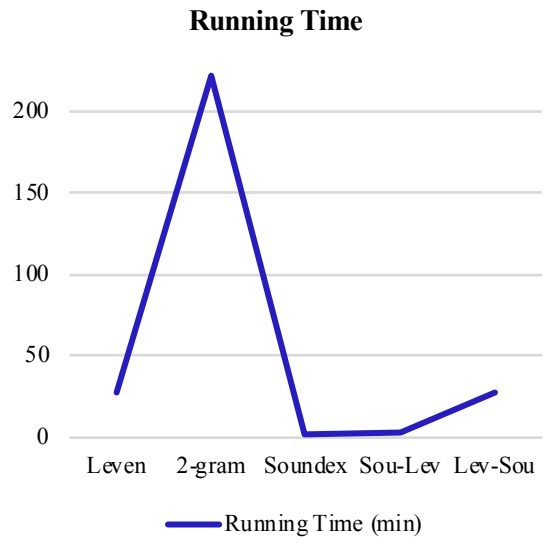
3.3 Implementation

Several experiments are carried out by implementing different algorithms. The first two experiments applied GED (Levenshtein distance) and N-gram (n equals 2) algorithms respectively to compute the distance between every misspelled word in "wiki_misspell.txt" and all correct words in the dictionary (dict.txt), then filter the most similar words for every misspelling as result. In most cases, these two algorithms returned multiple results, i.e., a word list. In the third experiment applied the Soundex algorithm which filters out the closest words with respect to pronunciation.

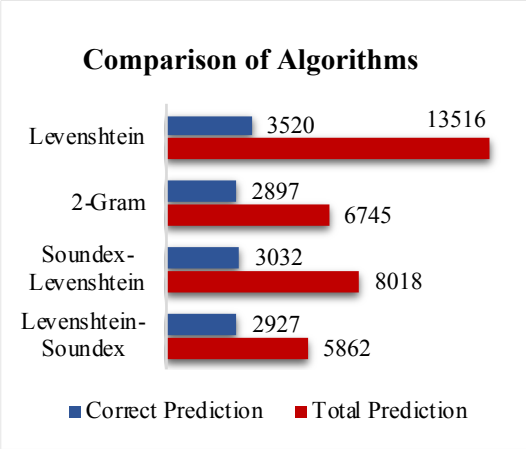
As for the last two experiments, I improved algorithms to some extent combining the advantages of different algorithms. One is "Soundex-Levenshtein" and the other is "Levenshtein-Soundex". That is, first processing the data using one algorithm, afterward, implementing another algorithm on the result.

3.4 Result and analysis

The results are shown below:



Pic 1. Running Time Comparison



Pic 2. Performance comparison

Since the total prediction amount of Soundex algorithm is excessive huge (1201142), it's not shown in the picture.

	Precision	Recall	Time per word (s)
Levenshtein	26.04%	79.05%	0.37
2-Gram	42.95%	65.06%	3
Soundex	0.003%	80.87%	0.027
Soundex-Levenshtein	37.81%	68.09%	0.03
Levenshtein-Soundex	49.93%	65.73%	0.38

Table 1. Results of different algorithms on spelling correction

(My computer configuration is posted in Appendix)

According to table 1, both Levenshtein and Soundex algorithms got high recall (almost the same). Comparing to the 2-Gram algorithm, their relatively high recall means these two algorithms can predict more correct words than 2-Gram. On the contrary, 2-Gram's precision is much higher than the other two algorithms and this may be due to the small amount of predictions. Considering running time, Levenshtein and Soundex algorithms implemented much faster than 2-Gram (3 seconds per word is unacceptable), therefore, 2-Gram was eliminated.

As you can see clearly, the precision of Soundex is incredibly low! Hence, it's necessary to develop an algorithm which maintains high efficiency, recall and increases precision at the same time. In order to achieve this goal, two updated algorithms came into picture by implementing Levenshtein and Soundex step by step. According to the experimental result, although the recall of Levenshtein-Soundex is slightly lower than the Soundex-Levenshtein algorithm, the precision is much higher than Soundex-Levenshtein. Both of them has very fast speed so the running time is not the vital consideration. From the above analysis, the improved algorithm Levenshtein-Soundex has the best performance on non-word error correction.

