An Efficient Approach to Restructure Natural Language Text

Dr. R. N. Kulkarni[1] and Swetha Koduri[2]

[1]Department of Computer science & Engineering, Ballari Institute of Technology & Management, Ballari.
[2]Department of Information Technology, Malla Reddy College of Engineering and Technology, Hyderabad, Research Scholar VTU Belgavi

[1]rnkulkarni07@gmail.com [2]koduriswetha@gmail.com

***Abstract**: During the recent times, there is enormous growth found in the domain of software development across the world. Many organizations are automating all the activities in the organization and for the development of any customized application; the developing organization needs to gather the requirements from the client organization. The gathered requirements may be structured or unstructured because of the flexibility and ambiguity in the English language. The natural language sometimes has the problem of flexibility because of which the same term may be written in a different way. To overcome the problems of natural language statements, in this paper an automated tool is proposed to restructure the sentences, paragraphs, simple statements, compound statements and pages which are grammatically correct. This approach takes the input text and converts compound statements into simple statements, removing duplicate statements, places one statement per line and assigns sequence numbers to each physical statement. This enables us to extract the correct and complete meaning of the statements for further processing and output achieved is used for abstraction of design elements.*

Keywords: **natural language processing, restructuring, simple statements, compound statements**

1. INTRODUCTION

Software restructuring is driven by natural language processing applications, current trends in technology, growing competition, customer expectations, marketing intelligence etc. to generate standard models. Customer analytics are well captured using unstructured data. As a result, it identifies required patterns for making intelligent business decisions and will maintain good relationship with customers in the competitive environment. At the same time, on the other side it is essential to have right methods and tools to make unstructured data to be structured for further processing to get timely responses and accuracy.

Processing and extracting information from unstructured data is a challenging task and requires frequent refinements to get into a meaningful information extraction. Despite the existing work on the extraction of required information from natural language, the important aspects such as elimination of imbalanced data and procedure selection in each stage of restructuring has not been explored and still requires manual interventions.

The above limitations triggered the need to design and develop a novel methodology for restructuring the natural language text. An effective approach for restructuring is essential to extract correct and complete meaning from the given input in grammatically correct text using a novel restructuring algorithm. Restructuring the given input is done by converting compound statements into simple statements, removing duplicate statements, making one line statement, and assigning sequence numbers to each physical statement.

2. LITERATURE SURVEY

[1], the author developed a tool to restructure the java program that could be amenable for reengineering. This paper helped us to foster a novel restructuring algorithm for natural language input. [2], the authors proposed an innovative novel approach and developed the tool for the abstraction of the class diagram from the java program and represent the class diagram in the form of a table. This paper helped us identify tool for extracting design components from java code and respective abstraction issues. [3], the authors recommended a tool to restructure the java program without changing its functionality. This paper helped us to identify space and time complexities associated to the result of restructuring.

[4] The authors proposed a tool to restructure the input C++ program. By this paper we learned to identify what all we can remove or ignore in program components while doing restructuring with maximum effort of looking at completeness and correctness of expected output within in the time.[5] Proposed a method using Basic Stanford Dependency, Collapsed Stanford Dependency and Modifies Stanford Dependency for simplifying sentence by converting complex and compound sentences into simple ones as well as it also organizes the simple sentences of an input corpus from other types of sentences. This paper helped us to deal with dependency structure effectively in our proposed work.

[6] In this study they introduced an interface NLQBI using dependency parsing for both technical and nontechnical users was. The buffering scheme is also proposed for natural language statements which will not store the whole sentence if it was done previously. Also, there was a need of generalized access to all tables from database which is handled in this system. The proposed scheme is an interface between user and database and fetches the outcome of user's query in structured form. The future scope of this approach is to support complex queries, nested queries and complex join operations which will make this system much stronger. The buffering scheme in this paper inspired us to initiate this scheme for repeated sentences elimination in our proposed work by having detailed investigation of the scheme.

