Mining Algorithmic Complexity in Full-text Scholarly Documents

Abu Bakar[1[0000-0002-8006-8886]], Iqra Safder[1[0000-0001-9818-4693]]
and Saeed-Ul Hassan[1[0000-0002-6509-9190]]

[1] Information Technology University, Ferozepur Road, Lahore 54000, Pakistan
saeed-ul-hassan@itu.edu.pk

1 Introduction

Non-textual document elements (NTDE) like charts, diagrams, algorithms play an important role to present key information in scientific documents [1]. Recent advancements in information retrieval systems tap this information to answer more complex queries by mining text pertaining to non-textual document elements. However, linking between document elements and corresponding text can be non-trivial. For instance, linking text related to algorithmic complexity with consequent root algorithm could be challenging. These elements are sometime placed at the start or at the end of the page instead of following the flow of document text, and the discussion about these elements may or may not be on the same page. In recent years, quite a few attempts have been made to extract NTDE [2-3]. These techniques are actively applied for effective document summarization, to improve the existing IR systems. Generally, asymptotic notations are used to identify the complexity lines in full text. We mine the relevant complexities of algorithms from full text by comparing the metadata of algorithm with context of paragraph in which complexity related discussion is made by authors. In this paper, we presented a mechanism for identification of algorithmic complexity lines using regular expressions, algorithmic metadata compilation of algorithms, and linking complexity related textual lines to algorithmic metadata.

2 Data and Methodology

Our dataset contains 47 articles, carefully selected from CiteSeerX repository. Note that, every document in our dataset must have an algorithm and its related time complexities mentioned in full text. Firstly, complexity lines are identified by using regular expressions and their context is built from five lines before and after complexity line. For linking purpose, we manually designed a reference document that contains some linking information related an algorithm and its complexity lines. The reference file contains around 471 links, identified between algorithms and their run time and space complexities. Secondly, inspired by the work [4-5], algorithmic metadata lines are extracted and combined for each algorithm. Afterwards, we created an association file, by linking and comparing both complexity synopsis and algorithmic metadata.

Note that these association files can be used for ranking and indexing the algorithms for IR systems.

Table 1. Precision, recall, f-measure and accuracy for algorithm and complexity linking

Name	Description	Precision	Recall	F1 Score	Accuracy
NFNC50	No-FK, No-CW, 50% threshold	0.66	0.77	0.71	0.56
FNC50	FK, No-CW, 50% threshold	0.78	0.76	0.77	0.64
FC55	FK, CW, 55% threshold	0.61	0.75	0.67	0.62
FNC50FC55	Combination of FNC50 and FC55	**0.81**	0.75	**0.78**	**0.65**

3 Experimental Results

In order to measure the effectiveness of our designed technique we performed four different experiments with different keywords such as frequent keywords (FK), cue words (CW) and threshold values. Table 1 shows the achieved results for all experiments. Overall, FNC50FC55 (combination of FNC50 and FC55) has outperformed with 0.78 f-score and 0.65 accuracy.

4 Conclusions

We presented a mechanism to link relevant complexities of algorithms from full text by comparing the metadata of algorithm with context of paragraph in which complexity related discussion is made by authors. In future we can use similar linking methodology to link different non-textual document elements like figures, tables and charts to their relevant paragraphs in a full-text document. Note that data and code used in this paper is available on following URL: https://github.com/slab-itu/icadl_link_algo

References

1. Al-Zaidy, R.A. and Giles, C.L.: A Machine Learning Approach for Semantic Structuring of Scientific Charts in Scholarly Documents. In: AAAI, pp. 4644-4649. (2017).
2. Safder, I., Hassan, S. U., Aljohani, N. R.: AI Cognition in Searching for Relevant Knowledge from Scholarly Big Data, Using a Multi-layer Perceptron and Recurrent Convolutional Neural Network Model. In: Companion of the The Web Conference 2018 on The Web Conference 2018, pp. 251–258 (2018).
3. Tuarob, S., Bhatia, S., Mitra, P., Giles, C. L.: AlgorithmSeer: A system for extracting and searching for algorithms in scholarly big data. IEEE Transactions on Big Data, 2(1), pp.3–17 (2016).
4. Safder, I., Sarfraz, J., Hassan, S.U., Ali, M. and Tuarob, S.: Detecting Target Text Related to Algorithmic Efficiency in Scholarly Big Data Using Recurrent Convolutional Neural
5. Network Model. In: International Conference on Asian Digital Libraries, pp. 30-40. Springer, Cham (2017).
6. I. Safder and S.-U. Hassan, "DS4A: Deep Search System for Algorithms from Full-text Scholarly Big Data" In: International Conference on Data Mining Workshop. (2018).