การอ่านทำความเข้าใจด้วยเครื่องเพื่อคำถามที่มีหลายความสัมพันธ์ด้วยวิธีการเรียนรู้เชิงลึก

นายณัฐชัย ตรีทศายุธ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ ที่ส่งผ่านทางบัณฑิตวิทยาลัย
The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository (CUIR)
are the thesis authors' files submitted through the University Graduate School.

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2560
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

Machine Reading Comprehension for Questions with Multiple Relationships Using Deep Learning

Mr. Nattachai Tretasayuth

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering Program in Computer Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2017

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การอ่านทำความเข้าใจด้วยเครื่องเพื่อคำถามที่มีหลายความสัมพันธ์ด้วยวิธีการเรียนรู้เชิงลึก
โดย	นายณัฐชัย ตรีทศายุธ
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ ดร. พีรพล เวทีกูล
อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม	ดร. ปรัชญา บุญขวัญ

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.....คณบดีคณะวิศวกรรมศาสตร์

(รองศาสตราจารย์ ดร. สุพจน์ เตชวรสินสกุล)

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ

(ศาสตราจารย์ ดร. บุญเสริม กิจศิริกุล)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(ผู้ช่วยศาสตราจารย์ ดร. พีรพล เวทีกูล)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

(ดร. ปรัชญา บุญขวัญ)

.....กรรมการ

(อาจารย์ ดร. เอกพล ช่วงสุวนิช)

.....กรรมการภายนอกมหาวิทยาลัย

(ศาสตราจารย์ ดร. ธนารักษ์ ธีระมั่นคง)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ณัฐชัย ตรีทศายุธ : การอ่านทำความเข้าใจด้วยเครื่องเพื่อคำถามที่มีหลายความสัมพันธ์ด้วยวิธีการเรียนรู้เชิงลึก (Machine Reading Comprehension for Questions with Multiple Relationships Using Deep Learning) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ. ดร. พีรพล เวทีกูล, อ.ที่ปรึกษาวิทยานิพนธ์ร่วม: ดร. ปรัชญา บุญขวัญ, 63 หน้า.

การอ่านทำความเข้าใจเพื่อใช้ตอบคำถาม เป็นหนึ่งในเรื่องที่สำคัญและยากที่สุดในงานสายการประมวลผลภาษาธรรมชาติ วิธีการที่ได้รับความนิยมและให้ผลที่ดีที่สุดในปัจจุบัน คือการใช้โมเดลที่นำเอาการเรียนรู้เชิงลึกเข้ามาช่วยตอบ โดยโมเดลจะทำการหาคำที่คล้ายกันระหว่างคำถามและบทความเพื่อนำไปใช้ในการตอบคำถาม แต่โมเดลในรูปแบบนี้จะมีข้อจำกัดที่ไม่สามารถจะตอบคำถามซึ่งคำตอบจะต้องใช้การเชื่อมต่อคำในหลายประโยคเข้าด้วยกัน หรือที่เรียกว่าคำถามที่มีหลายความสัมพันธ์ได้ งานวิจัยชิ้นนี้ต้องการที่จะเสนอแนวทางในการใช้คำอ้างอิงเข้ามาช่วยแก้ปัญหาดังกล่าว รวมถึงยังได้เสนอวิธีการตอบแบบสองทาง และฟังก์ชันต้นทุนจากความยาวของคำตอบเพื่อเพิ่มประสิทธิภาพโดยรวมของระบบ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาควิชา	วิศวกรรมคอมพิวเตอร์	ลายมือชื่อนิสิต .....
สาขาวิชา	วิศวกรรมคอมพิวเตอร์	ลายมือชื่อ อ.ที่ปรึกษาหลัก .....
ปีการศึกษา	2560	ลายมือชื่อ อ.ที่ปรึกษาร่วม .....

5970156021 : MAJOR COMPUTER ENGINEERING

KEYWORDS: MACHINE READING COMPREHENSION / DEEP LEARNING / COREFERENCE / QUESTION ANSWERING

NATTACHAI TRETASAYUTH: Machine Reading Comprehension for Questions with Multiple Relationships Using Deep Learning. ADVISOR: ASST. PROF. PEERAPON VATEEKUL, CO-ADVISOR: PRACHYA BOONKWAN, 63 pp.

Machine reading comprehension (MC) is a challenging problem in natural language processing. Recently, many deep learning models have been proposed. However, these models are limited by the soft attention relied heavily on keywords that appear in a question. Therefore, the performance is always poor in a question that needs to infer an answer from multiple sentences, which cannot solely depend on keywords in a question. In this thesis, our contributions are three folds: (i) coreference vector, to improve the performance of multiple sentence question, (ii) bi-directional answering technique, to avoid local maxima corresponding to the incorrect answer span and (iii) answer length loss function, to better refine an answer.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

Department:	Computer Engineering	Student's Signature .....
Field of Study:	Computer Engineering	Advisor's Signature .....
Academic Year:	2017	Co-Advisor's Signature .....

กิตติกรรมประกาศ

การที่วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดีนั้น นอกจากการทำงานของตัวผู้วิจัยแล้ว ยังมีบุคคลท่านอื่นที่เป็นส่วนสำคัญที่ให้ความช่วยเหลือในการทำวิทยานิพนธ์ฉบับนี้ขึ้นมา ผู้จัดทำต้องขอขอบคุณบุคคลเหล่านี้ผู้ซึ่งทำให้เกิดผลสำเร็จนี้ขึ้นมาได้

ขอขอบคุณอาจารย์ที่ปรึกษา ผศ. ดร. พีรพล เวทีกูล ผู้ที่คอยให้ความช่วยเหลืออย่างเต็มที่ในการให้คำแนะนำ และกระตุ้นอยู่เสมอจนทำให้ผลงานฉบับนี้เกิดขึ้นมาได้

ขอขอบคุณอาจารย์ที่ปรึกษา ดร. ปรัชญา บุญขวัญ ผู้ที่คอยให้ความช่วยเหลือ ให้กำลังใจ และสละเวลามาให้คำแนะนำอยู่อย่างสม่ำเสมอ

ขอขอบคุณ ศ. ดร. บุญเสริม กิจศิริกุล ผู้ให้คำแนะนำในการแก้ปัญหาในการทำวิจัย รวมถึงการเป็นประธานในการสอบวิทยานิพนธ์

ขอขอบคุณกรรมการการสอบวิทยานิพนธ์ ศ. ดร. ธนารักษ์ ธีระมั่นคง ที่ให้คำแนะนำและเสนอสิ่งที่ควรทำเพิ่มเติมในการทำงานวิจัย

ขอขอบคุณกรรมการการสอบวิทยานิพนธ์ อ. ดร. เอกพล ช่วงสุวนิช สำหรับคำแนะนำและสอนเกี่ยวกับแนวงานวิจัยใหม่ๆ รวมถึงการเสนอสิ่งที่ควรทำเพิ่มเติมในการทำงานวิจัย

ขอขอบคุณอาจารย์ทุกท่าน ที่ได้สั่งสอนเรื่องต่าง ๆ ในหลักสูตร ซึ่งแนวคิดและกระบวนการเหล่านั้น ล้วนประกอบกันจนทาให้ผลงานชิ้นนี้ลุล่วงไปได้

ขอขอบคุณเพื่อน ๆ พี่ ๆ น้อง ๆ ในห้องปฏิบัติการณ์ที่คอยช่วยเหลือสิ่งต่าง ๆ ทั้งคอยให้การสนับสนุน เป็นกำลังใจ จนวิทยานิพนธ์ฉบับนี้สำเร็จ และขอให้ผลเหล่านี้ย้อนกลับไปถึงตัวพวกท่านเอง

สุดท้าย ขอขอบคุณคุณพ่อ คุณแม่ และครอบครัว ที่ให้การสนับสนุน และส่งเสริมทั้งทางด้านการศึกษา และทางด้านการใช้ชีวิต จวบจนปัจจุบัน

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

สารบัญ

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญรูป

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

สารบัญตาราง

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของปัญหา

การตอบคำถาม (Question Answering) เป็นหนึ่งในปัญหาที่ยากที่สุดในการประมวลผลภาษาธรรมชาติ (Natural Language Processing) เนื่องจากการที่จะตอบคำถามได้อย่างถูกต้อง คอมพิวเตอร์จะต้องทำความเข้าใจคำถามที่เป็นภาษาธรรมชาติ และสามารถที่จะใช้หลักเหตุผลเพื่อที่จะหาตอบคำตอบที่ถูกต้อง ตัวอย่างของแอพพลิเคชั่นที่นำไปใช้ได้แก่ โปรแกรมสำหรับตอบคำถาม เช่น เว็บไซต์สำหรับตอบคำถาม วูลแฟรม อัลฟ่า (Wolfram Alpha) ไอบีเอ็มวัตสัน (IBM Watson) หรือเป็นส่วนนึงในระบบผู้ช่วยส่วนตัว เช่น คอร์ตาน่า (Cortana) สิริ (Siri)

งานวิจัยด้านการตอบคำถามสามารถแบ่งได้เป็น 2 กลุ่มใหญ่ ได้แก่ การตอบคำถามด้วยการค้นหาข้อมูล (Information Retrieval) และ การตอบคำถามด้วยฐานข้อมูลองค์ความรู้ (Knowledge Base)

ความแตกต่างที่สำคัญคือ การตอบคำถามด้วยการค้นหาข้อมูลจะเป็นการหาคำตอบจากข้อความในเอกสาร ส่วนการตอบคำถามด้วยฐานข้อมูลองค์ความรู้จะใช้องค์ความรู้ (Semantic) มาช่วยตอบ ตัวอย่างของฐานข้อมูลองค์ความรู้ที่สำคัญได้แก่ ฟรีเบส (Freebase) หรือ กราฟองค์ความรู้ (Knowledge Graph)

การอ่านทำความเข้าใจบทความ (Reading Comprehension) เป็นส่วนประกอบหนึ่งของงานวิจัยด้านการค้นหาข้อมูล โดยจะอยู่ในส่วนหลังจากที่ค้นหาบทความที่เกี่ยวข้องพบแล้ว ซึ่งคอมพิวเตอร์จะต้องอ่านทำความเข้าใจคำถามกับบทความ เพื่อจะเลือกคำตอบที่ถูกต้องจากบทความ ตัวอย่างการอ่านทำความเข้าใจบทความเพื่อตอบคำถามจะอยู่ในรูป 1.1 จะเห็นได้ว่าการตอบคำถามจากบทความ คำตอบสามารถที่จะเป็นคำเพียงคำเดียว (Factoid) หรือเป็นวลี (Phrase) ที่มาจากบทความก็ได้

ในช่วงเริ่มแรก การอ่านทำความเข้าใจบทความจะใช้กฎทางภาษามาช่วยหาคำตอบ [1] ต่อมา จึงเริ่มมีการใช้วิธีด้านการสกัดข้อมูล (Information Extraction) เพื่อสร้างเป็นกฎความสัมพันธ์ไว้ใช้ตอบคำถาม [2] ถึงกระนั้นวิธีที่กล่าวมาต่างก็มีข้อจำกัดในการใช้งาน คือระบบจะไม่สามารถทำงานได้ หากไม่ได้มีกฎมารองรับคำถามนั้นไว้ก่อน

การเรียนรู้ด้วยคอมพิวเตอร์ (Machine Learning) โดยเฉพาะการเรียนรู้เชิงลึก (Deep Learning) พึ่งจะถูกนำมาใช้อย่างจริงจังเมื่อปี 2015 พร้อมกับชุดข้อมูลขนาดใหญ่ [3-7] ทั้งนี้วิธีการที่

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ใช้กันเป็นหลักจะเป็นการรวมเอานิวรอลเน็ตเวิร์กแบบความจำระยะสั้นแบบยาวหลาย ๆ ตัว (Long-short term memory; LSTM) และกลไกความสนใจ (Attention Mechanism) เข้ามาสร้างเป็นระบบที่มีความซับซ้อนเพื่อใช้ในการตอบคำถามจากบทความ [3]

กลุ่มคำถามที่มีหลายความสัมพันธ์ (Multiple Relationships) คือ คำถามที่จะต้องใช้ประโยคอย่างน้อยสองประโยคมาเชื่อมต่อกันเพื่อหาคำตอบ เช่น จากรูป 1.1 คำถามข้อ 4 จำเป็นต้องใช้สองประโยคเพื่อหาคำตอบ คือ

1) ลินคอล์นถูกลอบสังหารโดยนักแสดงและผู้ฝักใฝ่สมาพันธรัฐ จอห์น วิลค์ส บูธ

2) การลอบสังหารลินคอล์นเป็นการลอบสังหารประธานาธิบดีสหรัฐอเมริกาครั้งแรก

คำถามกลุ่มมีหลายความสัมพันธ์นี้ยังไม่ได้มีงานวิจัยมากนักบนชุดข้อมูลขนาดใหญ่ แต่มีการวิจัยอยู่บ้างบนข้อมูลการตอบคำถามสังเคราะห์บาบิ (bAbI) ที่ถูกออกแบบมาทดสอบการใช้เหตุผลของโมเดลที่ใช้การเรียนรู้ด้วยคอมพิวเตอร์ [7]

วิธีการตอบคำถามกลุ่มมีหลายความสัมพันธ์ โมเดลที่ใช้จะต้องมีโครงสร้างซึ่งสามารถอ่านบทความได้หลายเที่ยว (Multiple hops) เพื่อจะสามารถหาจุดเชื่อมโยงระหว่างประโยคที่เกี่ยวข้องและต้องมีวิธีการสร้างองค์ความรู้ใหม่ จากความสัมพันธ์ระหว่างประโยค วิธีที่นำมาใช้ ได้แก่ การใช้นิวรอลเน็ตเวิร์กแบบหลายชั้นเพื่อสร้างองค์ความรู้ใหม่ (Neural Reasoner) [8], การติดตามสถานะของหลายเอนทิตี (Recurrent Entity Networks) [9] และการนำเอาเวกเตอร์บริบท (Context Vector) เข้ามาประกอบในการตอบคำถาม [10]

บทความ

ลินคอล์นถูกลอบสังหารโดยนักแสดงและผู้ฝักใฝ่สมาพันธรัฐ จอห์น วิลค์ส บูธ การลอบสังหารลินคอล์นเป็นการลอบสังหารประธานาธิบดีสหรัฐอเมริกาครั้งแรกและทำให้ทั้งประเทศโศกเศร้า นักวิชาการและสาธารณะจัดลินคอล์นเป็นหนึ่งในสามประธานาธิบดีสหรัฐอเมริกาที่ยิ่งใหญ่ที่สุดมาจนปัจจุบัน

คำถาม – คำตอบ

1) ใครเป็นคนฆ่าลินคอล์น? - จอห์น วิลค์ส บูธ
2) ลินคอร์นเคยทำอาชีพอะไร? – ประธานาธิบดี
3) ใครบ้างที่คิดว่าลินคอล์นเป็นประธานาธิบดีที่ยิ่งใหญ่ที่สุด? – นักวิชาการและสาธารณะ
4) ใครคือคนที่ลอบสังหารประธานาธิบดีสหรัฐเป็นคนแรก? - จอห์น วิลค์ส บูธ

รูป 1.1 ตัวอย่างการตอบคำถามจากบทความวิกิพีเดีย โดยคำตอบจะเป็นวลีที่มาจากบทความ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

งานวิจัยชิ้นนี้มีจุดมุ่งหมาย ที่จะทำการแก้ปัญหากลุ่มคำถามที่มีหลายความสัมพันธ์บนชุดข้อมูลขนาดใหญ่ด้วยวิธีการเรียนรู้เชิงลึก เพื่อเพิ่มความแม่นยำโดยรวมของโมเดล โดยได้เสนอวิธีการ 3 วิธีได้แก่ (1) การใช้เวกเตอร์คำอ้างอิง (Coreference Vector) (2) การตอบคำถามแบบสองทิศทาง (3) ฟังก์ชันต้นทุนจากความยาวของคำตอบ (Answer Length Loss Function)

1.2 วัตถุประสงค์ของงานวิจัย

เพื่อพัฒนาระบบการตอบคำถามจากบทความที่มีความแม่นยำโดยใช้วิธีการเรียนรู้เชิงลึกและได้ความแม่นยำที่สูงกว่าระบบการตอบคำถามด้วยวิธีการเรียนรู้แบบลึกมาตรฐาน

1.3 ขอบเขตของงานวิจัย

1.3.1 ชุดข้อมูล

ข้อมูลตัวอย่างที่จะนำมาใช้ทดสอบการตอบคำถามจากบทความในการวิจัยนี้คือ

1) ชุดข้อมูลคำถามสควอด (Stanford Question Answering Dataset ; SQuAD) [6] ประกอบด้วยข้อมูลวิกิพีเดีย 536 เรื่อง ที่ดึงมาจากหลายกลุ่มหัวข้อ โดยจะให้คนเป็นผู้ตั้งคำถามและให้คำตอบมาจากวลีภายในบทความ โดยจำนวนคำถามคำตอบทั้งหมดมีด้วยกันประมาณ 100,000 ชุด

2) ชุดข้อมูลคำถามสัพเพเหระ (TriviaQA) [5] ประกอบด้วยข้อมูลบทความจากวิกิพีเดียและอินเตอร์เน็ต โดยคำถามกับคำตอบมาจากเว็บไซต์ทดสอบความรู้รอบตัว บทความได้จากการค้นหาบนวิกิพีเดียและเว็บไซต์ต่าง ๆ ด้วยคำถามกับคำตอบ จำนวนคำถามคำตอบและจำนวนบทความที่ใช้เพื่อตอบคำถามจะมีด้วยกันประมาณ 650,000 ชุด

