國立交通大學

資訊科學與工程研究所

碩士論文

基於卷積神經網路的論文自動生成技術

A CNN-based Automatic Thesis Generation Technique

研 究 生：王大明

指導教授：吳小松 教授

中華民國 109 年 12 月

基於卷積神經網路的論文自動生成技術

A CNN-based Automatic Thesis Generation Technique

研究生：王大明　　　　Student：Ta-Ming Wang

指導教授：吳小松 教授　　　　Advisor：Xiao-Sung Wu

國立交通大學

資訊科學與工程研究所

碩士論文

A Thesis

Submitted to Institute of Computer Science and Engineering
College of Computer Science
National Chiao Tung University
in Partial Fulfillment of the Requirements for the Degree of
Master in Computer Science

Dec 2020
Hsinchu, Taiwan, Republic of China

中華民國 109 年 12 月

基於卷積神經網路的論文自動生成技術

學生：王大明
指導教授：吳小松 教授

國立交通大學資訊科學與工程研究所

摘　　要

在大 AI、ML 時代，自己寫論文已經不再是個有效率的做法，因此我們提出了一套基於卷積神經網路的論文自動生成技術。

關鍵字：卷積神經網路、機器學習

A CNN-based Automatic Thesis Generation Technique

Student : Ta-Ming Wang
Advisor : Prof. Xiao-Sung Wu

Institute of Computer Science and Engineering
National Chiao Tung University

ABSTRACT

In the era of Big AI and ML, it is not efficient to write thesis by yourself anymore so that we propose a CNN-based approach for automatic thesis generation.

Keywords: convolutional neural network, machine learning

誌　　謝

在此感謝吳小松 教授在論文上給我的指導...

感謝實驗室的朋友們...

最後，感謝我的家人...

王大明　謹誌于
資工系資料科學與工程研究所
中華民國一零九年十二月

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Here is the introduction.

Chapter 2

Background

Here is the background.

Chapter 3

Design

Here is the design.

3.1 Feature Extraction

3.2 Thesis Modeling

3.3 Thesis Generation

Algorithm 1 Get Maximum of Two Numbers

```
procedure GetMaximum($a, b$)
    if $a \geq b$ then
        return $a$
    else
        return $b$
```

Figure 1: Pseudo Code of GetMaximum

Chapter 4

Implementation

Figure 2: TensorFlow's Logo

We implement the prototype on TensorFlow[1] platform. Figure 2 shows the logo of TensorFlow, and Figure 3 shows the code snippet of model training.

```
import tensorflow as tf

def train(total_loss, global_step):
  # Variables that affect learning rate.
  num_batches_per_epoch = NUM_EXAMPLES_PER_EPOCH / FLAGS.batch_size
  decay_steps = int(num_batches_per_epoch * NUM_EPOCHS_PER_DECAY)

  # Decay the learning rate exponentially.
  lr = tf.train.exponential_decay(INITIAL_LEARNING_RATE,
                                  global_step,
                                  decay_steps,
                                  LEARNING_RATE_DECAY_FACTOR,
                                  staircase=True)
  tf.summary.scalar('learning_rate', lr)
```

Figure 3: Code Snippet of Model Training

Chapter 5

Evaluation

Here is the evaluation.

5.1 Datasets

5.2 Experiment Design

5.3 Experimental Results

5.3.1 Training Time

Table 1 lists the training time of different datasets.

Table 1: Training Time

Dataset	**Training Time**
A	1 hour
B	2 hours
C	3 hours
D	4 hours
E	5 hours

5.3.2 Example of Generated Thesis

Chapter 6

Related Work

Here are the related works[2].

Chapter 7

Discussion

Some previous researches[3] worked on detecting the machine generated paper. However, to the best of our knowledge, all of them can not effectively detect the thesis generated by our system.

Chapter 8

Conclusion

Here is the conclusion.

References

[1] *TensorFlow*. URL: `https://pdos.csail.mit.edu/archive/scigen/`.

[2] MIT CSAIL. *SCIgen - An Automatic CS Paper Generator*. URL: `https://pdos.csail.mit.edu/archive/scigen/`.

[3] Jiping Xiong and Tao Huang. “An effective method to identify machine automatically generated paper”. In: *Knowledge Engineering and Software Engineering, 2009. KESE’09. Pacific-Asia Conference on.* IEEE. 2009, pp. 101–102.