The authors in this study [7] proposed an effective tokenization approach on given documents to generate more precise and meaningful information. Tokenization involves pre-processing of documents and generates its respective tokens and reduces search space. Token generated is the parameters used for analysis of result. In this paper reduction procedure is very good reference for the reduction of input document if a useless word exists in our proposed researchwork [8] the authors adapted statistical machine translation to perform text simplification, taking advantage of large- scale paraphrases learned from bilingual texts and a small number of manual simplifications with multiple references.This work is the first

to design automatic metrics that are effective for tuning and evaluating simplification systems, which will facilitate iterative development for this task, this paper very much applicable for adapting and designing best methods which gives efficient results by using several metrics. In this study they [9] introduced a different model called DRESS (as shorthand for Deep Reinforcement Sentence Simplification), explores the space of possible simplifications while learning to optimize a reward function that encourages outputs which are simple, fluent, and preserve the essence of the input. In future Scope authors like to simplify entire document. From this paper we acquired knowledge to improve sentence simplification results with reinforcement learning.

[10] The authors have developed a syntax-driven rule-based text simplification framework that simplifies the linguistic structure of input sentences and produces high accuracy rate with very low information loss. From this paper we have learnt how to split conjoined clauses into separate sentences and how to apply paraphrasing operations this is very much useful for my proposed work by proposing additional step for unused information elimination. In this article they [11] presented an automated approach to simplify sentences and a tool is presented to tag syntactic complexity in sentence analysis achieved acceptable levels of accuracy. In this paper still there is a necessity in improving sentencetransformation. This is going to be relevant reference for syntactic ambiguities in our proposed method.

This paper, [12] proposed a new sentence simplification approach Split-and-Rephrase to split a complex sentence intoa meaning preserving sequence of shorter sentences This paper detailed the different models for sentence simplification with self-evidence of each model enabled us to propose new approach by it analyzed end results of each model.[13]In this paper they proposed a method by manually created parallel corpus of original and simplified sentences to preserve meaning between original and simplified sentences. This work is good reference for developing automatic syntactic simplification algorithm in proposed methodology.

[14] Proposed an approach to classify unstructured data, e.g., development documents, into natural language text and technical information with a mixture of text heuristics and agglomerative hierarchical clustering. In current and future work, authors want to evaluate a more detailed classification that splits technical data into the class's code, stack traces or log files and patches and evaluate the influence of clustering for these classes. Then, the heuristics and the clustering algorithm should consider a confidence weighting for each class. Finally, since the approach can be used as preprocessing for other, especially trained, algorithms like Naive Bayes or Support Vector Machines to 'clean up' noisy data with and without preprocessing. This papers work is a base for restructuring text document in our proposed work. The authors [15] proposed software for restructuring is used to textual content. As a procedural program is composed of functions calling each other, a document can be modeled as content fragments connected each other through links and rules and pre- and post-conditions could be defined and formalized for text.

3. TERMINOLOGY

Restructuring: Process of making unstructured data to be structured one without missing actual idea and

Compound sentences: Sentences of which the immediate constituents are two or more coordinate clauses.

Simple sentences: These independent clauses for which no element is clausal. Elements realized as phrases in simplesentences may themselves be complex and include embedded clauses of various types, including compounds (e.g., object elements realized as noun phrases with post-modifying relative clauses.

4. PROPOSED METHODOLOGY

The proposed methodology is to restructure the input sentence, paragraph, simple statement, compound statement which are grammatically correct, and this enables to generate meaningful statements with maximum scope of accuracyafter restructuring. Our proposed method for restructuring text has four steps:

1. Translation of compound statements into simple statements using carefully selected sentence splitting rulesaccording to Standard English grammar.
2. Removing duplicate statements.
3. Placing one statement in single line.
4. Assigning sequence number to each physical statement

Translation of compound statements into simple statements

Recently many translation methods were proposed for splitting long and compound statements into simple ones with some portion of manual interventions in it. Our proposed method translates compound statements into simple statements using novel restructuring algorithm.

Removing duplicate statements

Output of above step is considered for identifying and removing duplicate statements for reducing time and space complexity.

Placing one statement in one line

Statements generated by the above step are represented one statement in single line by considering simple statement macro algorithm. Further simplification gets very easy by classifying actual subject with its verb forms in a separate line; we can also avoid some relational errors and can make statements in a more comprehensible form.

Assign a sequence number to each physical statement

In this step assigning sequence numbers to each physical statement is done. A common example ofa sequence numbering is to identify independent tasks. Accuracy is generally improved by making the sequence numbers for given statements dependent on the choices of nearby statements.