ในภาพรวมคำถามจากชุดข้อมูลคำถามสัพเพเหระจะมีความยากกว่าสควอด ถ้าดูจากผลการทดลองที่ให้มนุษย์เป็นคนตอบคำถาม (Gold Standard) จะได้ผลเอฟวัน (F1) เป็น 86.8% และ 79.7% บนชุดข้อมูลคำถามสควอดและชุดข้อมูลคำถามสัพเพเหระตามลำดับ เนื่องมาจากชุดข้อมูลคำถามสัพเพเหระมีปริมาณสัดส่วนของคำถามที่มีหลายความสัมพันธ์มากกว่าชุดข้อมูลคำถามสควอดประมาณ 3 เท่าตัว โดยมีค่า 40% และ 14% ตามลำดับ ดังแสดงในตารางที่ 1.1

1.3.2 การวิเคราะห์ชุดข้อมูล

ในงานวิจัยจะมีการวิเคราะห์ชุดข้อมูลที่นำมาใช้ ในแง่ของลักษณะของคำถาม เช่น ความยาวของคำถาม ความยาวของคำตอบ ประเภทของคำถาม (ใคร ที่ไหน เมื่อไหร่ อย่างไร) นอกจากนี้จะมีการวิเคราะห์ผลการทดลองของคำถามที่มีหลายความสัมพันธ์ โดยทำการเลือกคำถามประเภทดังกล่าวออกมาเป็นจำนวน 100 คำถาม

ตารางที่ 1.1 สัดส่วนประเภทคำถามโดยแบ่งตามวิธีการหาคำตอบใน ชุดข้อมูลสควอด และชุดข้อมูลสัพเพเหระ (คำถามบางข้อสามารถเป็นได้หลายประเภท ผลรวมจึงเกิน 100% ได้)

ประเภทคำถาม	ชุดข้อมูลสควอด	ชุดข้อมูลสัพเพเหระ
คำถามที่วลีในคำถามตรงกับคำตอบบางส่วน หรือมีคำที่ใช้แทนกันได้ (Synonym)	33%	40%
คำถามที่ต้องใช้ความรู้รอบตัว (World Knowledge)	9%	17%
คำถามที่คำตอบเป็นการถอดความของกันและกัน (Paraphrase)	64%	67%
คำถามที่ต้องใช้ความสัมพันธ์ของหลายประโยคมาตอบ (Multiple Relationships)	14%	40%

1.3.3 การประเมินผลแยกเป็นส่วนประกอบ

ในงานวิจัยจะมีการประเมินประสิทธิภาพของแต่ละส่วนประกอบที่นำเสนอ โดยทำการตัดส่วนประกอบนั้น ๆ ออก (Ablation Test) แล้วนำผลการทดลองที่ได้ มาเปรียบเทียบกับโมเดลที่สมบูรณ์

1.3.4 การวัดค่านัยสำคัญทางสถิติ (Statistical Significance)

ในการเปรียบเทียบผลการทดลอง จะมีการทดสอบเพื่อดูว่าส่วนประกอบนั้น ๆ สามารถเพิ่มประสิทธิภาพให้กับระบบอย่างมีนัยสำคัญหรือไม่

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

1.4 ประโยชน์ที่ได้รับ

ได้วิธีการตอบคำถามจากบทความในกลุ่มคำถามที่ต้องทำการเชื่อมความสัมพันธ์ระหว่างประโยคในบทความ ที่มีความแม่นยำกว่าวิธีการอ่านแบบหลายครั้งที่มีอยู่ ในขณะที่ยังคงทำงานได้ดีบนคำถามกลุ่มอื่น

1.5 วิธีดำเนินการวิจัย

วิธีการดำเนินการวิจัย สามารถแบ่งออกได้เป็นขั้นตอน ดังแสดงในตารางที่ 1.2

1. ศึกษาวรรณกรรมและงานวิจัยที่เกี่ยวข้องกับหัวข้อที่จะวิจัย
2. ศึกษาข้อมูลจากชุดข้อมูลคำถามสควอดและชุดข้อมูลคำถามสัพเพเหระ โดยพิจารณาถึงรูปแบบของคำถาม ประเภทของคำถาม และวิธีการไปสู่คำตอบจากบทความ
3. ทดสอบแนวทางวิจัยเบื้องต้นปรับปรุงและประยุกต์เพื่อให้ได้ระบบที่มีประสิทธิภาพที่ดีขึ้น

4. วิเคราะห์ผลการทดลองเพื่อหาข้อดีและข้อเสีย สำหับทำการสรุปผลการทดลอง
5. นำผลการสรุปมาปรับปรุงแก้ไขระบบ
6. วิเคราะห์และสรุปผลการทดลองสุดท้าย
7. จัดทำวิทยานิพนธ์

1.6 ผลงานวิจัยที่ตีพิมพ์

"End-to-End Memory Network for QA with Multiple Relationships" โดย ณัฐชัย ตรีทศายุธ พีรพล เวทีกูล และปรัชญา บุญขวัญ ในงานประชุมวิชาการ "The 14th International Joint Conference on Computer Science and Software Engineering (JCSSE)" ซึ่งจัดขึ้น ณ โรงแรมทวินโลตัส จังหวัดนครศรีธรรมราช ประเทศไทย ระหว่างวันที่ 12 –14 กรกฎาคม 2560

"Enhance Machine Reading Comprehension on Multiple Sentence Questions with Gated and Dense Coreference Information" โดย ณัฐชัย ตรีทศายุธ พีรพล เวทีกูล และปรัชญา บุญขวัญ ในงานประชุมวิชาการ "The 15th International Joint Conference on Computer Science and Software Engineering (JCSSE)" ซึ่งจัดขึ้น ณ คณะเทคโนโลยีสารสนเทศและการสื่อสาร มหาวิทยาลัยมหิดล (ศาลายา) ระหว่างวันที่ 11 –13 กรกฎาคม 2561

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 1.2 แผนการดำเนินงาน

การดำเนินงาน	2560						2561											
	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	มิ.ย.	ก.ค.
ศึกษางานวรรณกรรมและงานวิจัยที่เกี่ยวข้องกับหัวข้อที่จะวิจัย												
ศึกษาข้อมูลจากผู้เชี่ยวชาญด้านภาษาไทย												
ทดสอบแนวทางวิจัยเบื้องต้น												
วิเคราะห์ผลการทดลอง												
นำผลการสรุปมาปรับปรุงแก้ไขระบบ												
วิเคราะห์และสรุปผลการทดลองสุดท้าย												
จัดทำเปเปอร์เพื่อส่งงานประชุม												
จัดทำวิทยานิพนธ์												
สอบวิทยานิพนธ์												

บทที่ 2
แนวคิดและทฤษฎี

ทฤษฎีที่เกี่ยวข้องกับงานวิจัยชิ้นนี้แบ่งออกได้เป็น 7 หัวข้อได้แก่ คำถามที่มีหลายความสัมพันธ์ การแทนข้อความ นิวรอลเน็ตเวิร์ก นิวรอลเน็ตเวิร์กคอนโวลูชัน นิวรอลเน็ตเวิร์กแบบวกกลับ กลไกความสนใจ และการคำนวณความแม่นยำ

2.1 คำถามที่มีหลายความสัมพันธ์ (Question with Multiple Relationships)

คำถามที่มีหลายความสัมพันธ์คือคำถามที่จำเป็นจะต้องเชื่อมต่อคำในเอกสารเข้าด้วยกันเพื่อหาคำตอบ ทั้งนี้การเชื่อมต่อคำอาจทำได้ในหลายระดับ เช่น ระหว่างประโยค หรือระหว่างวลีในประโยคที่มีความยาวมาก ทั้งนี้เราอาจมองได้ว่าคำถามในกลุ่มนี้จะเกิดในกรณีที่ระยะห่างระหว่างคำสำคัญกับคำตอบอยู่ห่างกันมาก

ความหมายของคำสำคัญคือคำที่ปรากฏในคำถามและบทความที่จำเป็นต้องใช้ในการหาคำตอบ โดยมากแล้วคำสำคัญจะเป็นชื่อเอนทิตี (Name Entity) แต่ก็อาจเป็นได้ทั้งกริยาหรือคำนามขึ้นอยู่กับลักษณะของคำถามนั้น ๆ

จากรูป 2.1 คำสำคัญคือ “V&A Theatre & Performance galleries” เนื่องจากปรากฏอยู่ทั้งในคำถามและบทความ เพื่อที่จะหาคำตอบคำ ๆ นี้จะต้องมีการเชื่อมโยงไปหา “They” ซึ่งอยู่กันคนละประโยค นอกจากนี้สังเกตเห็นได้ว่าระยะห่างระหว่างคำสำคัญและคำตอบจะอยู่ห่างกันมาก

บทความ : **The V&A Theatre & Performance galleries** opened in March 2009. ... **They** hold the UK’s biggest national collection of material about live performance
คำถาม : What collection does the **V&A Theatre & Performance galleries** hold?
คำตอบ : material about live performance

รูป 2.1 ตัวอย่างคำถามที่มีหลายความสัมพันธ์

2.2 การแทนข้อความ (Text Representation)

ในการจำแนกประเภทของข้อความนั้น สิ่งหนึ่งที่ต้องทำคือการแทนที่ข้อความด้วยฟีเจอร์เพื่อจะนำไปสู่กระบวนถัดไป วิธีการในการแทนข้อความนั้นมีดังต่อไปนี้

2.2.1 ถุงคำ (Bag-of-words; BoW)

เป็นการแทนข้อความในรูปแบบของเวกเตอร์ซึ่งมีขนาดเท่ากับจำนวนคำทั้งหมดในพจนานุกรมของชุดข้อมูลชุดนั้น โดยไม่มีการใช้ไวยากรณ์และลำดับของคำ ตัวอย่างเช่น หากมีประโยค ฉันไปโรงเรียนแล้วไปห้องสมุด ซึ่งหากมีเป็นพจนานุกรมของคำได้เป็น [ฉัน, ไป, โรงเรียน, กลับ, รถยนต์, แล้ว, ห้องสมุด] ซึ่งเมื่อใช้การแทนด้วยถุงคำ จะสามารถสร้างเวกเตอร์ของประโยคได้เป็น

ฉันไปโรงเรียนแล้วไปห้องสมุด = [1, 2 ,1 ,0 ,0 ,1 ,1]

2.2.2 ทีเอฟไอดีเอฟ (Term Frequency-Inverse Document Frequency; tfidf)

การแทนข้อความด้วยทีเอฟไอดีเอฟเป็นวิธีการใช้ถุงคำ (Bag-of-Word) อย่างหนึ่ง แต่มีการแทนค่าในแต่ละช่องของเวกเตอร์ด้วยความถี่ของคำในข้อความคูณด้วยค่าผกผันของความถี่ของคำนั้น ๆ จากทั้งชุดข้อมูล สามารถคำนวณทีเอฟไอดีเอฟได้ดังสมการต่อไปนี้

$$tfidf = tf \times idf \tag{2.1}$$

$$idf = log(\frac{N}{n_t}) \tag{2.2}$$

tf คือ ความถี่ของคำที่สนใจ

N คือ จำนวนของบทความทั้งหมดในชุดข้อมูล

n_t คือ จำนวนของบทความในชุดข้อมูลที่มีคำที่สนใจ

2.2.3 เวกเตอร์วันฮอต (One-hot Vector)

เวกเตอร์แบบวันฮอตคือการสร้างเวกเตอร์ที่มีขนาดเท่ากับจำนวนของคำศัพท์ที่เคยปรากฏขึ้นทั้งหมดในชุดข้อมูลที่นำมาใช้สอน โดยวันฮอตเวกเตอร์จะแทนค่าของคำเหล่านั้นด้วยเวกเตอร์ที่มีค่า 1 เพียงช่องเดียว นอกนั้นจะมีค่าเป็น 0 โดยแต่ละคำจะมีวันฮอตเวกเตอร์แตกต่างกัน ดังเช่นตัวอย่าง

[ฉัน, ไป, โรงเรียน, แล้ว, ไป, ห้องสมุด] แทนด้วย $\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

2.2.4 คำฝังตัว (Word Embedding)

คำฝังตัว [11] เป็นการแทนที่ข้อความด้วยกลุ่มของเวกเตอร์ เช่น การใช้เวกเตอร์ของคำ (Word vector) ในการแทนที่ข้อความ จะใช้จำนวนของเวกเตอร์เท่ากับความยาวของข้อความ และขนาดของเวกเตอร์จะสามารถกำหนดเองได้ ทั้งนี้การสร้างเวกเตอร์คำจะวิเคราะห์ข้อความจากชุด

ข้อมูลทั้งหมดก่อน แล้วจึงสร้างเวกเตอร์คำโดยให้คู่ของคำที่มีความหมายใกล้เคียงกันคือมีระยะห่างของเวกเตอร์คำใกล้เคียงกัน ตัวอย่างของเวกเตอร์ที่ได้จากการฝังคำเป็นดังนี้

[ฉัน, ไป, โรงเรียน, แล้ว, ไป, ห้องสมุด]

แทนด้วย $\begin{bmatrix} 0.56 & 0.01 & 0.97 & 0.08 & 0.01 & 0.57 \\ 0.21 & 0.07 & 0.62 & 0.17 & 0.07 & 0.58 \\ 0.79 & 0.14 & 0.17 & 0.26 & 0.14 & 0.36 \end{bmatrix}$

โกลฟ (Glove) [12] คือชุดข้อมูลของคำฝังตัว ที่ได้ผ่านการเรียนรู้เพื่อหาเวกเตอร์ของคำมาแล้ว (Pretrained word vector) โดยกูเกิลจากบทความในวิกิพีเดียและอินเตอร์เน็ต ข้อดีของโกลฟคือช่วยลดเวลาในการสอนของระบบเพราะสามารถที่จะนำเอาโกลฟไปใช้ได้เลย แทนที่จะต้องทำการสอนเองใหม่ทั้งหมด นอกจากนี้โกลฟยังช่วยแก้ปัญหาคำที่ไม่เคยพบในชุดข้อมูลสอนได้

อักขระฝังตัว (Character Embedding) เป็นการสร้างเวกเตอร์ของคำที่ได้จากการประมวลผลกลุ่มของตัวอักษร โดยขั้นแรกจะต้องทำการฝังตัวอักษรให้เป็นเวกเตอร์ ดังนั้นในคำหนึ่งคำจึงประกอบด้วยจำนวนเวกเตอร์เท่ากับจำนวนตัวอักษรในคำ

การสร้างเวกเตอร์ของคำจากกลุ่มของเวกเตอร์ตัวอักษร มีอยู่หลัก ๆ ด้วยกันสองแบบ คือ 1) นำเอากลุ่มเวกเตอร์ของตัวอักษรไปเข้าเน็ตเวิร์กแบบวกกลับ แล้วใช้เวกเตอร์ของตัวอักษรสุดท้ายของคำเป็นเวกเตอร์แทนคำ 2) การใช้เน็ตเวิร์กคอนโวลูชันแนล (Convolutional Neural Network) ด้วยการรวม (Pooling) ที่อ้างอิงเวลา [13]

2.3 นิวรอลเน็ตเวิร์ก (Neural Network)

นิวรอลเน็ตเวิร์กเป็นแบบจำลองที่มีแรงบันดาลใจมาจากการทำงานของสมองมนุษย์ โดยสามารถเปรียบเทียบการทำหน้าที่ได้คือการเรียนรู้จากข้อมูลที่มีอยู่แล้ว เพื่อใช้ทำนายข้อมูลในลักษณะเดียวกัน

Chulalongkorn University

2.3.1 นิวรอลเน็ตเวิร์กแบบป้อนไปข้างหน้า (Feed Forward Neural Network)

นิวรอลเน็ตเวิร์กประเภทนี้จะมีลำดับส่งผ่านข้อมูลในทิศทางเดียว โดยโครงสร้างจะแบ่งเป็นลำดับชั้น ในแต่ละลำดับชั้นเดียวกันจะมีเพอร์เซปตรอนที่ไม่มีการเชื่อมต่อกัน แต่จะมีเส้นที่เชื่อมต่อกันระหว่างเพอร์เซปตรอนที่อยู่ในชั้นซึ่งอยู่ติดกัน โดยข้อมูลที่ส่งออกจากชั้นก่อนหน้าจะกลายเป็นข้อมูลขาเข้าของชั้นปัจจุบัน ดังตัวอย่างในรูป 2.2 โดยสามารถคำนวนหาค่าของผลลัพธ์ในชั้นถัดไปได้จากสมการดังต่อไปนี้

$$z_j^l = \sum_{k=1}^{n} w_{jk}^l a_k^{l-1} + b_j^l \quad (2.3)$$

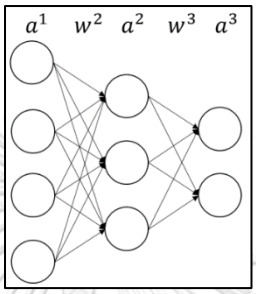
$$a_j^l = g(z_j^l) \quad (2.4)$$

a_k^{l-1} แทนผลลัพธ์ของเพอร์เซปตรอนตัวที่ k ในลำดับชั้น $l-1$

w_{jk}^l แทนน้ำหนักสำหรับเพอร์เซปตรอนตัวที่ j ในลำดับชั้น l ที่มีเส้นเชื่อมมาจากเพอร์เซปตรอนตัวที่ k ในลำดับชั้นก่อนหน้า

b_j^l แทนค่าไบแอส

นอกจากนี้ g เป็นสัญลักษณ์แทนฟังก์ชันกระตุ้น และให้ n แทนจำนวนเพอร์เซปตรอนในลำดับชั้นที่ $l-1$ จะสามารถเขียนสมการของการคำนวณ a_j^l ได้เป็น



รูป 2.2 แสดงตัวอย่างโครงสร้างนิวนอลเน็ตเวิร์กแบบป้อนไปข้างหน้า

2.3.1.1 ฟังก์ชันกระตุ้น (Activation Function)

ข้อมูลที่ส่งออกจากเพอร์เซปตรอนจะถูกส่งผ่านไปยังฟังก์ชันกระตุ้น ที่มีลักษณะเป็นแบบไม่ใช่ฟังก์ชันเชิงเส้น (Non-linear) เพื่อให้นิวรอลเน็ตเวิร์กทำงานที่ซับซ้อนมากขึ้นได้ รูปแบบของฟังก์ชันกระตุ้นที่นิยมใช้กันมีดังต่อไปนี้

