Project 1: Very tweetz. Such non-canonickal. Amayze

1. Introduction

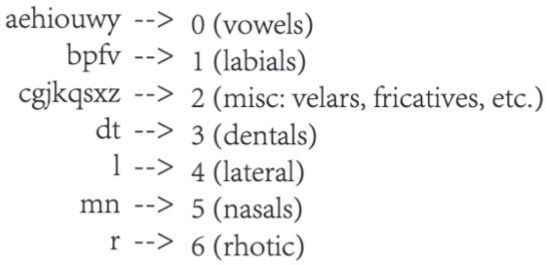
The missspelling in the system become crucial since the computer input still needs human intervention. That may affect the difficulties in understanding the word meaning (James, 1980). As a result, spelling correction is used to prevent this problem. The purposes are to develop the system that should be working on the spelling correction checkers which give recommendation words for every misspelling word and comparing the recommendation words with the correct words for metric measurement. Furthermore, this paper explains the spelling correction with the methods include edit distance and n-grams, and evaluate those methods using precision and recall metric.

2. Explanation and Methods

Spelling correction usually are divided into three types, which are typographic errors (e.g., the → teh), cognitive errors (e.g., receive → receive), and phonetic errors (e.g., abyss → abiss). (Kukich, 1992). The several popular methods are *n-gram, edit distance, soundex, probabilistic, rule-based* and *neuralnets*. Both of those methods have different benefit and drawback depends on the processed data. In this paper, we will discuss two methods which are edit distance and Soundex

Edit distance - Edit distance method commonly called the Levenshtein distance method because this method firstly presented by Levenshtein (1965). The methods calculate and measure two different string which is source string and target string. The measurement from source string to target string will obtain the distance result that indicates the number of deletions (d), insertion (i), or substitutions. The substitution parameter usually divided into two which are matching (m) or replacement (r). The simple edit methods will count the minimal number of insertions, deletion and substitutions. The standard parameter for each measurement is:

$$m = 0; I = 1; d = 1; r = 1$$



aehiouwy	-->	0 (vowels)
bpfv	-->	1 (labials)
cgjkqsxz	-->	2 (misc: velars, fricatives, etc.)
dt	-->	3 (dentals)
l	-->	4 (lateral)
mn	-->	5 (nasals)
r	-->	6 (rhotic)

Figure 1 the letter translation in soundex

Soundex – Zobel and Dart convey that the Soundex developed by Odell and Russel, and patented in 1918 [Hall and Dowling, 1980]. Soundex is the method who use the similarity of sound's letter. Soundex will change every letter in the word into code, but the maximum is four code. The first character indicates the first letter and the others show the letter translation. Figure 1 shows the letter translation.

The Soundex has four steps to change the word into code. Firstly, translate letters based on Figure 1, except for the first letter. Secondly, replace the adjacent similar code with one code. Then, eliminate all 0 (vowels) code. Lastly, keep the first four character.

Method	Misspelled	Correct	Matched set
Edit Distance	mesage	message	mesange, pesage, metage, menage, message, mesnage
Soundex	mesage	message	Meekest, meekoceras, mucocele, mucocellulose, mucocellulosic, myxogasteres, mesosalpinx, mesosaur,…

Figure 2 output demonstration

3. Dataset

For this experiment, we use data 3 data which are the correct.txt, misspell.txt, and dictionary.txt. The English dictionary data is retrieved by Infochimps. Also, the correct and misspell data are provided by Han and Baldwin (2011). The data represent common words that commonly happen in the real world. The data format is the string by line. Both the data have a similar sequence or word order so that it may have a comparison for evaluation purpose. Table 1 shows the statistics of our dataset.

Item	Number of words
Dictionary	370099
Misspelling	10322
Correct	10322

Table 1 Dataset Statistic

4. Result and Evaluation

In this part, we employ the result as similar to the real because we have aims to demonstrate and compare the characteristic of many methods. To understand the performance, we measure the accuracy and precision metric for measurement. The accuracy metric indicates several correct token predictions, and it divided with the total number of accurate predictions. Then, the precision means the number of tokens correctly predicted, and it shared with the total number of predictions whether it correct or wrong. (Junker and Hoch, 1999). Furthermore, we also use time measurement for the metrics result since the big data and plausible reasons which are the time effectiveness for running the program. As additional information, we use MacBook Pro (Quad core, 8 GB RAM) to develop and run the programs.

Table 2 shows the demonstration result. Compared with edit distance, the Soundex has a considerable matched set number rather than edit distance. The reason is the Soundex predicted the first letter and the rest of the letter would be changed into code that indicated voice similarity. For instance, if we input the letter translation into “message”, and the original code is “m022020”. After the processing, the final result will be “m22”. With that code, it will have much similarity, such as meekest, and meekoceras.

Metric	Accuracy	Precision	Time (Seconds)
Edit distance	77.37%	19.61%	9175
Soundex	79.02%	0.019%	1310

Figure 3 metrics result

However, edit distance only measures the minimum parameter (deletion, insertion, and substitution) between two strings. As a result, the edit distance has a higher percentage of precision. (Table 3)

As we can see in Table 3, Soundex has higher percentage in accuracy and faster in time measurement. The accuracy differences are having the short gap, while the time has the high gap.

Conclusion

We had discussed two methods for approximate string matching and evaluated with accuracy, precision, and running time. We found that edit distance has more precision with the longer processes time. Besides, we also illustrate the method characteristic and compared it.

References

Baldwin, Timothy, Marie Catherine de Marneffe, Bo Han, Young-Bum Kim, Alan Ritter, and Wei Xu (2015) Shared Tasks of the 2015 Workshop on Noisy User-generated Text: Twitter Lexical Normalization and Named Entity Recognition. In Proceedings of the ACL 2015 Workshop on Noisy User-generated Text, Beijing, China, pp. 126–135.

Infochimps, (2013) retrieved in https://web.archive.org/web/20131118073324/

James L. Peterson. 1980. Computer programs for detecting and correcting spelling errors. Commun. ACM 23, 12 (December 1980), 676-687. DOI: https://doi-org.ezp.lib.unimelb.edu.au/10.1145/359038.359041

Junker, M., Hoch, R. and Dengel, A., 1999, September. On the evaluation of document analysis components by recall, precision, and accuracy. In Proceedings of the Fifth International Conference on Document Analysis and Recognition. ICDAR'99 (Cat. No. PR00318) (pp. 713-716). IEEE.

Kukich, M., 1992. Techniques for automatically correcting words in text. ACM Comput. Surv. 24, 4 (December 1992), 377-439. DOI: https://doi-org.ezp.lib.unimelb.edu.au/10.1145/146370.146380

Levenshtein, V.I., 1966, February. Binary codes capable of correcting deletions, insertions, and reversals. In Soviet physics doklady (Vol. 10, No. 8, pp. 707-710).

Zobel, J. and Dart, P., 1996, August. Phonetic string matching: Lessons from information retrieval. In Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 166-172). ACM