New labeled dataset of interconnected lexical typos for automatic correction in the bug reports

Behzad Soleimani Neysiani¹ · Seyed Morteza Babamir¹

© Springer Nature Switzerland AG 2019

Abstract

Large-scale and especially open-source projects use software triage systems like Bugzilla to manage their user's requests like bugs, suggestions, and requirements. The software triage systems have many tasks like prioritizing, finding duplicate and assigning bug reports to developers automatically, which needs text mining, information retrieval, and natural language processing techniques. We already showed there are many typos in the bug reports which reduce the performance of artificial intelligence techniques. The connected terms were one of the most types of typos in the context of bug reports. Also, we introduce some algorithms to correct the connected terms earlier, but there was not any labeled dataset that can be used to evaluate the accuracy of process of typo correction. Now we made a new labeled dataset including 42,970 typos between 182,096 to can be used for the typo correction evaluation process. There are 52% connected typos in the labeled dataset, which show the previous results about the number of connected typos were correct. Then we used the typo correction algorithms which were introduced in prior studies to evaluate their accuracy. The experimental results show 81.6% and 83.3% accuracy in top-5 and top-10 suggestions of the list of typo corrections, respectively.

Keywords Natural language processing · Typo correction · Interconnected lexical typo · Tree structure · Bug reports

Mathematics Subject Classification 68T50 · 68T20 · 68U15 · 68P05 · 68P10 · 68P20 · 68P30 · 94A13 · 68Q25 · 68R15 · 68W10 · 68W32 · 68W40 · 05C05

JEL Classification C88 · L86 · D81 · D83 · L17 · Z13

1 Introduction

Software triage systems such as Bugzilla are software which usually gets bug reports online, and then the Triagers will deal with these bug reports to evaluate the importance and priority of each report, finding duplicate reports based on their contents, assign bug reports to developers for checking bugs and planning to modify the project in future. Because of the large count and volume of bug reports, many researchers have tried to automate these processes since 2004 by artificial intelligence techniques and algorithms. Duplicate bug reports detection is a great

problem in this research area [1, 2]. The algorithms and techniques of duplicate bug report detection such as Term Frequency and Inverse Document Frequency in information retrieval techniques need to check the similarity of two bug reports to each other word by word [3], so the lexical correctness of words and terms is essential for these techniques [4]. There are many typos in bug reports, e.g., more than 50% of bug reports have typos, and more than 2.5% of bug reports have more than 50% typos [4]. These typos distort the similarity detection process in duplicate detection. It is crucial to detect and correct these typos automatically because there are more than 1.5 million

✉ Behzad Soleimani Neysiani, B.Soleimani@grad.kashanu.ac.ir; Seyed Morteza Babamir, Babamir@kashanu.ac.ir | ¹University of Kashan, Kashan, Esfahan, Iran.

Fig. 1 The four steps of finding meaningful words in an inter-connected term [7]

typos [4] in Mozilla Firefox, Android, Open Office, and Eclipse datasets [5] and about 390-kilo unique typos in there. A scientific semi-dictionary is made for typo detection in bug reports to detect typos automatically [4].

There are many types of typos in texts like additional, removal, or substitute characters. Interconnected terms are a regular typo in a software context because there are many method or class names in this context which contains interconnected terms like 'LinkedList' or 'connect-ToServer'. Sometimes these words are camel case, and sometimes users typed them and have not any specific case sensitivity. Also, sometimes, users forgot to press space between words, so there will be many interconnected terms in the software bug reports. These interconnected terms must be separated; otherwise information retrieval techniques like term frequency, cannot detect text similarities for the duplicate bug report detection problem. Some new algorithms are introduced in prior studies [6, 7] for the correction of interconnected terms, but there is not any standard labeled dataset to evaluate the accuracy of them. The primary purpose of this research is to make a labeled dataset and evaluate the accuracy of the algorithms of the correction of interconnected terms.

2 Literature review

Typo detection and correction is a regular and an ancient issue in text mining and natural language processing [8, 9]. There are many efforts on typo detection and correction in a scientific context like clinical records, which uses Shannon's noisy channel model to predict the next words based on the previous word sequence [10]. In some cases, there is less last word sequence like web query, so the log of web query can be used as a baseline, and the maximum entropy model can help for rare queries to conquer the sparseness problem of prior data [11].

Some researchers focus on the correction of misspelled typos by different kinds of machine learning and natural language processing models, e.g., creating a confusion matrix for the other type of misspellings like additional or removal or transposal or replaced characters to searching these patterns in terms and predict the correction [12]. String transduction tries to map one string to another and can be used for misspelled typo corrections [13]. Machine learning is used in character scale to typo detection and corrections, but the recall rate is low (about 30%) [14].