1) ฟังก์ชันซิกมอยด์ (Sigmoid Function)

ทำการเปลี่ยนค่าที่นำเข้ามาให้อยู่ในช่วง 0 ถึง 1 โดยมีสมการดังนี้คือ

$$\sigma(z) = \frac{1}{1+e^{-z}} \tag{2.5}$$

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

2) ฟังก์ชันแทนเจนต์ไฮเพอร์โบลิก (Hyperbolic Tangent Function หรือ tanh)

ทำการเปลี่ยนค่าที่นำเข้ามาให้อยู่ในช่วง -1 ถึง 1 โดยมีสมการดังนี้คือ

$$tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{2.6}$$

3) ฟังก์ชันเรกทิไฟด์เชิงเส้น (Rectified Linear Unit Function)

ทำการเปลี่ยนค่านำเข้าที่ติดลบให้เป็น 0 ถ้าค่าที่เข้ามามากกว่าเท่ากับ 0 จะได้ผลเป็นค่าเดิมที่เข้ามา มีสมการดังนี้คือ

$$f(z) = \begin{cases} 0 & if\ z < 0 \\ z & if\ z \geq 0 \end{cases} \tag{2.7}$$

4) ฟังก์ชันค่าสูงสุดอย่างอ่อน (Softmax Function)

ฟังก์ชันค่าสูงสุดอย่างอ่อนหรือที่จะเรียกต่อไปว่า Softmax จะทำการเปลี่ยนค่าที่เข้ามาให้ผลลัพธ์อยู่ในช่วง 0 ถึง 1 ซึ่งเป็นค่าที่แสดงถึงความน่าจะเป็นของค่าที่นำเข้ามาแต่ละตัว โดยผลรวมของค่าความน่าจะเป็นที่ได้จะมีผลรวมเป็น 1 มีสมการดังต่อไปนี้

$$f(z)_j = \frac{e^{z_j}}{\sum_{i=1}^{K} e^{z_i}} \tag{2.8}$$

2.3.1.2 ฟังก์ชันต้นทุน (Cost Function หรือ Objective Function)

ในการเรียนรู้ของนิวรอลเน็ตเวิร์กนั้น จำเป็นจะต้องมีฟังก์ชันที่สามารถใช้วัดผลการเรียนรู้หรือก็คือฟังก์ชันต้นทุน โดยเป้าหมายของการเรียนรู้จะกำหนดให้เป็นการพยายามลดค่าที่ได้จากฟังก์ชันต้นทุนให้เข้าใกล้ 0 ให้มากที่สุด

สัญลักษณ์ของสมการในแต่ละข้อจะใช้ J แทนฟังก์ชันต้นทุน n คือจำนวนข้อมูลทั้งหมดที่ใช้ในการเรียนรู้ y_i แทนผลลัพธ์จริงที่ต้องการของข้อมูลชุดที่ i และ $\hat{y}_i$ แทนผลลัพธ์ที่ทำนายได้ของข้อมูลชุดที่ i ฟังก์ชันต้นทุนที่นิยมใช้กันมีดังต่อไปนี้คือ

1) ค่าเฉลี่ยความผิดพลาดกำลังสอง (Mean Squared Error หรือ MSE)

$$J = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \tag{2.9}$$

2) ค่าเฉลี่ยครอสเอนโทรปีแบบทวิภาค (Binary Cross-entropy)

$$J = -\frac{1}{n}\sum_{i=1}^{n} y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i) \tag{2.10}$$

3) ค่าติดลบลอการิธึมของความเป็นไปได้ (Negative Log Likelihood)

เป็นฟังก์ชันตรงข้ามกับการหาค่าที่มากที่สุดของลอการิธึมของความเป็นไปได้ (Maximum Likelihood) กำหนดให้ ค่า L แทน จำนวนคลาสทั้งหมดที่เป็นไปได้ และ a_y^L แทนผลต่างของความน่าจะเป็นของคลาสทั้งหมดที่ทำนายและคลาสที่ถูกต้อง

$$J = -\frac{1}{n}\sum_{i=1}^{n}(\ln a_y^L) \tag{2.11}$$

2.3.1.3 การหาค่าที่เหมาะสมที่สุด (Optimization)

การหาค่าที่เหมาะสมที่สุดเป็นวิธีการปรับปรุงอัตราการเรียนรู้ เพื่อให้สามารถลดค่าจากฟังก์ชันต้นทุนได้มากที่สุดในแต่ละรอบ เพื่อเพิ่มโอกาสไปยังจุดต่ำสุดทั้งหมด (Global Minima) สำหรับวิธีการที่นิยมกันมีดังต่อไปนี้

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

1) สโตแคสติกเกรเดียนเดสเซนท์ (Stochastic Gradient Descent หรือ SGD)

เมื่อกำหนด w แทนค่าพารามิเตอร์ซึ่งเป็นน้ำหนักที่ต้องการจะปรับค่า อัลฟา คือ อัตราการเรียนรู้ และเกรเดียนของฟังก์ชันต้นทุนเทียบกับ w การเรียนรู้แบบสโตแคสติกเกรเดียนเดสเซน จะทำการปรับค่า w ด้วยสมการดังนี้

$$w_t = w_{t-1} - \alpha \frac{\partial J_t}{\partial w} \tag{2.12}$$

โมเมนตัมเป็นอีกหลักการหนึ่งซึ่งมีการนำเอามาใช้ โดยมีจุดประสงค์เพื่อทำให้การเรียนรู้มีการลู่เข้าที่เร็วขึ้น และหลีกเลี่ยงการตกไปในจุดต่ำสุดโลคัล (Local Optima) โดยกำหนดให้ v แทนค่าความเร็วซึ่งจะถูกปรับไปพร้อมกับ w และค่าสัมประสิทธิ์ของโมเมนตัม γ สามารถเขียนได้ในรูปสมการดังนี้

$$v_t = \gamma v_{t-1} + \alpha \frac{\partial J_t}{\partial w} \tag{2.13}$$

$$w_t = w_{t-1} - v_t \tag{2.14}$$

2) วิธีเกรเดียนที่ปรับตัวได้ (Adaptive Gradient Method หรือ AdaGrad)

วิธีการนี้ได้ทำการพัฒนาให้การปรับการเรียนรู้สำหรับแต่ละเพอร์เซปตรอนมีค่าที่แตกต่างกันได้ เพื่อทำให้การลู่เข้ามีประสิทธิภาพมากยิ่งขึ้น โดยดูจากค่าเกรเดียนในอดีต กำหนดให้ g คือเกรเดียนในอดีตจะได้เป็นสมการดังนี้

$$g_t = \frac{\partial J_t}{\partial w} \tag{2.15}$$

$$w_t = w_{t-1} - \frac{\alpha}{\sqrt{\sum_{k=1}^{t} g_i{}^2}} g_t \tag{2.16}$$

จุฬาลงกรณ์มหาวิทยาลัย

ข้อดีของการใช้วิธีการปรับเกรเดียนแบบนี้คือทำให้ไม่จำเป็นต้องมีการปรับค่าการเรียนรู้ระหว่างการสอนเอง เนื่องจากวิธีการนี้จะทำการหาค่าที่เหมาะสมให้โดยอัตโนมัติ

2.3.2 การแพร่กระจายย้อนกลับ (Backpropagation)

เนื่องจากเกรเดียนที่ได้จากการหาค่าความผิดพลาดสุดท้าย จากฟังก์ชันต้นทุนนั้นมีไว้ให้สำหรับลำดับชั้นสุดท้ายในเน็ตเวิร์กเท่านั้น ดังนั้นหากจะต้องการทำการหาค่าเกรเดียนสำหรับปรับค่าของ w ของเพอร์เซปตรอนในลำดับชั้นก่อนหน้า จะต้องนำเอาการแพร่กระจายย้อนกลับมาใช้

จะสามารถเขียนสมการของค่าความผิดพลาดได้เป็น

$$\delta_j^l = \frac{\partial J}{\partial z_j^l} = \frac{\partial J}{\partial a_j^l}\frac{\partial a_j^l}{\partial z_j^l} = \frac{\partial J}{\partial a_j^l} g'(z_j^l) \tag{2.17}$$

δ_j^l แทนค่าความผิดพลาดของเพอร์เซปตรอนตัวที่ j ในลำดับชั้น l

j แทนฟังก์ชันต้นทุน

z เป็นค่าที่คำนวณได้ก่อนจะผ่านฟังก์ชันกระตุ้น g

สำหรับการหาค่า $\frac{\partial J}{\partial a_j^l}$ นั้น ในลำดับชั้นสุดท้ายสามารถคำนวณหาได้โดยตรง จากฟังก์ชันต้นทุนที่เลือกใช้ ส่วนในลำดับชั้นก่อนหน้าจะต้องหาโดยวิธีการแพร่กระจายย้อนกลับ โดยจะทำคล้ายกับการป้อนไปข้างหน้าเพียงแต่กลับทิศกันเท่านั้น กำหนด m คือจำนวนเพอร์เซปตรอนในลำดับชั้นที่ $l+1$ โดยคำนวณได้จากสมการดังนี้

$$\frac{\partial J}{\partial a_j^l} = \sum_{k=1}^{m} \frac{\partial J}{\partial z_k^{l+1}} \frac{\partial z_k^{l+1}}{\partial a_j^l} = \sum_{k=1}^{m} \delta_k^{l+1} w_{kj}^{l+1} \tag{2.18}$$

จากนั้น เมื่อคำนวณค่าความผิดพลาดของแต่ละระดับชั้นได้ ก็สามารถหาค่าผิดพลาดเทียบกับน้ำหนักและค่าไบแอสใด ๆ ได้จาก

$$\frac{\partial J}{\partial w_{jk}^l} = \frac{\partial J}{\partial z_j^l} \frac{\partial z_j^l}{\partial w_{jk}^l} = \delta_j^l a_k^{l-1} \tag{2.19}$$

$$\frac{\partial J}{\partial b_j^l} = \frac{\partial J}{\partial z_j^l} \frac{\partial z_j^l}{\partial b_j^l} = \delta_j^l \tag{2.20}$$

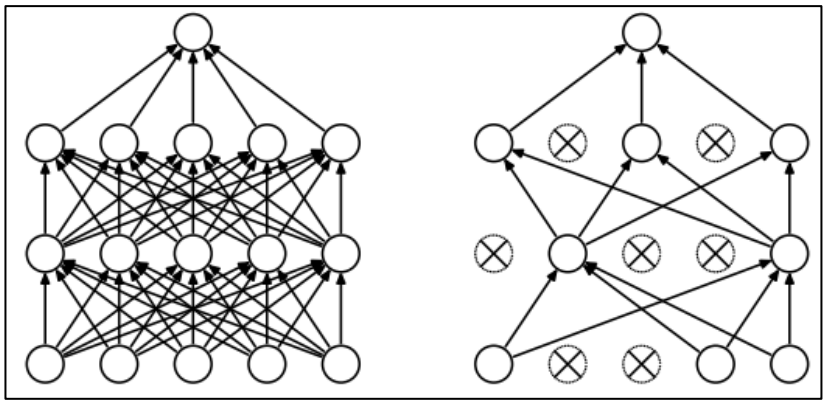
ในกรณีที่ใช้สโตแคสติกเกรเดียนเดสเซนท์การปรับปรุงค่าน้ำหนัก w_{jk}^l จะทำได้โดย

$$w_{jk,t}^l = w_{jk,t-1}^l - \alpha a_{k,t}^{l-1} \delta_{j,t}^l \tag{2.21}$$

2.3.3 การดรอปเอาท์ (Dropout)

การดรอปเอาท์ [14] เป็นวิธีป้องกันไม่ให้ตัวโมเดลยึดติดกับข้อมูลที่ใช้นำมาเรียนรู้มากเกินไป (overfitting) โดยใช้วิธีการสุ่มตัดเส้นเชื่อมของเน็ตเวิร์กในระหว่างการเรียนรู้ การสุ่มจะทำขึ้นใหม่ในทุกรอบของการเรียนรู้ของข้อมูลแต่ละรายการ ส่วนในระหว่างการทดสอบจะไม่ใช้การดรอปเอาท์ แต่จะใช้การเฉลี่ย รูปที่ 2.3 แสดงการทำดรอปเอาท์ โดยรูปขวาแสดงเส้นเชื่อมที่เหลืออยู่

การดรอปเอาท์ในเน็ตเวิร์กแบบวกกลับ (RNN) สามารถทำได้โดยการสุ่มตัดเส้นเชื่อมเช่นเดียวกับวิธีปกติ ข้อแตกต่างจะอยู่ที่ เส้นเชื่อมที่ถูกตัดจะเป็นเส้นเดียวกันตลอดลำดับของข้อมูลนั้น ๆ (timestep) เพื่อที่จะช่วยลดความผิดพลาดที่อาจเกิดจากการสุ่มตัดข้อมูลมากเกินไประหว่างช่วงเวลา [15]



รูป 2.3 รูปซ้ายแสดงนิวรอลเน็ตเวิร์กแบบปกติ รูปขวาแสดงการดรอปเอาท์ [14]

2.3.4 หน่วยรวมความหมาย (Semantic Fusing Unit; SFU)

Minghao Hu และคณะ [16] เสนอชั้นของเน็ตเวิร์กที่ทำงานคล้ายกับประตูสัญญาณในGRU (อธิบายไว้ใน2.5.2) เพื่อใช้ในการรวมเวกเตอร์เข้าด้วยกัน โครงสร้างของหน่วยรวมความหมายจะมีข้อมูลขาเข้าเป็นเวกเตอร์สองตัว ในที่นี้จะเรียกว่า เวกเตอร์ตั้งต้น ($\boldsymbol{r}$) และกลุ่มของเวกเตอร์เสริม ($\boldsymbol{f}$)

สมการ 2.22 ทำหน้าที่ในการบีบรวมเวกเตอร์ตั้งต้นเข้ากับเวกเตอร์เสริมให้มีขนาดเท่ากับ $\boldsymbol{r}$ โดยจะมีการคิดค่าของประตูสัญญาณที่จะใช้อัพเดท $\boldsymbol{g}$ เพื่อสร้างเป็น เวกเตอร์ของผลลัพธ์ $\boldsymbol{o}$ ทั้งนี้ W_r และ W_g มีหน้าที่ในการบีบเวกเตอร์ที่เข้ามาให้เท่ากับเวกเตอร์ตั้งต้น

$$\hat{r} = tanh(W_r([r; f_1; ..; f_n]) + b_r) \quad (2.22)$$

$$g = \sigma\big(W_g([r; f_1; ..; f_n]) + b_g\big) \quad (2.23)$$

$$o = g \circ \hat{r} + (1 - g) \circ r \quad (2.24)$$

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2.4 นิวรอลเน็ตเวิร์กคอนโวลูชัน (Convolutional Neural Network)

นิวรอลเน็ตเวิร์กคอนโวลูชันเป็นนิวรอลเน็ตเวิร์กเชิงลึกรูปแบบหนึ่ง มีจุดเริ่มต้นมาจากงานวิจัยทางด้านภาพตัวอักษร โดยมักจะใช้ข้อมูลรับเข้าเป็นเมทริกซ์ โครงสร้างของนิวรอลเน็ตเวิร์กคอนโวลูชันจะเกิดจากการนำชั้นของนิวรอลเน็ตเวิร์กหลายประเภทมาประกอบเข้าด้วยกัน ดังต่อไปนี้

2.4.1 ชั้นคอนโวลูชัน (Convolutional Layer)

เป็นชั้นที่ทำการหาฟีเจอร์จากกลุ่มของข้อมูลรับเข้าที่อยู่ใกล้ ๆ กัน โดยใช้ผลคูณเชิงสเกลาร์ (dot product) เมทริกซ์กับตัวกรอง (filter) โดยน้ำหนักของตัวกรองนั้น จะเป็นน้ำหนักที่มีการใช้ร่วมกันในทุก ๆ การทำคอนโวลูชันของข้อมูลรับเข้า กำหนดให้ข้อมูลรับเข้าแทนด้วยเมทริกซ์

a^{l-1} ขนาด $N \times N$ และมีตัวกรองที่มีน้ำหนัก w ขนาด $m \times m$ ผลลัพธ์ a^l ของการทำคอนโวลูชันสามารถคำนวณได้ดังสมการ 2.22 และสมการ 2.23

$$z_{ij}^{l} = \sum_{a=0}^{m-1} \sum_{b=0}^{m-1} w_{a,b}^{l} a_{i+a,j+b}^{l-1} + b^{l} \quad (2.25)$$

$$a_{ij}^{l} = g(z_{ij}^{l}) \quad (2.26)$$

ในชั้นคอนโวลูชัน มีองค์ประกอบที่ต้องคำนึงถึงดังต่อไปนี้

2.4.1.1 ขนาดของตัวกรอง (Filter Size)

คือความกว้างและความสูงของตัวกรองที่จะนำมาใช้ในการทำคอนโวลูชัน

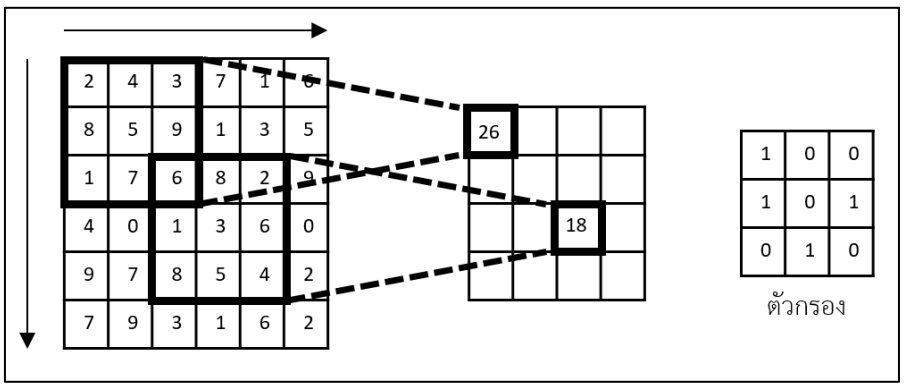
2.4.1.2 ชนิดของการทำคอนโวลูชัน (Convolution Type)