4. Error type analysis

The statistics show that about 80 percent of typographical errors are caused by letter addition, deletion, substitution, and transposition [8] and our experimental results happened to justify the hypothesis. From the above recall, we found that all values are between 65% - 80%. Furthermore, both 2-Gram and Levenshtein algorithms are good at solving letter addition, deletion, and substitution errors. No matter adding an extra letter, deleting or replacing a correct letter, the Levenshtein and 2-Gram distances should be very close to zero. For example, "knowldge" and "knowledge":

- The Levenshtein distance is 1
- The 2-Gram distance is $19 - 2 \times 8 = 3$ (the smallest value in this pattern)

```
['hydropobe', 'hydrophobe', True, 'hydrophobe']
['hydropobic', 'hydrophobic', 'hydroponic', True,
'hydrophobic']
['hygeine', 'hygrine', False, 'hygiene']
['hyjack', 'hijack', True, 'hijack']
['hyjacking', 'hijacking', True, 'hijacking']
['hypocracy', 'gynocracy', 'hypocrisy', 'popocracy',
'shopocracy', True, 'hypocrisy']
['hypocrasy', 'hypocrisy', True, 'hypocrisy']
['hypocricy', 'hypocrisy', True, 'hypocrisy']
['hypocrit', 'hypocrite', 'hyporit', True, 'hypocrite']
['hypocrits', 'hypocrite', 'hypocrites', True, 'hypocrites']
['iconclastic', 'iconoclastic', True, 'iconoclastic']
['idaeidae', 'cidaridae', 'dicaeidae', 'idoteidae',
'odacidae', False, 'idea']
```

Pic 3. Random screenshot of the result using Levenshtein algorithm (Appendix)

Picture 3 is a random screenshot from the result by implementing Levenshtein algorithm. As we can see in picture 1, the first item is the misspelled word from "wiki_misspell.txt" and the subsequent words are predicting words by Levenshtein algorithm until the Boolean value "True" or "False". The last word in "wiki_correct.txt" is the correct word which Wiki editor truly intended. There are 10 correctly predicted words and 2 mispredicted words. In the case of correct prediction, 4 of 10 are deletion errors and 6 of 10 are substitution errors. For those 2 incorrect predictions, one is transposition error and the other is a weird error.

```
['interm', 'inerm', 'inter', 'interim', 'interj', 'intern',
'inters', True, 'interim']
['internation', 'internation', False, 'international']
['interpet', 'intermet', 'internet', 'interpel', 'interpret',
'interset', True, 'interpret']
['interrim', 'interim', True, 'interim']
['interrugum', 'interregnum', True, 'interregnum']
['intertaining', 'entertaining', 'intertwining', True,
'entertaining']
['interupt', 'interrupt', True, 'interrupt']
['intervines', 'interlines', 'intervenes', True,
'intervenes']
['intevene', 'intervene', True, 'intervene']
['intial', 'inial', 'initial', 'intil', 'intill', 'intimal',
True, 'initial']
```

Pic 4. Another random screenshot of the result using Levenshtein algorithm (Appendix)

In this case (Pic. 4), there are 9 correctly predicted words and 1 mispredicted word. 6 of 9 are deletion errors, 2 of 9 are substitution errors and 1 of 9 is addition error. As for the incorrect prediction, it's a real-word error which is beyond our scope (machine learning).

The experimental result indicates that these four types of typographical errors account for a large percentage indeed, in which deletion and substitution are particularly common. There are also a variety of different typographical error types with respect to the remaining 20 percent errors, such as real-word error.

```
['miliary', 'miliary', False, 'military']
['miligram', 'milligram', True, 'milligram']
['milion', 'ilion', 'million', 'milton', 'minion', True, 'million']
['miliraty', 'miliary', 'military', 'militate', True, 'military']
['millenia', 'millenia', False, 'millennia']
['millenial', 'millenia', 'millennial', True, 'millennial']
['millenialism', 'millennialism', True, 'millennialism']
['millenium', 'millenium', False, 'millennium']
['millepede', 'millepede', False, 'millipede']
```

Pic 5. Random screenshot of the result in case of misprediction (Appendix)

As we can see clearly in picture 5, all these four mispredictions are real-word errors. Without a large number of statistics, it's impossible for the computer to predict the intended words successfully.

5. Conclusion

In this study proposed hypothesis of common types of typographical errors based on given data set and experimental results presented also preliminary verified the catholicity of four types spelling error which are addition, deletion, substitution, and transportation. In addition, we discussed the performance of different spelling correction algorithms and improved the algorithm by combining Levenshtein and Soundex algorithms. Although the improved algorithm reduced recall, precision is greatly increased as a trade-off.