Also, phonetic, language and keyboard models can be useful for correction prediction by decision tree as a machine learning-based technique [15, 16]. Another approach can be creating a model based on machine learning techniques to detect typos and predict the correction according to context and domain knowledge [17, 18].

Some other researchers focus on using tree structure for typo correction. It is possible to make a tree based on a probabilistic model of the relationship between characters of words which what characters can become after a particular character and in advance mode, after a sequence of characters. So, these models use Bayes's theory to make a prediction model on a tree called Trie and use it for typo correction as the user is typing [19, 20]. The tree structure can be used for grammatical checking and translating, too, by merging several grammatical trees in a Trie [21]. The simple Trie (without probability) is used for spell checking also [22]. The acyclic deterministic finite automata is a graph with a similar structure that can be used for spell checking and typo correction [23]. There are some methods for query in Trie by wild characters, too [24]. Trie-based index structure can be used for real-time interaction like search recommendation and query completion [25].

The interconnected terms problem was not significant a lot in other contexts, and there is no specific method for the correction of interconnected terms. As it was tested, the google translate, and Microsoft office word can detect two parts interconnected terms and recommend a correction for them, but if there are more than two meaningful terms, they cannot identify and suggest any correction. It shows that even huge companies have not been investigated with this problem until now in general-purpose situations.

A divide and conquer algorithm based on the longest common sequence algorithm can be considered to find out the meaningful terms in an interconnected term. It is a simple brute force algorithm which will consider all combinations of start and end index of a substring in an interconnected term to find a meaningful term. Meaningfully checking needs a dictionary. A good trustful dictionary for computer context has been made which can be used for this purpose too [4].

The procedure of correcting an interconnected typo is shown in Fig. 1, which has four steps [7]. In the first step, a Trie called Neural Match Tree [7] is created based on a trustful dictionary to find meaningful words in an interconnected term (I.T.). There are many possibilities for

Fig. 2 The methodology of evaluation of predicting algorithms of interconnected typos

meaningful words in an I.T., which can have overlap too. Consider ‘hellohelphissbookhishel’ as an I.T., so ‘he’ and ‘help’ or ‘his’ and ‘hiss’ are two meaningful words with overlap. These meaningful words should be found in the second step, which there are two algorithms until now for this purpose [6, 7]. Then in the third step, the meaningful terms should be arranged side by side to make the I.T. without any lack or additional characters in case of having no other type of typos in the I.T. Then it is possible to have different combinations of meaningful terms like ‘abstract hosts hell’ and ‘abstract host shell’ in ‘abstracthostshell’ which both combinations are meaningful and the best one should be chosen based on the main context.

3 Methodology

The primary purpose of this study is to evaluate the accuracy of algorithms used for the correction of interconnected typos. The prior dataset [4] has no label; in other words, the rectification of each typo was not given in the dataset. So, in this meanwhile, we select some typos randomly and divide them into 1000 items in separate files and ask some students of computer engineering to determine the correction of each typo manually. It was a time-consuming process. Then all the revisions have gathered and combined. There are 42,970 typos in this labeled dataset now.

A new procedure is used to determine the correction of interconnected typo between all combinations, as shown in Fig. 2. In this procedure, a typo is given to the predictor algorithms, as shown in Fig. 1, and the combination list will be made. The corrector algorithms are a kind of predictor algorithm because they have to predict the nearest combinations for end-users. Then the index of the correct form of typo will be searched in the list of combinations. If the correct form were not in the list, the result would be none (e.g. – 1); otherwise, it will be a positive number. The editor software like Microsoft word usually show a list of suggestion to end-user, which can select the best form between them. This behavior is like the operation of recommender systems, which show top-k suggestions to customers. If the answer is in the first suggestion, it is ideal, and it can be acceptable if the answer is in the top-5 or top-10 list; otherwise, the answer is not proper.

4 Experimental results

The new scientific semi dictionary as a word list and unique typo dataset of bug reports picked for evaluating the proposed procedure [4]. The implementation of all algorithms and procedures done in Python 3.7 programming language. In the first step, the new labeled dataset analyzed, and it was denoted that it has 42,970 typos. There is a 16,932 interconnected typo that has no other kind of typos like removal or additional characters. So, the selected part of the new labeled dataset has been chosen for the evaluation process as the primary dataset of this study.¹ The results of the evaluation process are shown in Table 1. The first row indicates the number of space (blank) in the correct form of the I.T., and the first column shows the index of correct form in the list of suggested combinations. The – 1 value in the index column indicates the correct form is not found in the list.

There are 13,820 true predictions between 16,932 I.T. in the top-5 suggestions and 14,120 in the top-10 recommendations. So, the accuracy of predictor algorithms is 81.62% and 83.39% for top-5 and top-10 lists, respectively. Interestingly, the accuracy of the top-1 suggestion is 66.69%, which is considerable.