1) คอนโวลูชันแบบแคบ (Narrow Convolution)

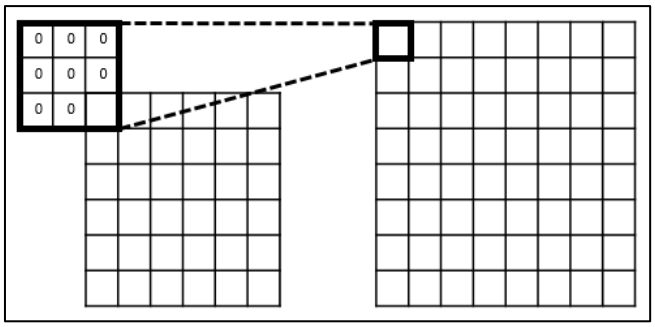
การทำคอนโวลูชันโดยทั่วไปมักจะเป็นการทำคอนโวลูชันแบบแคบ กล่าวคือ ตัวกรองที่นำไปทำผลคูณเชิงสเกลาร์กับเมทริกซ์นั้นจะไม่มีการกระทำเลยขอบของเมทริกซ์รับเข้า ส่งผลให้ผลลัพธ์ของการทำคอนโวลูชันที่มีข้อมูลรับเข้าขนาด $N \times N$ กับตัวกรองขนาด $m \times m$ จะได้เมทริกซ์ขนาด $(N - m + 1) \times (N - m + 1)$

2) คอนโวลูชันแบบกว้าง (Wide Convolution)

เป็นการทำคอนโวลูชันที่มีการกระทำเลยขอบของเมทริกซ์รับเข้าออกไป โดยพื้นที่ที่เกินออกไปนั้น จะมีการแทนค่าของข้อมูลช่องนั้น ๆ ด้วย 0 เรียกว่าการเสริมเติม (padding) ผลลัพธ์ของการทำคอนโวลูชันแบบกว้างที่มีข้อมูลรับเข้าขนาด $N \times N$ กับตัวกรองขนาด $m \times m$ จะได้เมทริกซ์ขนาด $(N + m - 1) \times (N + m - 1)$ ทั้งนี้การทำคอนโวลูชันแบบกว้างนี้มีขึ้นเพื่อป้องกันการสูญเสียข้อมูลตรงบริเวณขอบของข้อมูลรับเข้า รูป 2.5 แสดงการทำคอนโวลูชันแบบกว้างและการเสริมเติม



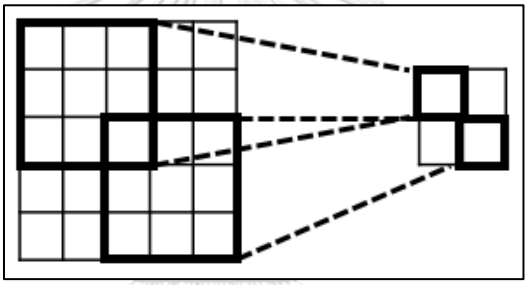
รูป 2.4 ตัวอย่างการทำคอนโวลูชัน โดยมีขนาดของข้อมูลรับเข้าขนาด 6 x 6 และเมทริกตัวกรองขนาด 3 x 3



รูป 2.5 การทำคอนโวลูชันแบบกว้างและการเสริมเติม

2.4.1.3 ขนาดของการก้าวข้าม (Stride Size)

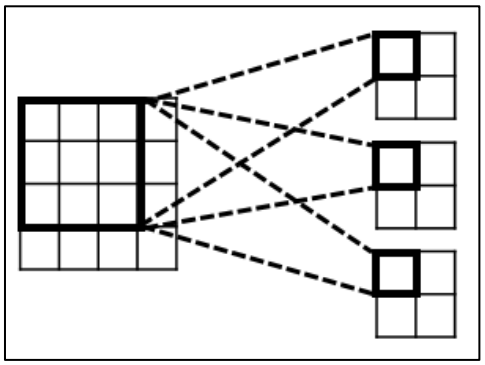
ขนาดของการก้าวข้ามคือจำนวนช่องของข้อมูลรับเข้า ที่จะทำการเลื่อนไปเมื่อทำการหาผลลัพธ์ของคอนโวลูชันในแต่ละช่อง โดยทั่วไปมักจะใช้ขนาดของการก้าวข้ามเป็น 1 รูป 2.6 แสดงลักษณะของการทำคอนโวลูชันที่มีขนาดของการก้าวข้ามเป็น 2



รูป 2.6 การทำคอนโวลูชันโดยมี ตัวกรองขนาด 3 x 3 และมีขนาดของการก้าวข้ามเป็น 2

2.4.1.4 จำนวนตัวกรอง (Number of Filters)

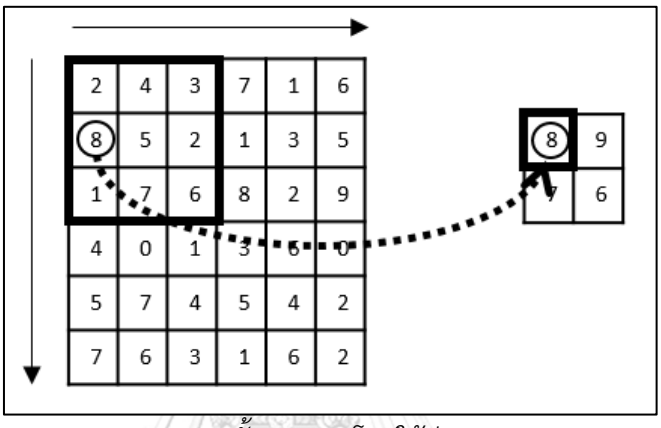
ในแต่ละชั้นคอนโวลูชันนั้นสามารถมีตัวกรองได้มากกว่าหนึ่ง โดยน้ำหนักของตัวกรองแต่ละตัวจะใช้แยกกัน ทั้งนี้จำนวนตัวกรองในชั้นคอนโวลูชันจะเป็นการกำหนดจำนวนช่องสัญญาณ (Channel) ของของมูลรับเข้าในชั้นถัดไป รูป 2.7 แสดงตัวอย่างการทำคอนโวลูชันโดยมีจำนวนตัวกรองเป็น 3



รูป 2.7 การทำคอนโวลูชันโดยมีจำนวนตัวกรองเท่ากับ 3

2.4.2 ชั้นการรวม (Pooling Layer)

ชั้นการรวมเป็นชั้นที่ทำหน้าที่ลดขนาดของข้อมูลลง เพื่อให้เหลือแค่เพียงข้อมูลที่สำคัญ ๆ เท่านั้น โดยทั่วไปมักจะทำการเลือกข้อมูลที่มีค่ามากที่สุดมาจากแต่ละช่วงของเมทริกซ์เพื่อสร้างเป็นเมทริกซ์ที่ขนาดเล็กลง (Max Pooling) ชั้นการรวมโดยใช้ค่ามากสุดจะทำการเลือกเฉพาะค่ามากที่สุดจากกลุ่มของข้อมูลที่เราสนใจ และนำไปใช้งานต่อไปในชั้นถัดไป จากรูป 2.8 เป็นการทำการรวมโดยค่ามากสุดบนเมทริกซ์ขนาด 6 x 6 โดยกลุ่มที่สนใจจะมีขนาด 3 x 3 ซึ่งขอบเขตของกลุ่มที่สนใจจะมีการเลื่อนไปจนครอบคลุมเมทริกซ์ต้นฉบับทั้งหมด



รูป 2.8 ชั้นการรวมโดยใช้ค่ามากสุด

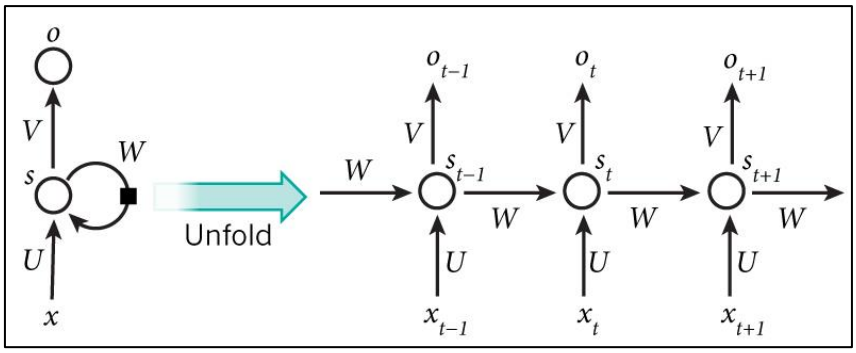
2.4.3 ชั้นการเชื่อมโยงเต็มรูปแบบ (Fully Connected Layer)

ในชั้นสุดท้ายของนิวรอลเน็ตเวิร์กคอนโวลูชัน มักจะเป็นการเชื่อมโยงเต็มรูปแบบนั่นคือ ในชั้นนี้ประกอบด้วยชั้นย่อย ๆ ที่มีเพอร์เซปตรอนอยู่จำนวนหนึ่ง โดยที่เพอร์เซปตรอนแต่ละตัวจะมีเส้นเชื่อมกับเพอร์เซปตรอนทุกตัวในชั้นก่อนหน้าและเพอร์เซปตรอนทุกตัวในชั้นถัดไป

Chulalongkorn University

2.5 นิวรอลเน็ตเวิร์กแบบวกกลับ (Recurrent Neural Network; RNN)

นิวรอลเน็ตเวิร์กแบบวกกลับหรือต่อไปนี้จะขอเรียกว่า RNN ถูกออกแบบมาให้ตอบสนองต่อการประมวลผลข้อมูลที่มีลำดับ (sequential) ซึ่งจะมีการส่งผ่านผลการประมวลผลจากข้อมูลในช่วงเวลาก่อนหน้าไปยังช่วงเวลาถัดไปดังตัวอย่างในรูป 2.9 ทั้งนี้โครงสร้างของ RNN จะคล้ายคลึงกับนิวรอลเน็ตเวิร์กทั่วไป ส่วนที่แตกต่างคือมีการส่งต่อชั้นลับ (hidden layer) ไปเป็นข้อมูลนำเข้าของช่วงเวลาถัดไป



รูป 2.9 แสดงตัวอย่างการทำงานของ ระบบเน็ตเวิร์กแบบวกกลับ [17]

สมการสำหรับ การคำนวณของ RNN เป็นดังต่อไปนี้

$$s_t = \sigma(Ws_{t-1} + Ux_t + b) \quad (2.27)$$

s_t คือ ค่าของชั้นลับ (hidden layer) ที่ช่วงเวลา t

W คือ ค่าของน้ำหนักที่ใช้คูณกับข้อมูลจากชั้นลับเมื่อช่วงเวลา t-1

U คือ ค่าของน้ำหนักที่ใช้คูณกับข้อมูลนำเข้า t

x_t คือ ค่าของข้อมูลนำเข้า ที่ช่วงเวลา t

σ คือ ฟังชันกระตุ้น

b คือ ไบแอส

จะเห็นได้ว่าค่าผลลัพธ์ของฟีเจอร์ซึ่งจะถูกส่งไปใช้หาคำตอบนั้นจะได้จากการรวมกันของผลลัพธ์ฟีเจอร์จากช่วงเวลาก่อนหน้าและข้อมูลที่เข้ามาในปัจจุบัน เน็ตเวิร์กประเภทนี้มีการนำไปใช้ในงานที่หลากหลายเช่น การแปลข้อความ การแบ่งประเภทของข้อความ รวมถึงการทำสรุปบทความ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2.5.1 นิวรอลเน็ตเวิร์กแบบความจำระยะสั้นแบบยาว (Long-Short Term Memory Neural Network; LSTM)

นิวรอลเน็ตเวิร์กประเภทนี้เป็นส่วนหนึ่งของการเรียนรู้แบบลึกที่มีขึ้นมาเพื่อแก้ปัญหาการลืมข้อมูลของ RNN ในกรณีที่ข้อมูลที่นำเข้านั้นมีความยาวมาก ทำให้การส่งค่าผลลัพธ์ฟีเจอร์ไม่สามารถที่จะเก็บรักษาข้อมูลในช่วงเวลาก่อนหน้าได้ทั้งหมด แนวคิดของนิวรอลเน็ตเวิร์กแบบความจำระยะสั้นแบบยาว [18] (ต่อไปนี้จะขอเรียกว่า LSTM) จึงนำเอาการเลือกจำข้อมูลเข้ามาช่วยเพื่อให้เน็ตเวิร์กทำการเลือกจำแต่ข้อมูลที่สำคัญเท่านั้น

LSTM ประกอบด้วยหน่วยความจำ (cell) แทนด้วย c , ประตูสัญญาณสำหรับรับเข้า (input gate) แทนด้วย i, ประตูสัญญาณสำหรับการลืม (forget gate) แทนด้วย f, ประตูสัญญาณสำหรับผลลัพธ์ (output gate) แทนด้วย o และผลคูณแบบอาดามาร์ (hadamard product) แทนด้วย $\circ$ สมการสำหรับการสร้างฟีเจอร์ผลลัพธ์มีดังนี้

$$f_t = \sigma(W_f s_{t-1} + U_f x_t + b_f) \quad (2.28)$$

$$i_t = \sigma(W_i s_{t-1} + U_i x_t + b_i) \quad (2.29)$$

$$o_t = \sigma(W_o s_{t-1} + U_o x_t + b_o) \quad (2.30)$$

$$c_t = f_t \circ c_{t-1} + i_t \circ (W_c s_{t-1} + U_c x_f + b_c)) \quad (2.31)$$

$$s_t = o_t \circ \sigma(c_t) \quad (2.32)$$

2.5.2 นิวรอลเน็ตเวิร์กแบบประตูสัญญาณวกกลับ (Gated Recurrent Unit; GRU)

นิวรอลเน็ตเวิร์กแบบประตูสัญญาณวกกลับหรือที่จะเรียกต่อไปว่า GRU [19] เน็ตเวิร์กนี้ได้นำเอา LSTM มาทำการปรับปรุงให้ลดจำนวนประตูสัญญาณลง โดยการใช้ประตูสัญญาณอัพเดท (update gate) มาใช้แทนที่ประตูสัญญาณการจำและประตูสัญญาณผลลัพธ์ อีกทั้งยังทำการรวมหน่วยความจำเข้ากับชั้นลับ ทำให้เน็ตเวิร์กมีความรวดเร็วมากขึ้นและใช้หน่วยความจำน้อยลง กำหนดให้ z แทนที่ประตูสัญญาณอัพเดท จะได้สมการออกมาดังนี้

$$f_t = \sigma(W_f s_{t-1} + U_f x_t + b_f) \quad (2.33)$$

$$z_t = \sigma(W_z s_{t-1} + U_z x_t + b_z) \quad (2.34)$$

$$\tilde{s}_t = tanh(W_c(z_t * s_{t-1}) + U_c x_t + b_c)) \quad (2.35)$$

$$s_t = (1 - f_t) * s_{t-1} + f_t \circ \tilde{s}_t \quad (2.36)$$

2.6 กลไกความสนใจ (Attention Mechanism)

กลไกความสนใจมักนำไปใช้ในนิวรอลเน็ตเวิร์กแบบลำดับต่อลำดับ (Sequence-to-Sequence Model) โดยการสร้างค่าความสนใจ $a_1 .. a_n$ให้กับแต่ละลำดับของข้อมูล ค่าความสนใจจะถูกนำไปคูณเข้ากับแต่ละลำดับของข้อมูล เพื่อสร้างเป็นเวกเตอร์ผลลัพธ์ที่จะถูกนำไปใช้ต่อ การหาค่าความสนใจ a ทำได้โดยดูจากลำดับชั้นลับในเน็ตเวิร์กแบบวกกลับ $s_1 .. s_n$ ร่วมกับลำดับชั้นลับของอีกลำดับ h_t ดังนี้

$$a_i = softmax(f_{att}(h_t, s_i)) \quad (2.37)$$

โดย f_{att} เป็นฟังก์ชันที่ใช้สำหรับการคำนวณค่าความสนใจ มีอยู่หลายประเภท ได้แก่

กำหนดให้

h_t คือ ค่าของชั้นลับ (hidden layer) ในลำดับชุด 1 ที่ช่วงเวลา t

s_i คือ ค่าของชั้นลับ (hidden layer) ในลำดับชุด 2 ที่ช่วงเวลา i

W_1 คือ ค่าของน้ำหนักที่ใช้คูณกับข้อมูลจากชั้นลับในลำดับชุด 1

W_2 คือ ค่าของน้ำหนักที่ใช้คูณกับข้อมูลจากชั้นลับในลำดับชุด 2

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

W_a คือ ค่าของน้ำหนักที่ใช้คูณกับข้อมูลจากชั้นลับในลำดับชุด 1 และลำดับชุด 2

v_a คือ ค่าของเวกเตอร์ของน้ำหนักที่ใช้หาค่าความสนใจ

2.6.1 ค่าความสนใจแบบบวก (Additive Attention)

การหาค่าความสนใจแบบบวกเป็นวิธีการหาค่าความสนใจที่ใช้นิวรอลเน็ตเวิร์กแบบป้อนไปข้างหน้าในการหาความสนใจระหว่างข้อมูลสองลำดับดังนี้ [20]

$$f_{att}(h_t,\ s_i)\ =\ v_a^T tanh(W_1 h_t + W_2 s_i) \tag{2.38}$$

2.6.2 ความสนใจแบบคูณ (Multiplicative Attention)

การหาค่าความสนใจแบบคูณ [21] เป็นวิธีการหาค่าความสนใจที่ใช้ การคูณด้วยค่าน้ำหนัก การหาค่าความสนใจนี้จะมีความเร็วกว่าการหาค่าความสนใจแบบบวก โดยสมการมีดังนี้

$$f_{att}(h_t,\ s_i)\ =\ h_t^T W_a s_i \tag{2.39}$$

2.6.3 ความสนใจส่วนตัว (Self-Attention)