(1454 words)

References

[1] P. Samanta, B. B. Chaudhuri, "A simple real-word error detection and correction using local word bigram and trigram," in *Proc. the Twenty-Fifth Conference on Computational Linguistics and Speech Processing*, 2013, pp. 221-220.

[2] E. Mays, F. J. Damerau, R. L. Mercer, "Context-based spelling correction," in *Proc. IBM Natural Language ITL, Paris, France*, 1990, pp. 517-522.

[3] E. M. Zamora, J. J. Pollock, A. Zamorat, "The use of trigram analysis for spelling error detection," *Information Processing & Management*, Vol. 17, 1981, pp. 305-316.

[4] S. Sumit, S. Gupta, "A Correction Model for Real-word Errors," *Procedia Computer Science*, 2015, pp. 99–106.

[5] Y. Zhou, S. Jing, G. Huang, S. Liu, Y. Zhang, "A correcting model based on Tribayes for real-word errors in English essays," In Computational Intelligence and Design (ISCID), Vol. 1, 2012, pp.407-410

[6] Zobel, Justin and P. Dart, "Phonetic String Matching: Lessons from Information Retrieval," in *Proceedings of the Eighteenth International ACM SIGIR Conference on Research and Development in Information Retrieval*, Zu¨rich, Switzerland, 1996, pp. 166–173.

[7] Wikipedia contributors (n.d.) Wikipedia: Lists of common misspellings. In Wikipedia: The Free Encyclopedia, http://en.wikipedia.org/w/index.php?title=Wikipedia:Lists_of_common_misspellings&oldid=813410985

[8] J. L. Peterson, "A note on undetected typing errors," *Communications of the ACM 29(7)*, 1986, pp. 633-637.

Appendix

Computer configuration:

- Computer model: MacBook Pro (15-inch, 2017)
- Processor: 2.8 GHz Intel Core i7
- Memory: 16GB of 2200MHz DDR3 onboard memory
- Graphics: Radeon Pro 555 with 2GB of GDDR5 memory and automatic graphics switching Intel UHD Graphics 630
- Storage: 256GB SSD

Content of picture 3:
['hydropobe', 'hydrophobe', True, 'hydrophobe']
['hydropobic', 'hydrophobic', 'hydroponic', True, 'hydrophobic']
['hygeine', 'hygrine', False, 'hygiene']
['hyjack', 'hijack', True, 'hijack']
['hyjacking', 'hijacking', True, 'hijacking']
['hypocracy', 'gynocracy', 'hypocrisy', 'popocracy', 'shopocracy', True, 'hypocrisy']
['hypocrasy', 'hypocrisy', True, 'hypocrisy']
['hypocricy', 'hypocrisy', True, 'hypocrisy']
['hypocrit', 'hypocrite', 'hyporit', True, 'hypocrite']
['hypocrits', 'hypocrite', 'hypocrites', True, 'hypocrites']
['iconclastic', 'iconoclastic', True, 'iconoclastic']
['idaeidae', 'cidaridae', 'dicaeidae', 'idoteidae', 'odacidae', False, 'idea']

Content of picture 4:
['interm', 'inerm', 'inter', 'interim', 'interj', 'intern', 'inters', True, 'interim']
['internation', 'internation', False, 'international']
['interpet', 'intermet', 'internet', 'interpel', 'interpret', 'interset', True, 'interpret']
['interrim', 'interim', True, 'interim']
['interrugum', 'interregnum', True, 'interregnum']
['intertaining', 'entertaining', 'intertwining', True, 'entertaining']
['interupt', 'interrupt', True, 'interrupt']
['intervines', 'interlines', 'intervenes', True, 'intervenes']
['intevene', 'intervene', True, 'intervene']
['intial', 'inial', 'initial', 'intil', 'intill', 'intimal', True, 'initial']

Content of picture 5:
['miliary', 'miliary', False, 'military']
['miligram', 'milligram', True, 'milligram']
['milion', 'ilion', 'million', 'milton', 'minion', True, 'million']
['miliraty', 'miliary', 'military', 'militate', True, 'military']
['millenia', 'millenia', False, 'millennia']
['millenial', 'millenia', 'millennial', True, 'millennial']
['millenialism', 'millennialism', True, 'millennialism']

['millenium', 'millenium', False, 'millennium']
['millepede', 'millepede', False, 'millipede']