The last row of Table 1 shows that 9004 typos have one space in the correct form, in other words, they have two meaningful words, and other ones have more than two connected words, which are about 46.8% of interconnected terms. So, it shows that the efforts of this study were not pointless, and it is essential to proceed with this study more, especially for software triage systems and similar systems like FAQ forums, e.g., Stackoverflow.

The second row of Table 1, which shows the not found correct form, indicates that many typos can not be solved by proposed algorithms of prior studies [6, 7]. The reason for this phenomenon has checked, and these challenges have found which should be considered for future works:

1. Sometimes there is some mistake in the dataset which human correctors don’t lend them. The good results show that these mistakes are few but are not zero. For

¹ <https://github.com/SabaRG/TypoCorrectionDatasets/tree/master/SNAS>.

Table 1 The evaluation results of correction of interconnected typos

Space→ Index↓	1	2	3	4	5	6	7	8	9	Total
-1	703	981	615	243	96	38	8	3		2,687
1	6,798	3,319	933	207	32	4				11,293
2	177	84	18	2						281
3	1,034	490	197	46	15	2	1			1,785
4	42	19	9	2						72
5	192	130	49	11	6	1				389
6	14	7	2							23
7	34	71	42	11	1	1				160
8	6	5	3	1						15
9	2	49	29	9	1					90
10		6	3	3						12
11	1	26	20	6	1					54
12		2								2
13	1	14	8	4	1			1		29
14		2			1					3
15		6	7	4	1					18
17		5	3	3	1				1	13
19		1	4	1						6
Total	9,004	5,217	1,942	553	156	46	9	4	1	16,932

example, there are some incomprehensible terms in the dataset like 'wszelkie' which should not be classified as an interconnected term, but the human corrector selected this wrong. There are many other examples that the predictor algorithms suggest accurate prediction, but the human correctors wrongly suggested.

- Sometimes the human correctors of the labeled dataset choose a prefix or postfix with the term as a single term like the plural 's' or 'un' prefix in 'students' and 'unregister.' The primary reference of predictor algorithms is the scientific dictionary, which may have not 'unregister' as a term. So, after predicting, the stem of both correct form and predicted combination should be checked too, and sometimes it is impossible, e.g. 'isenabling', which misleads the predictor because the term 'enabling' is not in the dictionary.
- There are many abbreviations or similar words like file extensions in the selected scientific dictionary, which cause misdirect predictor algorithms. For example, consider 'xredline' as I.T. The algorithms predict 'xre dli ne' as the first combination because the 'xre' and 'dli' are meaningful terms in the selected dictionary as an abbreviation.
- Sometimes there is some new idiom that is not in the scientific dictionary and leads the predictor algorithms wrong. For example, the term 'slideshow' was not in the dictionary and predictor select 'slide show' as a

result, which was not equal to the human corrector selection.

- The average word length (AWL) is not a useful metric in all cases. For example, the AWL of 'x red line' and 'xre dli ne' are the same and equal to 8/3, but the length of each word of every combination is not identical. Sometimes a combination with the most word length is more acceptable. It is better to introduce new metrics to cover these situations too.
- The prior predictor algorithms [6, 7] are heuristic and fast but do not check the total search space. They try to find the best combination by choosing the first term which has the most AWL for the beginning position of combination, then second term and so on from left to right. There are some cases that need choosing the first term in the middle or last of combination with the most AWL. For example, the predictor checks the 'mypassword' and return 'myp ass word', which is not correct, and if the predictor selects the longest meaningful term ('password') first and then chooses the 'my', the AWL was more than the selected combination.

The AWL of corrected terms based on the index of top-k suggestion is shown in Fig. 3. It seems the higher indexes have a lesser maximum AWL, but other metrics like average and minimum of different groups are similar. It shows

Fig. 3 Minimum, maximum and average of average word length of selected combinations based on the index of the selected combination by the predictor

that for lesser AWL, there is a problem in the predictor that should be improved to achieving more accuracy.

5 Conclusion

This study introduces a new labeled dataset for interconnected typo (I.T.) correction and prior supplement studies. The new dataset is used to evaluate the accuracy of previous studies. The experimental results show that more than 46% of interconnected terms have more than two meaningful words. Also, the accuracy of the correction of I.T.s was more than 81%, which is not so bad, but it can be improved in the future. It should be considered that the runtime of these algorithms is very much low (less than 1 s), and their memory usage is very low, too (in the order of the size of the dictionary) [6, 7]. So, it is good to use the Neural Match Tree-based algorithm for prediction and correction of interconnected terms.

In the future, many improvements can be used in the meaningfulness combination extraction process to achieve the best one between other combinations and also based on the main context. Also, other metrics can be introduced for this purpose instead of average word length, which was used in state of the art. The meaningfulness combination finding algorithm can be improved too.