การหาความสนใจโดยปกติจะหาจากการเทียบเคียงระหว่างลำดับสองลำดับ การหาความสนใจส่วนตัวจึงเป็นการหาค่าความสนใจโดยใช้เพียงลำดับของข้อมูลลำดับเดียว [22] การหาค่าความสนใจสำหรับความสนใจแบบดังกล่าวจึงมีความใกล้เคียงกับวิธีมาตรฐาน

การหาค่าความสนใจแบบบวก สำหรับการหาความสนใจส่วนตัว เป็นดังนี้

$$f_{att}(h_t)\ =\ v_a^T tanh(W_a h_t) \tag{2.40}$$

การหาค่าความสนใจแบบคูณ สำหรับการหาความสนใจส่วนตัว เป็นดังนี้

$$f_{att}(h_t)\ =\ h_t^T W_a h_t \tag{2.41}$$

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

2.6.4 ความสนใจแบบควบ (Co-Attention)

การหาความสนใจแบบควบ มีข้อดีที่สามารถที่จะหาความสนใจระหว่างสองลำดับไปได้พร้อมกัน [23] โดยใช้วิธีการสร้างเมทริกซ์ความใกล้ชิด (Affinity Matrix) ระหว่างข้อมูลในลำดับต่อลำดับขึ้นมา ดังแสดงในรูป 2.10 แต่ละช่องของตารางจะเก็บค่าความสนใจระหว่างข้อมูลเอาไว้

ฟังก์ชันความคล้าย (F) ที่นิยมใช้กันได้แก่

การหาค่าความสนใจด้วยการคูณเชิงสเกลาร์ (dot product) [23]

$$F(h_t, s_i)\ =\ h_t\ \cdot\ s_i \tag{2.42}$$

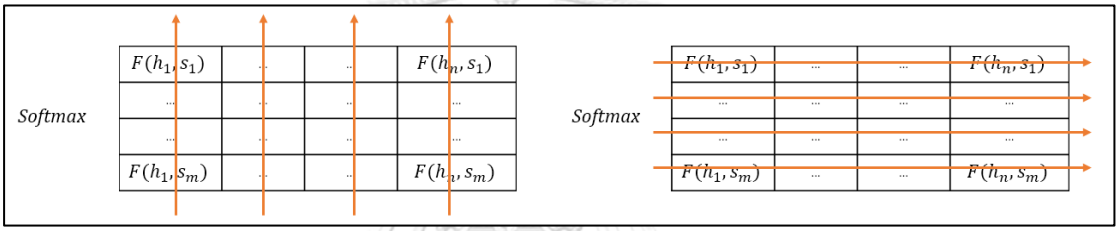
$F(h_1, s_1)$	...	...	$F(h_1, s_n)$
...	...	...	...
...	...	...	...
$F(h_m, s_1)$	...	...	$F(h_m, s_n)$

รูป 2.10 ตารางความคล้ายที่เกิดจากลำดับ h และ s

การหาค่าความสนใจแบบคูณ [24]

$$F(h_t, s_i) = h_t W_a s_i \tag{2.43}$$

สุดท้ายค่าความสนใจจะหาได้จากการนำเอา ตารางความสนใจไปเข้าฟังก์ชัน Softmax ซึ่งจะแสดงให้เห็นดังในรูป 2.11 ในแนวคอลัมน์และแถว



รูป 2.11 การหาค่าความสนใจจากเมทริกซ์ความใกล้ชิดในแนวคอลัมน์ (ซ้าย) และแนวแถว (ขวา)

2.7 การคำนวณความแม่นยำ (Evaluation Metrics)

การวัดประสิทธิภาพของการตอบคำถามจากบทความจะวัดจาก ค่าเอฟวัน (F1) และค่าความถูกต้อง ตัวชี้วัดที่จะใช้ในการคำนวณมีดังต่อไปนี้

TP คือ จำนวนของคำที่คำตอบของระบบตรงกับคำที่ปรากฏในเป้าหมาย

FP คือ จำนวนของคำที่คำตอบของระบบไม่ตรงกับคำที่ปรากฏในเป้าหมาย

FN คือ จำนวนของคำที่คำตอบของระบบไม่ได้ตอบแต่ปรากฏในเป้าหมาย

$Match$ มีค่าเป็น 1 เมื่อ คำตอบของระบบตรงกับคำตอบเป้าหมายทุกประการ

2.7.1 ตัววัดประสิทธิภาพ (F1 Measurement)

การคำนวณหาค่าความเที่ยง (Precision) ค่าความระลึก (Recall) และค่าเอฟวัน (F1 หรือ F_1) นั้น สามารถคำนวณได้จาก

$$Precision = \frac{TP}{TP+FP} \tag{2.44}$$

$$Recall = \frac{TP}{TP+FN} \tag{2.45}$$

$$F_1 = \frac{2*Precision\&recall}{Recision+Recall} \tag{2.46}$$

ค่าเอฟวันที่ใช้วัดจะเป็นแบบเฉลี่ยรวม (Macro F1) ซึ่งได้จากการหาค่าเฉลี่ยของเอฟวันจากทุกคำตอบ กำหนดให้ N คือจำนวนของคำถามทั้งหมด กำหนดให้ $F_{1,i}$ คือค่าเอฟวันของคำถามข้อ i โดยเราสามารถหาค่าเอฟวันเฉลี่ยรวม ได้จากสูตรดังต่อไปนี้

$$F_1 = \frac{\sum_{i=1}^{N} F_{1,i}}{N} \tag{2.47}$$

2.7.2 ค่าความถูกต้อง (Exact Match)

ค่าความถูกต้องจะใช้วัดว่าผลลัพธ์โดยรวมเมื่อมองจากความถูกต้องทุกประการเป็นเท่าไหร่ กำหนดให้ $target$ คือวลีที่เป็นคำตอบเป้าหมาย และ $predict$ คือวลีที่เป็นผลลัพธ์สำหรับคำถามข้อดังกล่าว โดยสามารถหาได้จากสมการดังต่อไปนี้

$$Exact\ Match = \frac{\sum_{i=1}^{N} Match(target_i,\ predict_i)}{N} \tag{2.48}$$

2.7.3 การทดสอบแม็คนมาร์ (Mcnemar Test)

การทดสอบนี้มีจุดประสงค์เพื่อหาความแตกต่างของผลลัพธ์ที่ได้จากก่อนและหลังการใช้วิธีการ A ว่ามีการเปลี่ยนแปลงอย่างมีนัยสำคัญหรือไม่ โดยเปรียบเทียบผลการทดลองที่ต่างกัน

	หลังใช้วิธีการ A ไม่สำเร็จ	หลังใช้วิธีการ A สำเร็จ
ก่อนใช้วิธีการ A ไม่สำเร็จ	N_{ff}	N_{fs}
ก่อนใช้วิธีการ A สำเร็จ	N_{sf}	N_{ss}

โดย N_{ss} คือจำนวนตัวอย่างที่สำเร็จทั้งคู่

N_{ff} คือจำนวนตัวอย่างที่ไม่สำเร็จทั้งคู่

N_{sf} คือจำนวนตัวอย่างที่ก่อนใช้วิธีการ A สำเร็จแต่หลังใช้ไม่สำเร็จ

N_{fs} คือจำนวนตัวอย่างที่ก่อนใช้วิธีการ A ไม่สำเร็จแต่หลังใช้สำเร็จ

มีสมมุติฐานดังนี้

$H_0 : P(N_{sf}) = P(N_{fs})$ หรือ การใช้วิธีการ A ไม่ให้ผลที่แตกต่าง

$H_1 : P(N_{sf}) \neq P(N_{fs})$ หรือ การใช้วิธีการ A ให้ผลที่แตกต่าง

โดยสามารถหาค่าการกระจายตัวแบบไคสแควร์ χ^2 (Chi-Square Distribution) ของการทดสอบแม็คนมาร์ ได้ดังต่อไปนี้

$$\chi^2 = \frac{(|N_{fs} - N_{sf}| - 1)^2}{\sqrt{N_{fs} + N_{sf}}} \tag{2.49}$$

โดยจะนำค่า χ^2 ที่ได้ไปเทียบหาจากตาราง χ^2 ด้วยระดับความอิสระ (d.f.) เป็น 1 เพื่อทดสอบความมีนัยสำคัญ

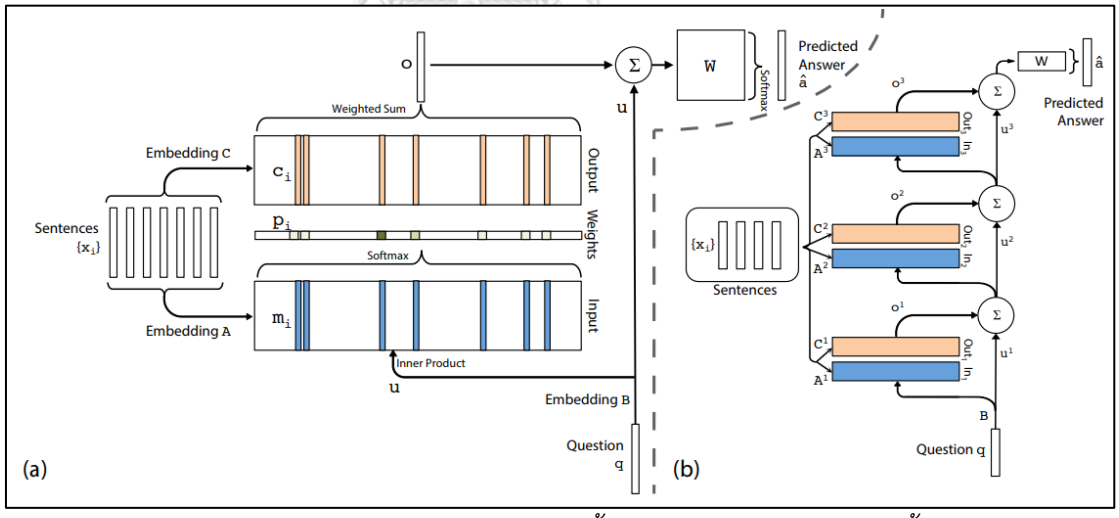
จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 3
งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์ฉบับนี้ เป็นงานวิจัยที่ใช้สำหรับการอ่านตอบคำถามด้วยคอมพิวเตอร์ โดยงานวิจัยแต่ละชิ้นจะใช้นิวรอลเน็ตเวิร์กที่ประกอบขึ้นจากเน็ตเวิร์กแบบต่าง ๆ ที่ได้กล่าวไปข้างต้น สำหรับในหัวข้อนี้จะแบ่งกลุ่มของงานวิจัยออกเป็นสองส่วนหลัก ๆ ได้แก่ นิวรอลเน็ตเวิร์กที่ใช้สำหรับการตอบคำถามที่มีหลายความสัมพันธ์ และนิวรอลเน็ตเวิร์กที่ใช้สำหรับการอ่านบทความเพื่อตอบคำถามด้วยการเรียนรู้เชิงลึก

3.1 วิธีการตอบคำถามที่มีหลายความสัมพันธ์ (Question with Multiple Relationships)

ในกลุ่มงานวิจัยที่ทำการศึกษาหาวิธีแก้ปัญหาในกลุ่มคำถามที่มีหลายความสัมพันธ์ เน็ตเวิร์กที่ได้รับความนิยมคือ เน็ตเวิร์กความจำ (End-to-End Memory Network) ซึ่งถูกเสนอในปี 2015 โดย Sainbayar Sukbhaatar และคณะ [25] เนื่องจากมีการใช้หน่วยความจำถาวรและการอ่านซ้ำหลายครั้ง (Multiple Hops) มาทำการตอบคำถาม ทำให้สามารถเชื่อมโยงความสัมพันธ์ของคำที่อยู่ห่างกันมากได้ดังแสดงในรูป 3.1

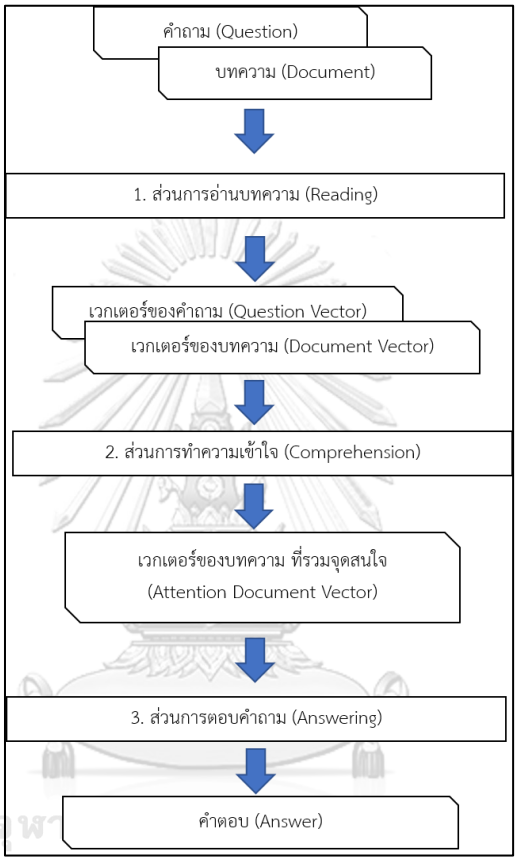


รูป 3.1 เน็ตเวิร์กความจำ (a) แบบทำการอ่านครั้งเดียว (b) แบบอ่านหลายครั้ง [25]

จุดเด่นของเน็ตเวิร์กความจำคือตัวเน็ตเวิร์กมีหน่วยความจำและส่วนการประมวลผลที่แยกออกจากกัน โดยคำถามและบทความที่เข้ามาจะถูกเปลี่ยนให้เป็นเวกเตอร์ด้วยการฝังคำ (Word Embedding) ซึ่งตัวบทความกับคำถามจะถูกอ่านซ้ำหลายครั้ง โดยในแต่ละครั้งเน็ตเวิร์กจะมีการสรุปรวมสิ่งที่ได้อ่านไปในรอบนั้น แล้วส่งต่อไปใช้ประกอบในการอ่านในรอบถัดไป ดังเช่นจากรูป 3.1b ที่ผลการอ่านครั้งแรก o^1 กับคำถาม u^1 จะถูกรวมกัน เป็น u^2 เพื่อใช้ในการอ่านรอบถัดไป

3.2 นิวรอลเน็ตเวิร์กที่ใช้สำหรับการอ่านบทความ เพื่อตอบคำถามด้วยการเรียนรู้เชิงลึก

ปัจจุบันงานวิจัยด้านการอ่านบทความเพื่อตอบคำถามด้วยการเรียนรู้เชิงลึกมีรูปแบบที่คล้ายคลึงกัน โดยจะมีส่วนประกอบย่อย 3 ส่วน ได้แก่ ส่วนการอ่านบทความ ส่วนการทำความเข้าใจ และส่วนการตอบคำถาม ดังแสดงในรูป 3.2



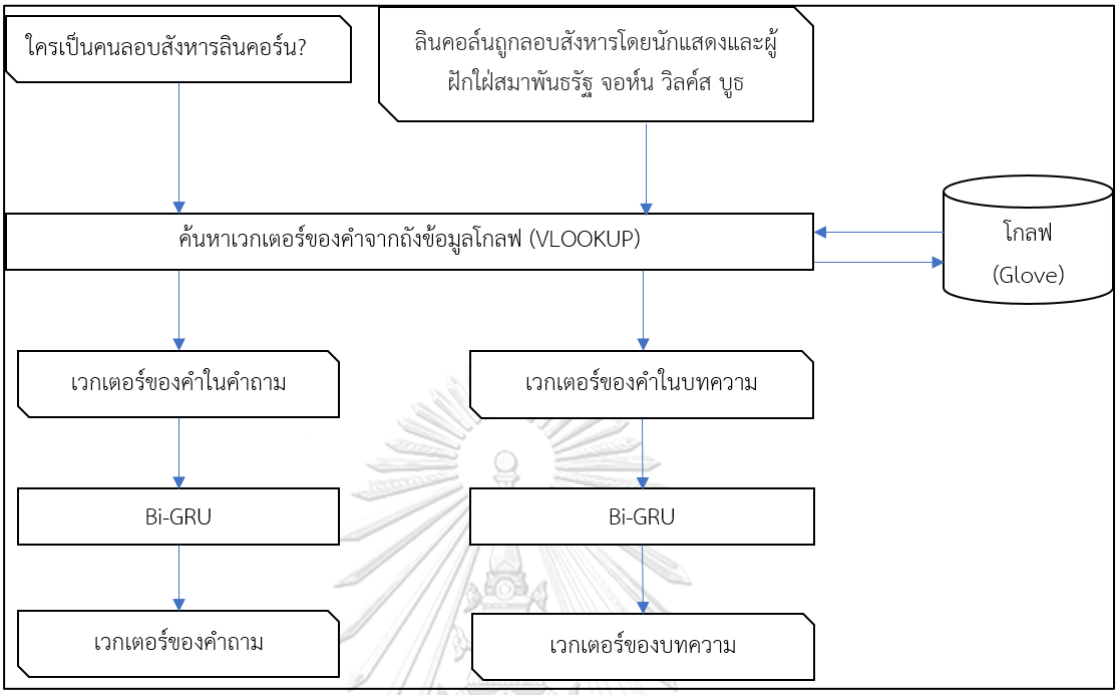
รูป 3.2 กระบวนการทำงานของนิวรัลเน็ตเวิร์กที่ใช้สำหรับการอ่านบทความเพื่อตอบคำถาม ที่มีข้อมูลเข้าเป็นคำถามกับบทความ และมีผลลัพธ์เป็นคำตอบของคำถามซึ่งเป็นวลีจากบทความ

3.2.1 ส่วนการอ่านบทความ

ข้อมูลที่เข้ามาในส่วนนี้จะประกอบด้วยคำถามและบทความ ซึ่งจะถูกเปลี่ยนให้อยู่ในรูปของเวกเตอร์ด้วยการฝังคำ เพื่อที่จะนำไปอ่านแล้วส่งไปยังส่วนการทำความเข้าใจบทความ ดังแสดงไว้ในรูป 3.3 ทั้งนี้วิธีการฝังคำและวิธีการอ่านในแต่ละงานวิจัยมีรูปแบบที่แตกต่างกันดังนี้

