DOCUMENT SUMMARIZATION USING TRANSFER LEARNING

CHONG JING WEN

UNIVERSITI TEKNOLOGI MALAYSIA

Replace this page with form PSZ 19:16 (Pind. 1/07), which can be obtained from SPS or your faculty.

Replace this page with the Cooperation Declaration form, which can be obtained from SPS or your faculty. This page is ***OPTIONAL*** *when your research is done in collaboration with other institutions that requires their consent to publish the finding in this document.]*

DOCUMENT SUMMARIZATION USING TRANSFER LEARNING

CHONG JING WEN

A project report submitted in partial fulfilment of the
requirements for the award of the degree of
Master of Engineering (Computer and Microelectronic Systems)

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

JUNE 2018

To my family and friends and whoever providing support to me. Thank You

ACKNOWLEDGEMENT

First of all, I wanted to thank you to my supervisor, Prof. Muhammad Mun'im Ahmad Zabidi. He is putting a lot of efforts in guiding me in this project. Thank you for your patience and guidance which bring me to complete this project.

Besides that, I would like to express my sincere gratitude to my family. They actually supporting me in all the time. Especially my little brother who keep sharing technologies. Thank you for all the cares.

I would also like to thank the developers of the utmthesis LaTeX project for making the thesis writing process a lot easier for me. Thanks to them, I could focus on the content of the thesis, and not waste time with formatting issues. Those guys are awesome.

Lastly, thank you to Dr. Usman Ullah Sheikh who reviewd on my design and provide some suggestion on debugging my model. His suggestion have provide me a clear direction on deep learning which enable me in further understanding the deep learning model design.

Jing Wen, Penang, Malaysia

ABSTRACT

Document summarization refers to an automation method to shortening a document into a short and meaningful article. In computing, automatic summarization can basically split to two approaches where it can be done with classical method where the rank of the text is calculated and word is extracted with respect to the word's rank. In the other hand, the task can be completed by using modern method, deep learning. In deep learning, it is slightly different that the programming is mainly prepare a model where it will learn to summarize the document. However, som pre-processing on the data is needed before it is fit into the deep learning model. All the detailed part will be discussed in this project. For this project, the sequence to sequence model will be used as the main computing unit. On top of that, word embedding layer will helps in the summarization by providing the knowledge of word's relationship. By combining these two design, the deep learning model is able to differentiate the word with respect to the relationship. Of course, this project included some pre-processing where the data will pre-filtered and convert to data that recognized by the model. Similarly, the output will be converted back to word that understand by the human. At the end, the summarized output will be evaluate by BLUE, a benchmark for sentence similarity. As a result, the model can achieve loss as low as 0.8% and accuracy of 32%. Overall, the accuracy is capped by the size of the model. It is due to the reason that the model does not support high number of vocabulary. The design can be further improve by increasing the vocabulary size. However, the training process need to be completed by using a better hardware. In addition, the covered text is evaluated the BLEU value with respect to the expected output summary. Overall, a trainable model is designed. The model can be further improve by adding vocabulary size as well as increasing all the training set.

ABSTRAK

Kesimpulan dokumen merujuk kepada kaedah automasi untuk memendekkan dokumen ke dalam artikel ringkas dan bermakna. Dalam pengkomputeran, ringkasan automatik pada asasnya boleh dibahagikan kepada dua pendekatan di mana ia boleh dilakukan dengan kaedah klasik di mana pangkat teks dikira dan perkataan diekstrak berkenaan dengan pangkat perkataan. Di sisi lain, tugas itu boleh diselesaikan dengan menggunakan kaedah moden, pembelajaran mendalam. Dalam pembelajaran yang mendalam, sedikit berbeza bahawa pengaturcaraan ini terutama menyediakan model di mana ia akan belajar untuk meringkaskan dokumen itu. Walau bagaimanapun, pra-pemprosesan data diperlukan sebelum ia sesuai dengan model pembelajaran yang mendalam. Semua bahagian terperinci akan dibincangkan dalam projek ini. Untuk projek ini, urutan ke urutan model akan digunakan sebagai unit pengkomputeran utama. Di samping itu, lapisan embedding perkataan akan membantu dalam ringkasan dengan memberikan pengetahuan tentang hubungan perkataan. Dengan menggabungkan dua reka bentuk ini, model pembelajaran mendalam dapat membezakan perkataan berkenaan dengan hubungan. Sudah tentu, projek ini termasuk beberapa pra-pemprosesan di mana data akan diprataskan dan diubah kepada data yang diiktiraf oleh model. Begitu juga, output akan ditukar kembali kepada perkataan yang difahami oleh manusia. Pada akhirnya, output yang diringkaskan akan dinilai oleh BLUE, tanda aras untuk persamaan ayat. Oleh itu, model boleh mencapai kerugian serendah 0.8 % dan ketepatan 32 %. Secara keseluruhan, ketepatannya dihadkan oleh saiz model. Ia disebabkan oleh sebab model itu tidak menyokong bilangan perbendaharaan kata yang tinggi. Reka bentuk ini boleh terus ditingkatkan dengan meningkatkan saiz perbendaharaan kata. Walau bagaimanapun, proses latihan perlu diselesaikan dengan menggunakan perkakasan yang lebih baik. Di samping itu, teks yang dilindungi akan dinilai nilai BLEU berkenaan dengan ringkasan output yang diharapkan. Secara keseluruhannya, model yang boleh dilatih direka. Model ini boleh bertambah baik dengan menambahkan saiz perbendaharaan kata.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
	DECLARATION	ii
	DEDICATION	iii
	ACKNOWLEDGEMENT	iv
	ABSTRACT	v
	ABSTRAK	vi
	TABLE OF CONTENTS	vii
	LIST OF TABLES	ix
	LIST OF FIGURES	x
	LIST OF ABBREVIATIONS	xi
1	**INTRODUCTION**	1
	1.1 Problem Background	1
	1.2 Problem Statement	2
	1.3 Research Aim and Objectives	2
	1.4 Scope	3
	1.5 Thesis Outline	4
2	**LITERATURE REVIEW**	5
	2.1 Natural Language Processing	5
	2.1.1 Classical Model	5
	2.1.2 Deep Learning	8
	2.1.2.1 Neural Networks	9
	2.1.2.2 Recurrent Neural Networks	9
	2.1.2.3 Long Short Term Memory	11
	2.1.2.4 Sequence to sequence model	13

