1. The author used three different approximate strings matching methods implementing this blend word detecting program. Based on these results, the author gave some analyses to each method respectively. And the author also gave some improvements about the problems.

In the first method, the author divided the candidate words in to two parts and used global edit distance to calculate the scores between the words in the dictionary and these two parts, if the first part and the second part of the candidate word can find different words in the dictionary that meet the distance requirement. Then the candidate words returned as a blend word.

In the second method, the author used local edit distance to find the best matching sub-strings between candidate word and words in the dictionary. If there are the best matching substrings located in the beginning of the word and located in the end of the word respectively. The word will be returned as a blend word.

In the third method, Jaro-Winkler are used to calculate similarity between the candidate word and words in the dictionary, the author choose the first word in the dictionary which has the similarity over 0.9 to the candidate word, then author remove the similar prefix of the candidate word and reverse the remaining part and calculate the Jaro-Winkler Similarity with the reversed dictionary. If existing a word in the dictionary, the candidate word will be returned as a blend word.

1. The structure is very clear, and the analysis is very detailed. For example, the author gave some instances in each analysis. This will make the reader easier to understand the problem. The third method is novel. Because the Jaro-Winkler is perfect at matching words that have similar prefix, the author reversed the remaining part of the word and every words in the dictionary before matching possible word that compose the suffix of the candidate word. Besides, the author also gave some further improvement ideas about the problems generated in the analysis phase.
2. The author can do more pre-processes in the dictionary and candidates, this will reduce the CPU computation time and it will also improve the performance of the system.

The author also can write more detailed information in this paper about the parameters, for example, when the distance of two strings is smaller than which number, the string should be considered as part of component. And the author can use the data to support the case. Besides, the author can display result that is the number of the returned predicted blend word by the system.

1. The author analyzed 5 cases of blend words, the characteristics of each blend word are given respectively. Then the author use 5 methods to detect the blend word.
2. The first method is to find the prefix of the candidate word and analysis rest part of the candidate word with the dictionary word. And giving different weights to different part of matching. Through calculating the value of each word, then return the best one.
3. The second method is preprocess the dataset by removing some nonsensical words in the candidate and using another dictionary, then the author use multi-processing to do the same algorithm with the new datasets;
4. The third method is using local edit distance to matching the second component with the candidate word;
5. The fourth method is that the author added a letter conversion algorithm, this algorithm can transform letters to other similar-sounding letters, the remaining processes are the same;
6. The last method is that the author tokenized those substrings that cannot be separated before doing the remaining things, and let those substrings not apart, the remaining processes are the same.
7. 2.1. Introduction is very good, which can attract readers and clearly clarify the value of this research.

2. Each method is very detailed and novel, every steps the author give his/her idea and why he could do like this, this make the reader more easy to understand these methods.

3. The author used multithreading program to handle the dataset and the data set is divided, because the dataset is too large. The author is good at applying appropriate methods to practical problems. That inspired me.

3. 1. The structure of this paper is not so good, perhaps the author can use the table to instead the screen capture, I think it’s better.

2. It is better to have a introduction chapter of dataset, otherwise the readers will be confused about the quality and representativeness of the data set. Because the quality of the data set can impact the outcome.

3. It is better to have an improvement guesses for the problems and [deficiency](javascript:;) that given by the experiments, even if the idea is difficult to implement (it is not necessary to implement), but it can give readers some thinking and inspiration.

The author preprocesses the candidate dataset, removing some words that exist continuous 3 same letters or repeating continuous substrings. Then the author creates dictionaries for each candidates to record the maximum matching prefix and suffix with the words dictionary. The author use global edit distance (Levenshtein edit distance) to calculate the every possible combination, choosing the pair of prefix and suffix that has the smallest distance. If the combination of the prefix and suffix is short than this candidate, this candidate is the blend word. The author then use the Jaro-winkler to find the best prefix and the 2-gram to choose the best suffix to improve the system, instead of using edit distance. In the end the author give the results and analyses.

1. The structure of this paper is clear and pretty good, because the author clearly expresses what he wants to study, how he deals with the process, and his analysis of the process and result.
2. The methods of this paper is easy to understand, because the author try to explain the process of his experience very detailed.
3. It is a good idea to create a dictionary for each candidate word and choose the value list of the maximum key in those dictionaries. It’s a very novel idea and it can reduce a lot of unnecessary computation.

1. It is better for this paper to add improvement strategies. That is the author could give his/her guesses to deal the problem he/she find. For example, he/she talked about the questions in 6.3 chapter, the author can give some guesses to solve these problems, even if these guesses are not correct, it can inspire readers to explore these problems.

2. In the analysis table, It is better for author to give the number of denominators, Otherwise it's hard to explain where the recall came from.

3. In 4.2 chapter, the reason of ignore those words that the maximum matching prefix or suffix should use data to support the idea, for example, what is the percentage of those kind of words.