วิธีการฝังคำ งานวิจัยเกือบทั้งหมดที่ใช้การเรียนรู้เชิงลึกนิยมที่จะเลือกใช้โกลฟ (Glove) ซึ่งเป็นเวกเตอร์ของคำที่ได้รับการสอนมาก่อน (Pretrained word vector) [16, 23, 26-29] เพื่อให้ได้เวกเตอร์ที่สามารถแทนความหมายของคำได้ใกล้เคียงมากขึ้น ต่อมาได้มีการนำเอาการฝังคำ

ด้วยอักขระ (Character Embedding) เข้ามาเสริมกับโกลฟเพื่อช่วยในกรณีของคำศัพท์ที่ไม่อยู่ในพจนานุกรม (Out-of-Vocabulary) [16, 26, 29]



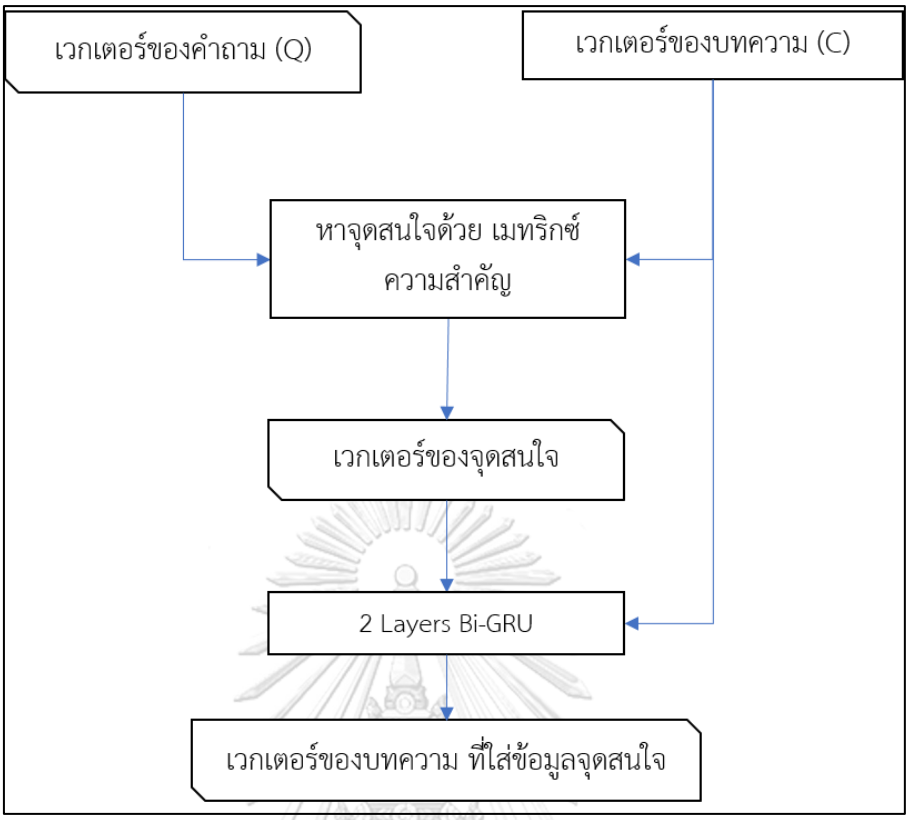
รูป 3.3 ตัวอย่างส่วนการอ่านบทความ ข้อมูลนำเข้าจะเป็นประโยคคำถามและบทความ โดยมีผลลัพธ์เป็นเวกเตอร์ของข้อมูลที่ได้ทำการอ่านมาแล้ว ในที่นี้ส่วนการอ่านใช้ Bi-GRU

การอ่านจะใช้นิวรอลเน็ตเวิร์กในตระกูล RNN แบบสองทิศทาง (Bidirectional; Bi) เช่น Bi-RNN [29] Bi-GRU [27] และ Bi-LSTM [16, 23, 26] ทั้งนี้ก็เพื่อให้ผลลัพธ์ที่จะถูกส่งไปยังส่วนการทำความเข้าใจมีข้อมูลจากการอ่านทั้งหน้าไปหลังและหลังมาหน้า

3.2.2 ส่วนการทำความเข้าใจบทความ (Comprehension)

ในส่วนนี้จะมีการใช้กลไกความสนใจแบบต่าง ๆ (Attention Mechanism) เพื่อสร้างเป็นฟีเจอร์เพิ่มเติมให้กับโมเดลในการอ่านทำความเข้าใจ แล้วจึงนำไปทำการอ่านซ้ำด้วยนิวรอลเน็ตเวิร์กในตระกูล RNNดังแสดงในรูป 3.4

วิธีการที่ใช้กันเป็นหลักในการหาความสนใจได้แก่ การหาความสนใจแบบควบที่เริ่มมีการนำเอามาใช้โดย Minjoon Seo และคณะ [26] โดยการหาความสนใจจะมีการสร้างเมทริกซ์ความใกล้ชิดขึ้นมา ซึ่งหาได้จากการหาความคล้ายระหว่างเวกเตอร์ของคำทุกคำในประโยคคำถามและคำทุกคำในบทความ ซึ่งสามารถนำไปหาความสนใจได้สามแบบคือ



รูป 3.4 ส่วนการทำความเข้าใจบทความของเน็ตเวิร์ก

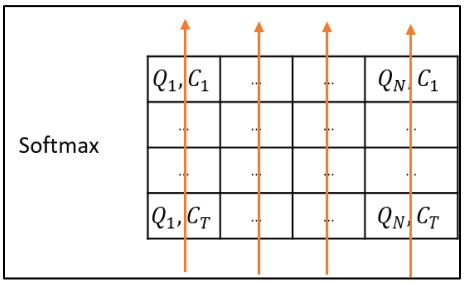
3.2.2.1 ความสนใจเมื่อมองคำถามต่อบทความ (Context-to-Query; C2Q)

ความสนใจนี้หาได้จากการนำเอาเมทริกซ์ความใกล้ชิดไปหาค่า Softmax ตามแนวแถว (row) ดังรูป 3.5 ความหมายของการหาความสนใจแบบนี้คือ เมื่อมองคำจากบทความแล้ว คำในคำถามคำไหนบ้างที่มีความคล้ายกับคำในบทความมากที่สุด

รูป 3.5 แสดงการหาความสนใจเมื่อมองคำถามต่อบทความ ด้วยการหาค่า Softmax ในแนวแถว

3.2.2.2 ความสนใจระหว่างบทความต่อคำถาม (Query-to-Context; Q2C)

ความสนใจนี้หาได้จากการนำเอาเมทริกซ์ความใกล้ชิดไปหาค่า Softmax ตามแนวคอลัมน์ดังรูป 3.6 ความหมายของการหาความสนใจแบบนี้คือ เมื่อมองจากคำถามแล้ว คำในบทความคำไหนบ้างที่มีความคล้ายกับคำในคำถาม



รูป 3.6 แสดงการหาความสนใจเมื่อมองบทความต่อคำถาม ด้วยการหาค่า Softmax ในแนวคอลัมน์

3) ความสนใจจากการปรับแนวส่วนตัว (Self-Alignment Attention; C2C)

ความสนใจนี้ต้องกาสร้างเมทริกซ์ความใกล้ชิดภายในระหว่างตัวบทความเอง โดยสามารถหาได้จากการนำเอาเมทริกซ์ความใกล้ชิด ไปหาค่า Softmax ตามแนวแถว ความหมายของการหาความสนใจแบบนี้คือ เมื่อมองคำในบทความด้วยตัวเองแล้ว คำอื่นในบทความคำไหนบ้างที่มีความเกี่ยวข้องกันมาก

โมเดลที่ใช้ความสนใจจากเมทริกซ์ความใกล้ชิดได้แก่ [16, 23, 26, 30] ซึ่งแต่ละโมเดลจะใช้ประเภทของความสนใจคละกันไป แต่ความสนใจเมื่อมองคำถามต่อบทความจะเป็นตัวที่เพิ่มประสิทธิภาพได้มากที่สุด

3.2.3 ส่วนการตอบบทความ (Answering)

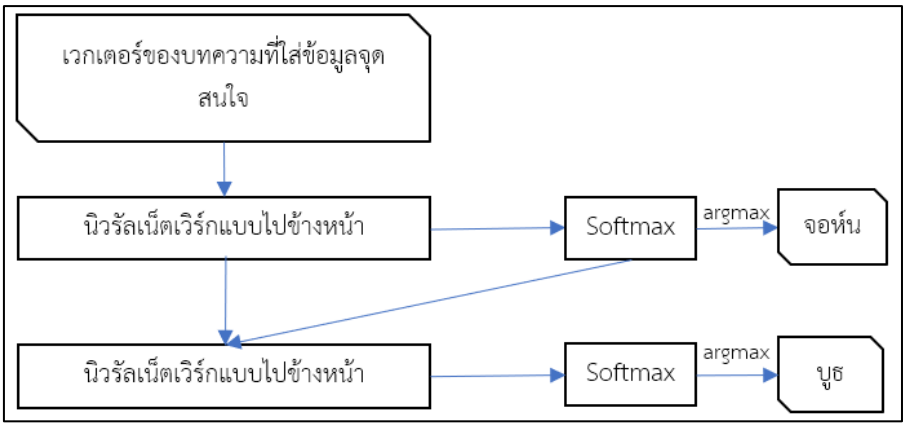
Shuohang Wang และคณะ [28] ได้เสนอวิธีการหาคำตอบจากบทความโดยการหาตำแหน่งอ้างอิงของคำตอบคำแรกและคำตอบคำสุดท้ายจากบทความ (Answer Pointer Prediction) เพื่อที่จะสร้างคำตอบ ดังแสดงในรูป 3.7 คำว่า ‘จอห์น’ กับ ‘บูธ’ คือ คำแรกและคำสุดท้ายของคำตอบตามลำดับ

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ลินคอล์น	ถูก	ลอบ สังหาร	...	จอห์น	วิลค์ส	บูธ	การ	ลอบ สังหาร
				Start		End		

รูป 3.7 เน็ตเวิร์กการอ่านเพื่อตอบคำถามจะทำการทำนายตำแหน่งของ ‘จอห์น’ คำตอบคำแรก และ ‘บูธ’ คำตอบคำสุดท้าย

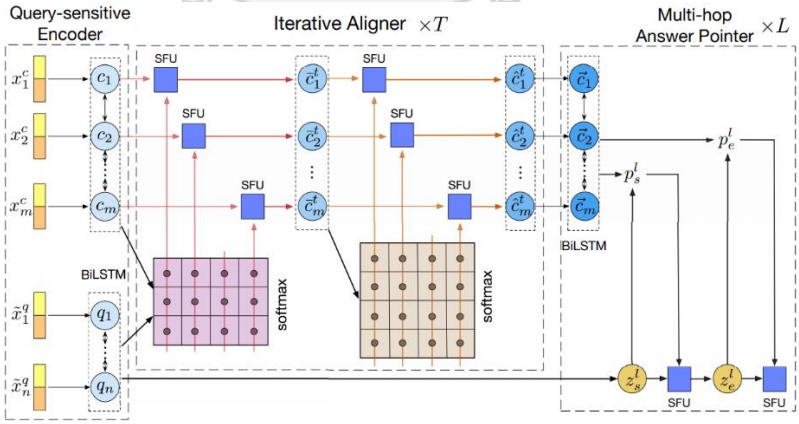
วิธีการตอบคำถามที่ใช้กันส่วนมากจะนำเอาเวกเตอร์ของบทความที่มีการอ่านทำความเข้าใจมาแล้วไปผ่านนิวรอลเน็ตเวิร์กแบบไปข้างหน้า ก่อนจะนำไปสร้างค่าความน่าจะเป็นด้วยฟังก์ชัน Softmax เพื่อหาคำแรกของคำตอบจากตำแหน่งที่มีค่าความน่าจะเป็นที่สูงที่สุด แล้วค่อยนำเอาค่าความน่าจะเป็นที่ได้ไปใช้ช่วยหาคำสุดท้ายของคำตอบ



รูป 3.8 ส่วนการหาคำตอบที่ใช้ของเน็ตเวิร์ก

Yelong Shen และคณะ [27] ได้เสนอโมเดลที่มีให้มีการอ่านผ่านนิวรอลเน็ตเวิร์กแบบไปข้างหน้าหลายครั้ง (Multiple Hops) มารวมเข้ากับการเรียนรู้แบบสนับสนุน (Reinforcement Learning) เพื่อที่จะให้โมเดลสามารถเรียนรู้ได้เองว่าจะให้ทำการหยุดอ่านเมื่อไหร่

Minghao Hu Yuxing Peng และ Xipeng Qiu [16] ได้เสนอโมเดลการอ่านด้วยการจำ (Mnemonic Reader) ที่มีชั้นสำหรับการอ่านและจำ (Mnemonic Pointing Layer) มาใช้ร่วมกับการหาคำตอบโดยการอ่านหลายครั้ง ซึ่งมีความคล้ายกับเน็ตเวิร์กความจำดังรูป 3.9b ที่มีการนำเอาเวกเตอร์ของคำถาม q มารวมเข้ากับเวกเตอร์ของบทความที่มีการคูณความสนใจแล้ว v ไปทำการจำเข้าไปในหน่วยความจำ m เพื่อนำมาตอบปัญหาที่ต้องใช้หลายความสัมพันธ์



รูป 3.9 ตัวอย่างการทำงานของระบบการอ่านเพื่อตอบคำถามของ โมเดลการอ่านด้วยการจำ [16]

งานวิจัยโมเดลการตอบคำถามจากบทความด้วยการเรียนรู้เชิงลึก ส่วนประกอบของโมเดลในงานวิจัยเกือบทั้งหมดจะมีวิธีการสร้างที่ซ้อนทับกันอยู่ โดยโมเดลที่เหมาะสมกับการแก้ปัญหาที่มีหลายความสัมพันธ์มากที่สุดคือโมเดลการอ่านด้วยการจำ เนื่องจากมีระบบการอ่านหลายครั้งและมีหน่วยความจำที่ทำให้สามารถเชื่อมโยงคำตอบได้ดีกว่าวิธีอื่น

ทั้งนี้ลักษณะของโมเดลการอ่านเพื่อตอบคำถามพื้นฐานที่จะนำไปใช้งาน เมื่อเปรียบเทียบกับโมเดลการอ่านด้วยการจำจะมีลักษณะดังตารางที่ 3.1 ต่อไปนี้

ตารางที่ 3.1 แสดงส่วนประกอบที่ใช้ใน Mnemonic Reader กับโมเดลพื้นฐานที่ใช้ในงานวิจัย

ส่วนประกอบ	**Mnemonic Reader**	**โมเดลพื้นฐานที่ใช้**	**หมายเหตุ**
1. ส่วนการอ่านบทความ			
วิธีฝังคำ	โกลฟ, การฝังคำด้วยอักขระ	โกลฟ, การฝังคำด้วยอักขระ	-
วิธีอ่าน	Bi-LSTM	Bi-GRU	Bi-GRU มีประสิทธิภาพที่เทียบเท่ากับ LSTM แต่ทำงานได้เร็วกว่า
2. ส่วนการทำความเข้าใจ			
กลไกความสนใจ	ความสนใจแบบควบ [C2Q, C2C]	ความสนใจแบบควบ [C2Q, Q2C, C2C]	เพิ่มความสนใจแบบควบ Q2C
วิธีอ่าน	Bi-LSTM	Bi-GRU 2 ชั้น	การใช้นิวรอลเน็ตเวิร์กแบบวกกลับ (RNN) ที่ต่อกันสองชั้น จะสามารถดักจับข้อมูลที่มีความซับซ้อนได้ดีกว่า
3. ส่วนการตอบคำถาม			
วิธีการอ่านหลายครั้ง	การอ่านและจำ	การอ่านและจำ	-
วิธีการให้คำตอบ	นิวรอลเน็ตเวิร์กแบบไปข้างหน้า	นิวรอลเน็ตเวิร์กแบบไปข้างหน้า	-

จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4
แนวคิดในการดำเนินงานและวิธีการที่นำเสนอ

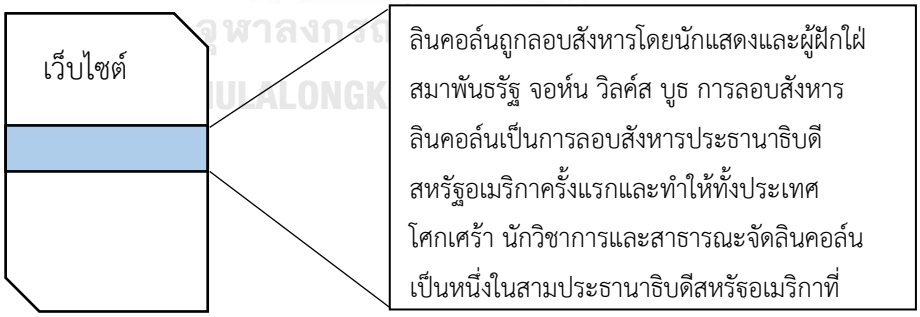
ในบทนี้นำเสนอวิธีการเรียนรู้เชิงลึกเพื่อนำมาใช้ในการตอบคำถามจากบทความ โดยจะกล่าวถึง (1) การประมวลผลข้อมูลเบื้องต้น (2) วิธีการที่นำเสนอ (3) เน็ตเวิร์กสำหรับการอ่านเพื่อตอบคำถามโดยใช้การเรียนรู้เชิงลึก

4.1 การประมวลผลข้อมูลเบื้องต้น (Preprocessing)

ข้อมูลที่ได้จำเป็นที่จะต้องมีการประมวลผลก่อนที่จะนำไปใช้ ได้แก่ การคัดเลือกบทความ การตัดคำ การนอร์มอลไลซ์ตัวอักษร และการสกัดคำอ้างอิง เครื่องมือที่นำมาใช้คือ Stanford Core NLP ที่เป็นชุดคำสั่งสำหรับการประมวลผลทางภาษา