	2.2	Previous Work	14
3	**RESEARCH METHODOLOGY**		17
	3.1	Project Flow	17
	3.2	Project Tools	19
		3.2.1 Tensorflow	20
		3.2.2 NLTK	20
		3.2.3 Keras	20
		3.2.4 Anaconda	21
		3.2.5 PyCharm	21
	3.3	Overall Structure	21
	3.4	Transfer Learning	23
	3.5	Text Parsing	24
	3.6	Bilingual Evaluation Understudy	25
	3.7	GloVe	25
	3.8	Sequence to sequence model	26
	3.9	Integrating GloVe and sequence to sequence model	27
	3.10	Training Strategy	31
4	**RESULT AND ANALYSIS**		33
	4.1	Training the Model	33
	4.2	BLEU Score	35
5	**CONCLUSION**		38
	5.1	Overall design	38
	5.2	Limitation	39
	5.3	Future Work	39
REFERENCES			40

LIST OF TABLES

TABLE NO.	TITLE	PAGE
2.1	Reviewed papers	16

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE
2.1	Classical Model Flow Chart	7
2.2	Rolled Recurrent Neural Network	10
2.3	Unrolled Recurrent Neural Network	10
2.4	Standard RNN with single layer	11
2.5	Standard Sequence to sequence model	13
3.1	Project Flow Chart	19
3.2	Overall Structure	22
3.3	Changes of data inside the system. Left side is the data outside the sequence to sequence model. While the right side represent the data changes inside the sequence to sequence model.	23
3.4	Text Parsing	24
3.5	Modified Sequence to Sequence model	26
3.6	Flow Chart for text parsing and seq2seq training	28
3.7	Full Model Training Mode	29
3.8	Full Model Inference Mode	30
3.9	Inference Mode Handling	31
4.1	Accuracy graph	34
4.2	Accuracy graph	36
4.3	BLEU Analysis	37

LIST OF ABBREVIATIONS

CNN	-	Convolutional Neural Network
GloVe	-	Global Vectors for Word Representation
LSTM	-	Long Short Term Memory
NLP	-	Natural Language Processing
NLTK	-	Natural Language Tool Kit
RNN	-	Recurrent Neural Network
seq2seq	-	Sequence to sequence
TF-IDF	-	Term Frequencyv-Inverse Document Frequency

CHAPTER 1

INTRODUCTION

This is thesis documents information, methods, and ideas on how to implement deep learning for text summarization task

1.1 Problem Background

The world is facing an ever-growing volume of information of various forms. The latest technology is able to gather large amounts of information in a short period. For example, the news from United States can deliver to South East Asia with a period of seconds. Besides that, search engines such as Google provides information from multiple sources in a single search. All the mentioned scenarios creates information which humans are simply incapable of handling. Let's consider the case of academic research. The candidate needs to read a lot of documents in order to complete literature review of his research. The candidate needed to spend months or year to understand the research topic clearly. This process is unavoidable and needed in order for the candidate to do a quality research.

Looking deeper, the documents being reviewed by the candidate above consists a lot of unwanted information. The unwanted information may include the motivation of the document, repeated messages, or examples attached in the document. The unwanted information is an extra load to the candidate, and these information does not benefit the review progress and yet slow it down. Imagine that the document is pre-processed to remove the unwanted information and shortening the length of the document by extracting only the important information from the document. As we understand, the mentioned pre-processing is named as summarization. With the existence of the document summaries, the candidate can have a quick view on the document. The full document is needed when the topic is related and useful to the candidate, else that document can be skipped without spending extra time to review it. Summarization greatly reduces the workload of the candidate. The process of summarization providing definite benefits to parties that constantly need to read multiple documents.