Acknowledgements It is my duty to thanks my dear students from the University of Kashan, Islamic Azad University of Isfahan (Khorasan), and Shahid Ashrafi Esfahani University, which help us to make the new labeled dataset.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Soleimani Neysiani B, Babamir SM (2019) Improving performance of automatic duplicate bug reports detection using longest common sequence. In: IEEE 5th international conference on knowledge-based engineering and innovation (KBEI), Tehran, Iran
2. Soleimani Neysiani B, Babamir SM (2019) New methodology of contextual features usage in duplicate bug reports detection. In: IEEE 5th international conference on web research (ICWR), Tehran, Iran
3. Soleimani Neysiani B, Babamir SM (2016) Methods of feature extraction for detecting the duplicate bug reports in software triage systems. Paper presented at the international conference on information technology, communications and telecommunications (IRICT), Tehran, Iran
4. Soleimani Neysiani B, Babamir SM (2018) Automatic typos detection in bug reports. Paper presented at the IEEE 12th international conference application of information and communication technologies, Kazakhstan
5. Alipour A, Hindle A, Rutgers T, Dawson R, Timbers F, Aggarwal K (2013) Bug reports dataset. <https://github.com/kaggarwal/Dedup>. Accessed 25 Feb 2019
6. Soleimani Neysiani B, Babamir SM (2019) Automatic interconnected lexical typo correction in bug reports of software triage systems. Paper presented at the international conference on contemporary issues in data science, Zanjan, Iran
7. Soleimani Neysiani B, Babamir SM (2019) Fast language-independent correction of interconnected typos to finding longest terms. Paper presented at the 24th international conference on information technology (IVUS), Lithuania
8. Zhuang L, Jing F, Zhu X-Y (2006) Movie review mining and summarization. In: Proceedings of the 15th ACM international conference on information and knowledge management, 2006. ACM, pp 43–50
9. Kukich K (1992) Techniques for automatically correcting words in text. *ACM Comput Surv (CSUR)* 24(4):377–439
10. Lai KH, Topaz M, Goss FR, Zhou L (2015) Automated misspelling detection and correction in clinical free-text records. *J Biomed Inform* 55:188–195
11. Chen Q, Li M, Zhou M (2007) Improving query spelling correction using web search results. In: Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)

12. Noaman HM, Sarhan SS, Rashwan M (2016) Automatic Arabic spelling errors detection and correction based on confusion matrix-noisy channel hybrid system. *Egypt Comput Sci J* 40(2):54–64
13. Ribeiro J, Narayan S, Cohen SB, Carreras X (2018) Local string transduction as sequence labeling. In: *Proceedings of the 27th international conference on computational linguistics*, 2018, pp 1360–1371
14. Korpusik M, Collins Z, Glass J (2017) Character-based embedding models and reranking strategies for understanding natural language meal descriptions. In: *Proc Interspeech*, pp 3320–3324
15. Almeida GAM (2016) Using phonetic knowledge in tools and resources for natural language processing and pronunciation evaluation. Universidade de São Paulo
16. de Mendonça Almeida GA, Avanço L, Duran MS, Fonseca ER, Nunes MG, Aluísio SM (2016) Evaluating phonetic spellers for user-generated content in Brazilian Portuguese. In: *International conference on computational processing of the Portuguese language*, 2016. Springer, pp 361–373
17. Huang Y, Murphey YL, Ge Y (2015) Intelligent typo correction for text mining through machine learning. *Int J Knowl Eng Data Min* 3(2):115–142
18. Huang Y, Murphey YL, Ge Y (2013) Automotive diagnosis typo correction using domain knowledge and machine learning. In: *IEEE symposium on computational intelligence and data mining (CIDM)*, 2013. IEEE, pp 267–274
19. Duan H, Hsu B-JP (2011) Online spelling correction for query completion. In: *Proceedings of the 20th international conference on World Wide Web*, 2011. ACM, pp 117–126
20. Hsu B-J, Wang K, Duan H (2012) Online spelling correction/phrase completion system. Google Patents
21. Oflazer K (1996) Error-tolerant tree matching. In: *Proceedings of the 16th conference on computational linguistics*, vol 2. Association for Computational Linguistics, pp 860–864
22. Shang H, Merrettal T (1996) Tries for approximate string matching. *IEEE Trans Knowl Data Eng* 8(4):540–547
23. Deorowicz S, Ciura MG (2005) Correcting spelling errors by modeling their causes. *Int J Appl Math Comput Sci* 15:275–285
24. Ito N (1997) Character-string retrieval system and method. Google Patents
25. Fafalios P, Tzitzikas Y (2015) Type-ahead exploratory search through typo and word order tolerant autocompletion. *J Web Eng* 14(1&2):80–116

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.