4.1.1 การคัดเลือกบทความ (Paragraph Selection)

ขั้นตอนหนึ่งที่สำคัญในการใช้ชุดข้อมูลสัพเพเหระคือการเลือกบทความที่จะนำมาใช้เป็นข้อมูลขาเข้า เนื่องจากในชุดข้อมูลสัพเพเหระจะให้เอกสารที่จะใช้ในการตอบคำถามมาทั้งหน้าเว็บไซต์ ซึ่งแตกต่างจากชุดข้อมูลสควอดที่จะเลือกเอาเฉพาะย่อหน้าที่มีคำตอบมาให้ ดังนั้นขั้นตอนแรกในการที่จะเอาชุดข้อมูลสัพเพเหระไปใช้จึงจะต้องเลือกย่อหน้าที่คิดว่าเหมาะสมที่จะใช้เป็นคำตอบเสียก่อน ดังแสดงในรูป 4.1



รูป 4.1 ชุดข้อมูลสัพเพเหระจะให้ทั้งหน้าเว็บไซต์มา การเอาไปใช้งานจึงต้องมีการคัดเลือกย่อหน้าที่เกี่ยวข้องที่สุดจากคำถามเพื่อไปใช้ในการตอบคำถาม

ทั้งนี้วิธีการในการเลือกย่อหน้าที่จะนำมาใช้มีกระบวนการดังรูป 4.2

(1) นำเอาคำตอบไปทำการเลือกเซตของย่อหน้าในเอกสารที่มีคำตอบอยู่เท่านั้น แต่สำหรับชุดข้อมูลที่คำตอบถูกซ่อนอยู่ เราจะเลือกทุกย่อหน้าในบทความเข้าไปในเซต

(2) คำนวณค่าทีเอฟไอดีเอฟให้กับคำถามและย่อหน้าทุก ๆ ย่อหน้าที่เลือกมา

(3) นำค่าทีเอฟไอดีเอฟของคำถามไปเปรียบเทียบกับค่าทีเอฟไอดีเอฟของย่อหน้าที่เลือกมาทั้งหมด เพื่อหาย่อหน้าที่มีความคล้ายคลึงกับคำถามมากที่สุด โดยดูจากค่าความคล้ายแบบโคไซน์ (Cosine Similarity)

รูป 4.2 กระบวนการคัดเลือกย่อหน้าจากชุดข้อมูลสัพเพเหระ เพื่อไปใช้ในการเรียนรู้เชิงลึก

4.1.2 การตัดคำ (Word Tokenization)

ชุดข้อมูลที่ใช้จำเป็นที่จะต้องนำมาทำการตัดคำด้วยเครื่องมือให้เป็นคำที่เล็กที่สุดที่เป็นไปได้ แม้ว่าข้อความดังกล่าวจะเป็นภาษาอังกฤษก็ตาม ทั้งนี้ก็เพื่อที่จะลดโอกาสที่จะเจอคำที่ไม่เคยเห็น (unknown word) ให้น้อยที่สุด ตัวอย่างของการตัดคำได้แสดงไว้ในตารางที่ 4.1

ตารางที่ 4.1 ตัวอย่างการตัดคำโดยใช้ Stanford Core NLP

ประโยค	ประโยคที่ถูกตัดคำแล้ว
He’s a technician.	[He, ’s, a, technician, .]
She gave birth to a son, Jochi (1185–1226).	[She, gave, to, a, son, ‘,’ , Jochi, (, 1185, –, 1226,), .]

4.1.3 การนอร์มอลไลซ์ตัวอักษร (Character Normalization)

แม้ว่าข้อมูลที่ได้จะมาจากบทความภาษาอังกฤษ แต่ตัวเนื้อหานั้นมักที่จะมีการพิมพ์ด้วยตัวอักษรตระกูลยูนิโคด (Unicode) ทำให้มีโอกาสเจอตัวอักษรที่แม้ว่าจะเหมือนกันแต่รหัสยูนิโคดต่างกัน เช่น วงเล็บในภาษาไทยกับภาษาอังกฤษ ทำให้จำเป็นที่จะต้องมีการนอร์มอลไลซ์ตัวอักษรเหล่านั้นให้เป็นตัวเดียวกัน ทั้งนี้ก็เพื่อที่จะลดปริมาณของการเจอคำที่ไม่เคยเห็นลง ประเภทของการนอร์มอลไลซ์ที่นำมาใช้ ได้แสดงอยู่ใน ตารางที่ 4.2

4.1.4 การสกัดคำอ้างอิง (Coreference Extraction)

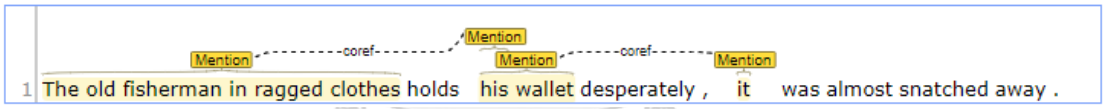
คำอ้างอิงคือคำที่ใช้ในการเรียกอ้างอิงไปถึงบุคคล สถานที่ หรือสิ่งของที่เป็นอย่างเดียวกัน ในรูป 4.3 คำว่า “his” มีการอ้างอิงไปถึง “The old fisherman in ragged clothes” ซึ่งในวิทยานิพนธ์ฉบับนี้ได้มีการเอาคำอ้างอิงไปใช้งานเพื่อเพิ่มประสิทธิภาพของคำถามที่มีหลายความสัมพันธ์ (Multiple Relationships)

ตารางที่ 4.2 ประเภทและตัวอย่างผลลัพธ์การนอร์มอลไลซ์ตัวอักษร

ประเภทการนอร์มอลไลซ์ตัวอักษร	ตัวอย่าง	ตัวอย่างผลลัพธ์หลังการนอร์มอลไลซ์
กลุ่มวงเล็บ	(,), [,], {, }	-LRB-, -RRB-, -LSB-, -RSB--LCB-, -RCB-
กลุ่มขีด	-, —	--
กลุ่มเศษส่วน	$^1/_2$	1 / 2

เครื่องมือที่นำมาใช้ในการสกัดคำอ้างอิงได้แก่ Statistical Coreference Resolution จาก Stanford Core NLP โดยมีผลความแม่นยำอยู่ที่ F1 56.2 น้อยกว่าแบบ Neural Coreference Resolution ที่แม่นยำที่สุดอยู่ F1 3.8 แต่สามารถสกัดคำอ้างอิงได้เร็วกว่าเกือบ 6 เท่าตัว

จากรูป 4.3 กลุ่มของคำอ้างอิงจะมีอยู่ 2 กลุ่มด้วยกัน ได้แก่ “The old fisherman in ragged clothes” ซึ่งถูกอ้างอิงโดยคำว่า “his” กับกลุ่มของ “his wallet” ที่ถูกอ้างอิงโดยคำว่า “it” ทั้งนี้ตัวเครื่องมือจะมองว่าส่วนขยายของคำหนึ่งคำจะถือว่ารวมอยู่ในคำ ๆ นั้นด้วย ซึ่งในบางครั้งส่วนขยายจะมีความยาวมากเกินไป ดังนั้นเราจึงทำการตัดความยาวของคำให้เหลือไม่เกิน 3 คำเท่านั้น ในทีนี้คำที่เรานำไปใช้จาก “The old fisherman in ragged clothes” จึงเหลือเพียง “The old fisherman”



รูป 4.3 ผลลัพธ์การสกัดคำอ้างอิงโดยใช้ Stanford Core NLP

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

4.2 วิธีการที่นำเสนอ (Proposed Method)

ในส่วนนี้จะอธิบายถึงการทำงานของวิธีการที่เสนอ 3 วิธีด้วยกัน ได้แก่ เวกเตอร์คำอ้างอิง การตอบแบบสองทาง และฟังก์ชันต้นทุนจากความยาวของคำตอบ

4.2.1 เวกเตอร์คำอ้างอิง (Coreference Vector)

การใช้การเรียนรู้ด้วยคอมพิวเตอร์เพื่อหาคำอ้างอิง (Coreference Resolution) นั้นเป็นปัญหาที่ค่อนข้างแพร่หลายมากในการประมวลผลภาษาธรรมชาติ [31, 32] แต่ในทางกลับกันการนำเอาเวกเตอร์คำอ้างอิงเหล่านั้นไปใช้งานกลับไม่ได้รับความสนใจนัก [33] โดยเฉพาะในกลุ่มงานทางด้านการเรียนรู้เชิงลึก

ในงานวิจัยชิ้นนี้จึงได้เสนอวิธีการสร้างเวกเตอร์อ้างอิงเพื่อนำมาใช้ในการเรียนรู้เชิงลึก โดยมีแนวคิดที่จะแทนความเป็นสมาชิกให้กับกลุ่มคำที่มีการอ้างอิงไปยังคำ ๆ เดียวกัน โดยหากคำ ๆ

นั้นอยู่ในกลุ่มไหน ค่าเวกเตอร์ของกลุ่มนั้นจะมีค่าเป็น 1 สำหรับคำนั้น แต่หากไม่อยู่จะมีค่าเป็น 0 เพื่อจะทำการเชื่อมคำที่อยู่ห่างจากกันโดยดูจากค่าเวกเตอร์ของกลุ่มคำอ้างอิง

ตัวอย่างเช่นจากรูป 4.3 จะสามารถแปลงคำอ้างอิงที่สกัดออกมาให้เป็นเวกเตอร์ได้ดังตารางที่ 4.3 โดยแถวที่สองของตารางจะแสดงถึงกลุ่มของคำที่อ้างอิงไปหา“fisherman” ดังนั้นแถวที่สองของคำว่า “The” “old” “fisherman” และ“his” จึงมีค่าเป็น 1 แต่จะมีค่าในแถวที่สามเป็น 0 ยกเว้นแต่คำว่า “his” ซึ่งอ้างอิงไปยังทั้งสองกลุ่มทำให้มีค่าเป็น 1 ทั้งหมด

วิธีการนี้จะช่วยเพิ่มข้อมูลให้ระบบสามารถเรียนรู้เพื่อเชื่อมโยงคำที่อยู่ห่างกันได้ โดยดูจากเลขเวกเตอร์ของกลุ่มคำอ้างอิงที่เหมือนกัน

ตารางที่ 4.3 ตัวอย่างการใส่ค่าเวกเตอร์เพื่อบอกกลุ่มของคำอ้างอิง

คำ	Coreference 1	Coreference 2
The	1	0
old	1	0
fisherman	1	0
…	…	…
his	1	1
wallet	0	1
…	…	…
It	0	1
was	0	0

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

4.2.2 การตอบแบบสองทาง (Bidirectional Answer)

วิธีการหาคำตอบโดยการหาตำแหน่งของคำตอบที่ได้กล่าวไว้ในหัวข้อ 3.2.3 มีข้อจำกัดที่หากการหาตำแหน่งเริ่มต้นผิดพลาด จะทำให้การหาตำแหน่งของคำสุดท้ายในคำตอบมีโอกาสสูงที่จะผิดพลาดตามไปด้วย ดังที่แสดงในรูป 4.4 จะเห็นได้ว่าเมื่อเลือกคำแรกเป็น “บูธ” จะทำให้การเลือกคำตอบที่ถูกต้องเป็นไปได้ยาก

ลินคอล์น	ถูก	ลอบสังหาร	…	จอห์น	วิลค์ส	บูธ	การ	ลอบสังหาร
						Start		End

รูป 4.4 การตอบแบบทางเดียวเมื่อทำนาย ตำแหน่งเริ่มต้นของคำตอบผิดจะทำให้ การทำนายตำแหน่งสุดท้ายมีโอกาสสูงที่จะทำนายผิดพลาดตามไปด้วย

การตอบแบบสองทางเป็นวิธีการเพิ่มชั้นที่ใช้สำหรับการหาคำตอบอีกหนึ่งชั้น โดยชั้นที่เพิ่มมาจะทำการหาตำแหน่งของคำตอบจากตำแหน่งสุดท้ายก่อน แล้วค่อยย้อนกลับมาหาตำแหน่งแรก หลังจากนั้นจึงค่อยนำผลลัพธ์ของการตอบทั้งไปข้างหน้าและย้อนกลับมาหาค่าเฉลี่ยเพื่อใช้สำหรับหาคำตอบสุดท้าย วิธีนี้สามารถลดความผิดพลาดที่อาจเกิดขึ้นจากการตอบแบบไปข้างหน้าเพียงอย่างเดียว

จากตารางที่ 4.4 ในแต่ละแถวจะแสดงถึงค่าความน่าจะเป็นของตำแหน่งเริ่มต้นของคำตอบที่ได้จากการผ่านฟังก์ชัน Softmax จะเห็นได้ว่าการตอบแบบสองทางจะช่วยลดความเสี่ยงจากการตอบผิดลงได้ด้วยการใช้ค่าเฉลี่ยของการตอบที่มาจากสองทิศทาง

ตารางที่ 4.4 ตัวอย่างการหาคำตอบสุดท้ายจากค่าเฉลี่ยความน่าจะเป็นของการตอบแบบไปข้างหน้าและแบบย้อนกลับ ช่องที่มีสีเข้มแสดงถึงค่าความน่าจะเป็นที่มากที่สุด

	ลินคอล์น	...	จอห์น	วิลค์ส	บูธ	การ	ลอบสังหาร
การตอบแบบไปข้างหน้า	0.01	...	0.42	0.10	0.47	0.09	0.04
การตอบแบบย้อนกลับ	0.02	...	0.65	0.12	0.18	0.12	0.01
ผลการรวมคำตอบโดยหาค่าเฉลี่ย	0.015	...	0.535	0.11	0.325	0.105	0.025

4.2.3 ฟังก์ชันต้นทุนจากความยาวของคำตอบ (Answer Length Loss Function)

งานวิจัยชิ้นนี้ได้มีการเพิ่มฟังก์ชันต้นทุนจากความยาวของคำตอบเข้าไปด้วยเพื่อให้ตัวเน็ตเวิร์กสามารถตอบคำถามได้กระชับมากขึ้น โดยใช้เป็นค่าเฉลี่ยความผิดพลาดกำลังสอง (MSE) ระหว่างความยาวคำตอบดังสมการ 4.1

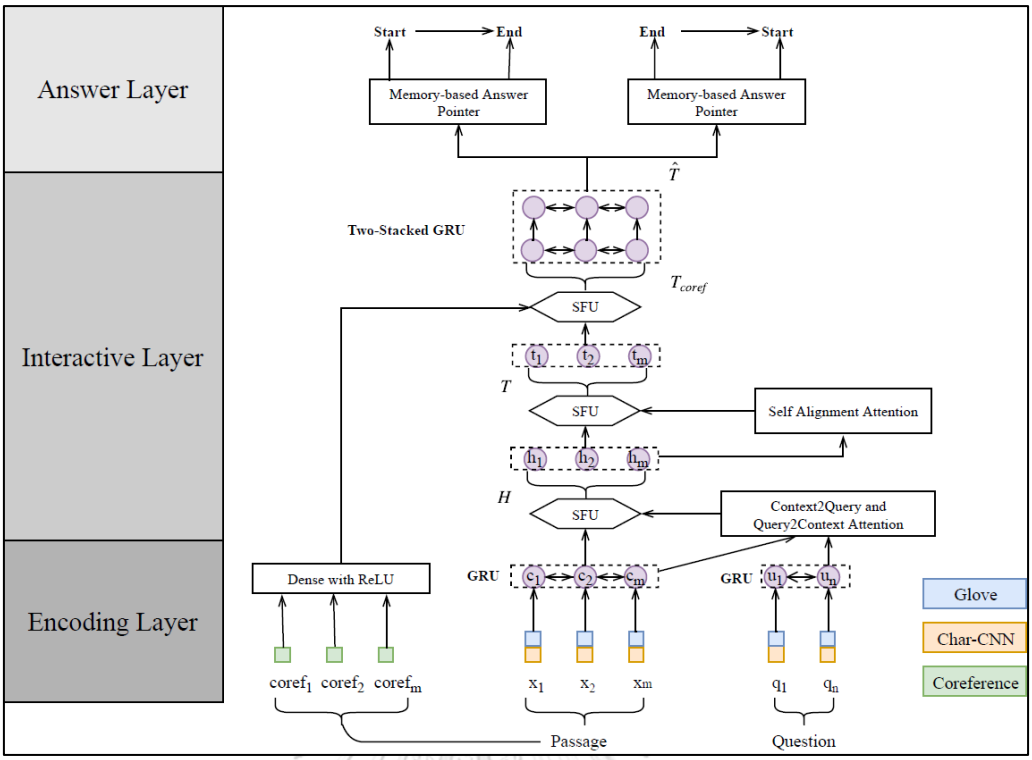
$$L_{length}(l_t, l_p) = MSE(l_t, l_p) \quad (4.1)$$

โดย l_t คือ ความยาวของคำตอบจริงที่ผ่านการนอร์มอลไลซ์แบบ min-max

l_p คือ ความยาวของคำตอบจากเน็ตเวิร์กที่ผ่านการนอร์มอลไลซ์แบบ min-max

4.3 เน็ตเวิร์กสำหรับการอ่านเพื่อตอบคำถามโดยใช้การเรียนรู้เชิงลึก

โครงสร้างเน็ตเวิร์กสำหรับการอ่านเพื่อตอบคำถามโดยใช้การเรียนรู้เชิงลึกสามารถแบ่งได้เป็น 3 ส่วนตามที่ได้กล่าวไปในบทที่ 3.2 ได้แก่ ส่วนการอ่านบทความ ส่วนการทำความเข้าใจบทความ และส่วนการตอบคำถาม โดยภาพรวมของเน็ตเวิร์กได้แสดงไว้แล้วในรูป 4.5