1.2 Problem Statement

For document summarization, we might need to summarize it with the help of human and write out a summary according to the document that have read as a pre-work and next read by candidates who need spend effort on that document. With this approach, it is clearly showing that the process does not improve the time required to complete the literature reviews. There are few approaches that can be taken to overcome the problem stated. We can either pre-summarize the document and store them in a database, in the case that document is needed, the user can go to the database and query the required summary. But, this does not really solve the problem. In this new era, technology advancement is rapid and the document about the latest implementation is updating with an unpredictable manner where the summaries database update speed is slow compare to the document release speed. Therefore, it can be concluded that pre-summarize is not a solution to the problem we have. In this case, we need a just-in-time solution to summarize the document which will fulfill the need of document summary and the short wait time in getting the summary.

1.3 Research Aim and Objectives

To speed up the summarization process, automatic summarization is needed in this project where machine is playing a role to summarize the document and generate the respective summary. For the automatic summarization solution, it is part of machine learning and data mining where machine will find the most informative sentences and generate a respective summary from there. In this case, it is faster compare to the summarization that did by human. Automatic summarization can classified to extractive and abstractive. Extractive method works by selecting the words inside the document and reconstruct them as a summary of the document. Classically, this is done with some scoring method on the input texts and generate the output with respect to the scores. Or alternatively, this can be performed with machine learning. On the other hand, abstractive approach can be applied to perform the summarization. Typically, abstractive approach requires machine learning as part of the software for summarization. For abstractive summarization, it is defined as the software will ”understand” the original document first before generate summary for the document. In this approach, the output summary might or might not consist the words from the original. According to Giuseppe Carenini and Jackie Chi Kit Cheung [1], abstraction outperforms extraction by a larger amount in more controversial corpora. Therefore, abstraction is chosen as a strategy in this project. Therefore, the project is proceeding by choosing a suitable deep learning algorithms. The following are the research objectives:

1. To investigate the most suitable deep learning architecture for document summarization
2. To design a software to summarize document automatically
3. To evaluate the performance of the summarization software

1.4 Scope

Scope defines the limitations of this project:

- Using Sequence to sequence model as the language model in this project
- The model should expect 400 words input and 10 words output
- This project mainly design by using Python 3 on Window 10 with the help of Anaconda
- The model expect to output related word

1.5 Thesis Outline

This dissertation constructed by three chapters. For chapter 1 discuss about the background of the problem which included intention, problem statement, objectives, scope and outline of dissertation. In the chapter 2, it will be consisted literature review on the previous works and other related research. Consequently, on chapter 3, the research methodology will be discussed. In this chapter, detailed process on achieving the design will be presented as well as the benchmark system. Next, chapter 4 will display the result of this project and some evaluation. Lastly, limitation and future work of this project is discussed in chapter 5.

REFERENCES

1. Carenini, G. and Cheung, J. C. K. Extractive vs. NLG-based abstractive summarization of evaluative text: The effect of corpus controversiality. *Proceedings of the Fifth International Natural Language Generation Conference*. Association for Computational Linguistics. 2008. 33–41.

2. Perkins, J. *Python text processing with NLTK 2.0 cookbook*. United Kingdom: Birmingham. 2010.

3. Perkins, J. *Python 3 text processing with NLTK 3 cookbook*. United Kingdom: Birmingham. 2014.

4. Hochreiter, S. and Schmidhuber, J. Long Short-Term Memory. *Neural Comput.*, 1997. 9(8): 1735–1780. ISSN 0899-7667. doi:10.1162/neco.1997.9.8.1735. URL `http://dx.doi.org/10.1162/neco.1997.9.8.1735`.

5. M. Iyyer, J. B.-G., V. Manjunatha and Iii, H. D. Deep Unordered Composition Rivals Syntactic Methods for Text Classification. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, 2015. Volume 1: Long Papers.

6. J. Niu, Q. Z.-L. S., H. Chen and Atiquzzaman, M. Multi-document abstractive summarization using chunk-graph and recurrent neural network. *2017 IEEE International Conference on Communications (ICC)*, 2017.

7. Yin, W. and Pei, Y. Optimizing Sentence Modeling and Selection for Document Summarization. *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

8. Nallapati, e. a., Ramesh. Abstractive Text Summarization Using Sequence-to-Sequence RNNs and Beyond. *Proceedings of The 20th SIGNLL Conference*

on Computational Natural Language Learning, 2016.

9. Sutskever, I., Vinyals, O. and Le, Q. V. Sequence to Sequence Learning with Neural Networks. *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*. Cambridge, MA, USA: MIT Press. 2014, NIPS'14. 3104–3112. URL `http://dl.acm.org/citation.cfm?id=2969033.2969173`.

10. Pennington, J., Socher, R. and Manning, C. D. Glove: Global Vectors for Word Representation. *EMNLP*. 2014, vol. 14. 1532–1543.

11. Lori Buckland, E. A. G. DUC 2007 Past Data. *In Proceedings of the North American Chapter of the Association for Computational Linguistics Workshop on Document Understanding*, 2007: 112–119.