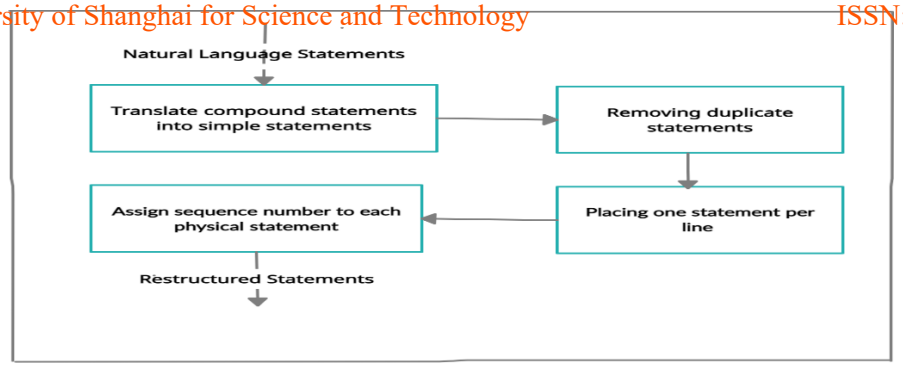

Fig 1: Block Diagram of Novel Restructuring Process

NOVEL RESTRUCTURING ALGORITHM: To restructure the input Natural language Statements.

Input - Natural Language Statements
Output – Restructured statements

Notations –**CS**: **Compound Statements, SS: Simple Statements**

1. Read the entire file until the end of line and parse each statement until full stop.
2. for Sentence S1 contains different CS
3. do
4. Syntactic Parsing (S1 for CS)
5. Dependency Parsing (S1)
6. if found
7. then
8. Replace CS (“for, and,nor, but,or, yet”, “.”)
9. Split at full stop (S1, “.”)
10. Simple Statements get marked independently with SS-tags
11. if found SS-tags
12. then
13. RemoveDuplicateStatements(SS-tags)
14. NewLine(SS-tags)
15. AssignSequenceNumber(SS-tags)
16. Return

5. CASE STUDY

“A woman was trying to compose an essay, but the essay was too difficult, and instead of simply giving up, the woman decided that she would do some research on internet to learn more about how to compose more effectively. The women’s progress was quick, and she learned a lot in short time. She was in fact quite happy about her progress.”

By considering the above paragraph as input and applying novel restructuring steps are shown in the below table

Table 5.1: Output of Restructuring

S. No.	Type of Statement	Input Statement	Method	Result
1	Compound Statement	A woman was trying to write an essay, but the essay was too difficult, and instead of simply giving up, the woman decided that she would do some research on	Steps to be applied **Step1:** In the statement ‘,’,’but’, ’and’ we identified, remove and replace with full stop. **Step2:** There are no duplicate statements	**1.** A woman was trying to write an essay. **2.**the essay was too difficult **3.** instead of simply giving up the woman decided that she would
		internet to learn more about how to write more effectively.	**Step3:** Place one line one statement. **Step4:** Assigns sequence number to physical statement.	do some research on internet to learn more about how to write more effectively.
2	Compound Statement	The women’s progress was quick, and she learned a lot in short time.		**4.**The women’s progress was quick **5.** She learned a lot in short time.
3	Simple statement	She was in fact quite happy about her progress.	Steps to be applied **Step1:** Go to step2 as it is not compound statement. **Step2:** Go to step3 as it is not having duplicate statements **Step3:** Place one line one statement. **Step4:** Assigns sequence number to physical statement.	**6.** She was in fact quite happy about her progress.

6. PERFORMANCE ANALYSIS

In this part, the comparison on both cases of Restructured and Unstructured on given input text documents are shown. The analysis shown in this paper is based on test conducted on data set which consists of around 3000 paragraphs. The comparison is made based on the following parameters.

(1) Number of simple sentences: The total number of simple sentences generation varies in the unstructured text document and Restructured text document. Simple sentence generation with novel restructuring algorithm is more accurate and effective with respect to input document, which results more accurate results to the user. Simple sentence generation without Restructuring leads to huge number of incorrect sentences and incomplete sentences, which is difficult to process and influence user outcome skeptically.

(2) Method: The method used in the paper restructuring the Natural Language text document. Restructuring is a

process which generates more meaningful simple sentences for the accurate subject and object extraction based on carefully selected sentence splitting rules and grammar principles.

(3) Time Constraint: Time taken to generate simple sentences in entire process is directly proportional to performance measure of a Natural Language processing system, as it deeply affects the Indexing and Semantic information.

The performance analysis mainly shows the given input text document gets into restructuring algorithm can maintain accurate statements based on result retrieval. Using novel restructuring algorithm, we will get a greater number of sentences, but it is the right way to extract more accurate and while for same set of input documents another strategy (without Restructuring) effect in accuracy of results retrieval.

7. CONCLUSION

In this paper, we have proposed a novel restructuring algorithm that takes input text which is grammatically correct and then translates the compound statement into simple statement, removes the duplicate statements, places one line per statement, and finally assigns a physical sequence number to each statement. Further, the output achieved is used for the abstraction of design elements.

8. REFERENCES

[1] Dr. R. N. Kulkarni, P. Pani Rama prasad, “Restructuring of Java Program to be amenable for Reengineering”, is published Journal of Engineering Science and Technology, Volume 02, Issue 06 (2019) (Pages: 01-07).

[2] Dr. R.N. Kulkarni, P.Pani Rama Prasad, "Abstraction of UML Class Diagram from the Input Java Program", Int. J. Advanced Networking and Applications, Volume: 12 Issue: 04 Pages: 4644-4649(2021) ISSN: 0975-0290.

[3] Dr. R.N. Kulkarni, AparnaK.S, "A Novel Approach to Restructure the Input Java Program", Int. J. Advanced Networking and Applications, Volume: 12 Issue: 04 Pages: 4621-4626(2021) ISSN: 0975-0290.

[4] Dr. R.N. Kulkarni ,Venkata Sandeep Edara,” A Novel Approach to Restructure The Legacy C++ Program”, Journal of Huazhong University of Science and Technology, Vol:50 Issue:05 ISSN-1671-4512.

[5] Das B, Majumder M, Phadikar S, “A novel system for generating simple sentences from complex and compound sentences” International Journal of Modern Education and Computer Science. 2018; Vol: 12 Issue: 01PgNo:57-64 Published Online January 2018 in MECS (http://www.mecs-press.org/) DOI: 10.5815/ijmecs.2018.01.06.

[6] Kokare R, Wanjale K., “A natural language query builder interface for structured databases using dependency parsing”, International Journal of Mathematical Sciences and Computing, 2015 Nov;Vol: 04Issue: 02 P:11-20. DOI: 10.5815/ijmsc.2015.

[7] Singh V, Saini B., “An Effective tokenization algorithm for information retrieval systems”, Department of Computer Engineering, National Institute of Technology Kurukshetra, Haryana, India. 2014. p. 109–119, 2014. © CS & IT-CSCP 2014 DOI: 10.5121/csit.2014.4910

[8] Xu W, Napoles C, Pavlick E, Chen Q, "Callison-Burch C. Optimizing statistical machine translation for text simplification", Transactions of the Association for Computational Linguistics.2016Jul;4:401-15.DOI: 10.1162/tacl_a_00107.

[9] Zhang X, Lapata M, "Sentence simplification with deep reinforcement learning",Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing2017 Mar 31. P.584–594, DOI:10.18653/v1/D17-1062.

[10] Niklaus C, Bermeitinger B, Handschuh S, Freitas A, "A sentence simplification system for improving relation extraction" , Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations, pages 170–174, Osaka, Japan, December 11-17 2016, DOI:arXiv:1703.09013 .

[11] Evans R, Orasan C., "Identifying signs of syntactic complexity for rule-based sentence simplification. Natural Language Engineering ", 2019 Jan; Vol: 25 Issue:01, P:69-119.DOI:10.1017/S1351324918000384.

[12] Narayan S, Gardent C, Cohen SB, Shimorina A.," Split and rephrase", Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, P: 606–616, DOI:10.18653/v1/D17-1064.

[13] Collados JC," Splitting complex sentences for natural language processing applications: Building a simplified spanish corpus", Procedia-Social and Behavioral Sciences. 2013 Oct 25, Vol:95, P: 464-72.DOI: 10.1016/j.sbspro.2013.10.670.

[14] Aversano L, Canfora G, De Ruvo G, Tortorella M.," An approach for restructuring text content", In2013 35th International Conference on Software Engineering (ICSE) 2013 May 18 (pp. 1225-1228).
DOI: 10.1109/ICSE.2013.6606684.

[15] Andrianjaka RM, Luc RJ, Mahatody T, Ilie S, Raft RN,"Restructuring extended Lexical elaborate Language", In2019 23rd International Conference on System Theory, Control and Computing (ICSTCC) 2019 Oct 9 ,P:266-272, DOI: 10.1109/ICSTCC.2019.8886081