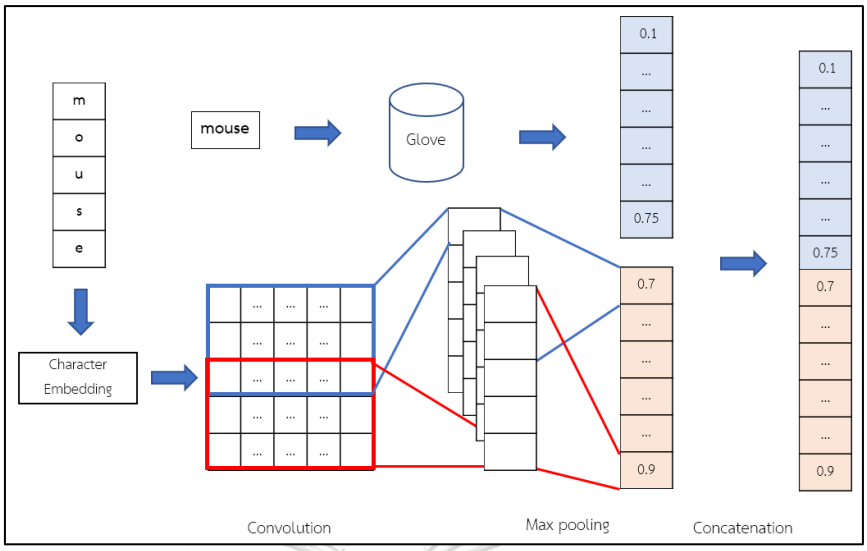


รูป 4.5 นิวรอลเน็ตเวิร์กสำหรับการอ่านเพื่อตอบคำถามที่ใช้ในการทดลอง

4.3.1 ส่วนการอ่านบทความ (Encoding Layer)

ข้อมูลของคำที่เข้ามาจะถูกเปลี่ยนให้อยู่ในรูปของเวกเตอร์ โดยการนำไปค้นหาในโกลฟ พร้อมทำการฝังคำระดับตัวอักษรด้วยการใช้เน็ตเวิร์กคอนโวลูชันแบบกว้าง ที่มีขนาดความกว้างของตัวกรองเท่ากับความยาวของเวกเตอร์ตัวอักษร แล้วนำเวกเตอร์ของคำที่ได้จากโกลฟและการฝังคำระดับตัวอักษรมาต่อกัน ดังแสดงในรูป 4.6 โดยหากคำใดที่ไม่อยู่ในโกลฟ ค่าเวกเตอร์จากโกลฟจะถูกแทนค่าด้วยเวกเตอร์ 0 ล้วน หรือแทนด้วยเวกเตอร์ 1 ล้วนหากค่านั้นเป็นตัวเลข

ทั้งนี้ข้อมูลขาเข้าที่เข้ามาได้แก่ เวกเตอร์ของบทความ $x_1 \ldots x_m$ ที่มีความยาว M เวกเตอร์ของคำถาม $q_1 \ldots q_n$ ที่มีความยาว N และเวกเตอร์คำอ้างอิง $coref_1 \ldots coref_m$ ที่มีขนาดเท่ากับกลุ่มของคำอ้างอิงทั้งหมด g เวกเตอร์ของบทความและคำถามจะถูกนำไปอ่านด้วย Bi-GRU ที่มีขนาดของชั้นลับเป็น $d/2$ ซึ่งจะทำให้ได้ผลลัพธ์จากการอ่านที่ออกมาเป็น $C \in \mathbb{R}^{dxM}$ และ $\mathrm{U} \in \mathbb{R}^{dxN}$ สำหรับบทความและคำถามตามลำดับตามสมการ 4.2 และสมการ 4.3



รูป 4.6 การสร้างเวกเตอร์ของคำ เพื่อใช้เป็นข้อมูลในการอ่าน

$$C = BiGRU(X) \quad (4.2)$$

$$U = BiGRU(Q) \quad (4.3)$$

ส่วนข้อมูลคำอ้างอิงจะถูกนำไปผ่านนิวรอลเน็ตเวิร์กแบบไปข้างหน้าที่มีจำนวนเพอร์เซปตรอน s ตัว โดยมีฟังก์ชันเรกทิไฟด์เชิงเส้น (ReLU) เป็นฟังก์ชันกระตุ้น ซึ่งจะทำให้ได้ผลลัพธ์เป็น $D \in \mathbb{R}^{sxM}$ ตามในสมการ 4.4

$$D = ReLU(W_d coref + b_d) \quad (4.4)$$

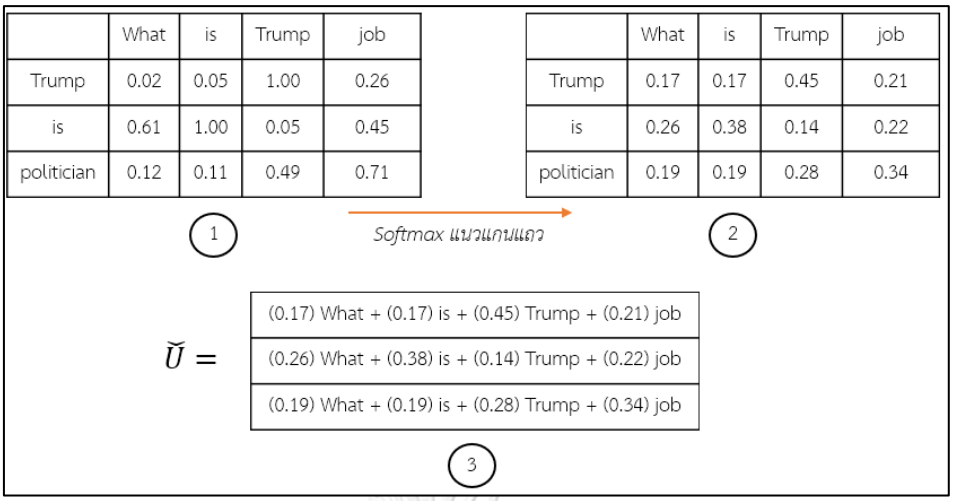
4.3.2 ส่วนการทำความเข้าใจบทความ (Interactive Layer)

ข้อมูลขาเข้าในขั้นตอนนี้จะมาจากส่วนการอ่านบทความ ได้แก่ C, U และ D โดยข้อมูลการอ่านที่ได้จากบทความและคำถามจะถูกนำไปหาความสนใจแบบควบ (Co-Attention) โดยใช้ฟังก์ชันการหาค่าความสนใจแบบคูณดังสมการ 4.5 จะทำให้ได้เมทริกความสนใจ $S \in \mathbb{R}^{MxN}$

$$S = C^T W_a U \quad (4.5)$$

หลังจากนั้นจึงจะหาความสนใจทั้งแบบ ความสนใจเมื่อมองคำถามต่อบทความ และความสนใจเมื่อมองบทความต่อคำถาม ดังที่กล่าวไปในบทที่ 3.2.2

การหาค่าความสนใจแบบมองคำถามต่อบทความ (Context-to-Query) จะนำเอาฟังก์ชัน Softmax มาใช้ในแนวแกนแถว $a_m = Softmax(S_{m:}) \in \mathbb{R}^N$ ทำให้ค่าในแนวแถวมีค่ารวมกันเป็น 1 หลังจากนั้นจึงนำค่าความสนใจที่บอกว่าคำไหนในคำถามมีความสนใจต่อคำที่ m ในบทความ ไปคูณกลับเข้าไปในเวกเตอร์ของคำถาม $\breve{U}_{:m} = \sum_n a_{mn} U_{:n}$ จะทำให้ได้ $\breve{U} \in \mathbb{R}^{dxM}$ โดยมีตัวอย่างดังรูป 4.7



รูป 4.7 ตัวอย่างการหาค่าความสนใจแบบมองคำถามต่อบทความ

การหาค่าความสนใจแบบมองบทความต่อคำถาม (Query-to-Context) ต้องการที่จะหาว่าคำในบทความคำใดบ้างที่สำคัญต่อคำถามทำได้โดย $b = Softmax(max_{col}(S)) \in \mathbb{R}^M$ เมื่อ max_{col} เป็นฟังก์ชันการหาค่าที่มากที่สุดตามแกนแถว หลังจากนั้นจึงนำค่าความสนใจที่ได้คูณกลับไปใน $\check{C}_{:m} = b_m * C_{:m}$ จะทำให้ได้ $\check{C} \in \mathbb{R}^{dxM}$ โดยมีตัวอย่างดังรูป 4.8

เวกเตอร์ที่ได้จะถูกนำไปขยายด้วยวิธีการฮิวริสติกตามที่ได้ใช้กันในเครื่องการอ่านด้วยการจำ (Mnemonic Reader) ดังสมการ 4.6 จะได้ค่า $G \in \mathbb{R}^{(5d+1)xM}$ หลังจากนั้น G จะถูกนำไปรวมกับข้อมูลการอ่านบทความ C โดยใช้หน่วยรวมความหมาย (SFU) ที่ได้กล่าวไปแล้วในบทที่ 2.3.4 เพื่อเลือกเฉพาะข้อมูลที่สำคัญไปใช้งานต่อตามสมการ 4.7 ซึ่งจะได้ผลลัพธ์เป็น $H \in \mathbb{R}^{dxM}$

$$G = [C;\ \check{U};\ C - \check{U};\ C \circ \check{U};\ \check{C};\ b\ ] \tag{4.6}$$

$$H = SFU_H(C, G) \tag{4.7}$$

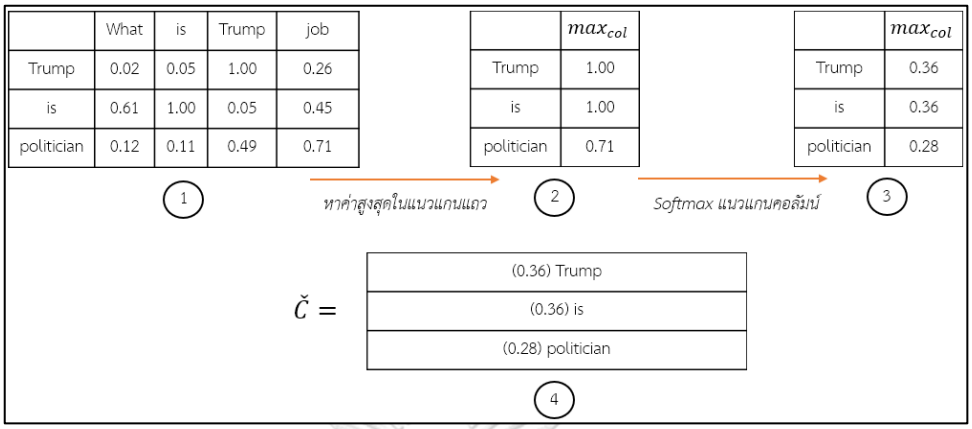
ผลลัพธ์ที่ได้จะถูกส่งต่อไปยังชั้นสำหรับหาความสนใจจากตัวเอง ด้วยวิธีการเดียวกับการหาค่าความสนใจแบบมองคำถามต่อบทความ แต่ใช้เพียง H ในการสร้างเมทริกซ์ความใกล้ชิดตามสมการ 4.8 จะได้ $S^H \in \mathbb{R}^{MxM}$

$$S^H = H^T W_b H \tag{4.8}$$

เช่นเดียวกับการหาค่าความสนใจแบบมองคำถามต่อบทความที่เคยทำไปแล้ว ก่อนอื่นเราจะนำฟังก์ชัน Softmax มาใช้ในแนวแกนแถว $a_m = Softmax({S^H}_{m:}) \in \mathbb{R}^m$ หลังจากนั้นจึงนำค่าความสนใจคูณกลับเข้าไป $\check{H}_{:m} = \sum_m a_{mm} H_{:m}$ จะทำให้ได้ $\check{H} \in$

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

$\mathbb{R}^{dxM}$ โดยผลลัพธ์จะถูกนำไปขยายด้วยวิธีการฮิวริสติกและรวมเข้ากับข้อมูลตั้งต้นตามสมการ 4.9 และสมการ 4.10 โดยจะมีผลลัพธ์เป็น $T \in \mathbb{R}^{dxM}$



รูป 4.8 ตัวอย่างการหาค่าความสนใจแบบมองบทความต่อคำถาม

$$\check{G} = [H;\ \check{H};\ H - \check{H};\ H \circ \check{H}\ ] \quad (4.9)$$

$$T = SFU_T(H, \check{G}) \quad (4.10)$$

ผลลัพธ์ T ที่ได้จะถูกนำไปต่อรวมเข้ากับเวกเตอร์คำอ้างอิง D ที่ถูกส่งมาจากชั้นการอ่านบทความ โดยใช้หน่วยรวมความหมายตามสมการด้านล่างจะได้ $T_{coref} \in \mathbb{R}^{dxM}$

$$\check{D} = [T;\ D\ ] \quad (4.11)$$

$$T_{coref} = SFU_T(H, \check{D}) \quad (4.12)$$

ในขั้นตอนสุดท้าย T_{coref} ซึ่งเป็นเวกเตอร์ของบทความที่ใส่ความสนใจทั้งหมดเข้าไปแล้ว จะถูกนำไปอ่านซ้ำอีกครั้งด้วย Bi-GRU แบบสองชั้น เพื่อให้ได้เวกเตอร์ $\hat{T} \in \mathbb{R}^{dxM}$ **ส่วนการตอบคำถาม (Answer Layer)**

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

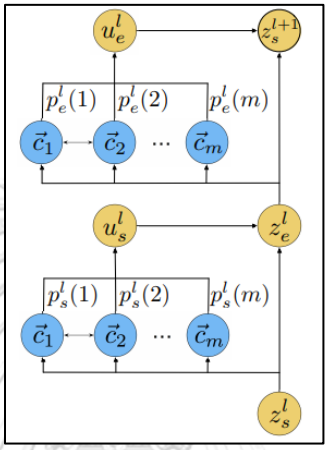
วิธีการตอบคำถามจะใช้ Mnemonic Pointing Layer จากเครื่องอ่านด้วยการจำ ซึ่งมีจุดเด่นที่การใช้หน่วยความจำมาช่วยให้สามารถเชื่อมโยงการตอบคำถามที่มีหลายความสัมพันธ์ได้ดีขึ้น ทั้งนี้ขั้นตอนในการตอบคำถามจะแบ่งเป็น 2 ขั้นตอนด้วยกัน คือการหาตำแหน่งของคำแรกของคำตอบ และการหาตำแหน่งของคำสุดท้ายในคำตอบ

ข้อมูลขาเข้าที่ได้รับมาในส่วนนี้คือ $\hat{T}$ และเวกเตอร์ในตำแหน่งสุดท้ายจากของคำถาม U_m ที่จะถูกใช้เป็นหน่วยความจำเริ่มต้น $z_s = U_m \in \mathbb{R}^d$ โดยการหาตำแหน่งเริ่มต้นจะทำได้โดยการใช้ นิวรอลเน็ตเวิร์กแบบไปข้างหน้าที่มีฟังก์ชัน tanh เป็นฟังก์ชันกระตุ้น ตามสมการ 4.13 หลังจากนั้นจึงนำผลลัพธ์ไปคูณกับเวกเตอร์ของน้ำหนัก $w_s \in \mathbb{R}^d$ แล้วนำไปหาค่าความน่าจะเป็นของตำแหน่งแรกด้วยฟังก์ชัน Softmax ดังในสมการ 4.14

$$s_i = tanh(W_{sff}[\hat{T}_i; z_s; \hat{T}_i \circ z_s] + b_{sff}) \quad (4.13)$$

$$p_s(i) = softmax(w_s s_i) \quad (4.14)$$

หลังจากนั้นการหาตำแหน่งสุดท้ายของคำตอบ จะทำได้โดยการอัพเดทหน่วยความจำด้วยความน่าจะเป็นในการหาตำแหน่งแรกของคำตอบโดย $z_e = SFU(z_s, \hat{T} \cdot p_s)$ ดังแสดงในรูป 4.9 แล้วนำค่าหน่วยความจำใหม่ที่ได้ไปแทน z_s ในสมการ 4.13 กับสมการ 4.14 ก็จะได้ค่าความน่าจะเป็นของตำแหน่งสุดท้ายของคำตอบ p_e ตามสมการ 4.15 และ สมการ 4.16



รูป 4.9 วิธีการอัพเดทหน่วยความจำ [16]

สำหรับการตอบแบบย้อนกลับเพื่อนำไปใช้ในการตอบแบบสองทิศทาง สามารถทำได้คล้ายกับการตอบแบบไปข้างหน้าที่ได้ทำไปแล้ว เพียงแต่จะหาตำแหน่งของคำสุดท้ายก่อน แล้วค่อยนำความน่าจะเป็นที่ได้อัพเดทหน่วยความจำ เพื่อย้อนกลับไปหาตำแหน่งแรกของคำตอบแทน

ในการหาคำตอบสุดท้าย สามารถทำได้โดยการนำเอาความน่าจะเป็นของการตอบแบบไปข้างหน้าและการตอบแบบย้อนกลับมาหาค่าเฉลี่ยกันแล้วค่อยนำไปเลือกตำแหน่งของคำตอบ

4.3.4 ฟังก์ชันต้นทุนที่ใช้ (Loss Function)

$$J(\theta) = -\frac{1}{N}\sum_i^N log\, p_s(y_i^s) + logp_e(y_i^e) + logp_s^r(y_i^s) + logp_e^r(y_i^e) + L_{length}(l_i, l_p) + L_{length}(l_i, l_{p^r}) \quad (4.15)$$

p_s, p_e คือ ค่าความน่าจะเป็นที่เน็ตเวิร์กให้ออกมาจากการตอบแบบหน้าไปหลัง

p_s^r, p_e^r คือ ค่าความน่าจะเป็นที่เน็ตเวิร์กให้ออกมาจากการตอบแบบหลังมาหน้า

y_i^s, y_i^e คือ ตำแหน่งจริงของคำตอบคำแรกและคำสุดท้าย ตัวที่ i

l_i คือ ความยาวจริงของคำตอบ

l_p, l_{p^r} คือ ความยาวของคำตอบที่เน็ตเวิร์กให้ออกมาจากแบบหน้าไปหลัง และแบบหลังมาหน้า

บทที่ 5
การทดลองและผลการทดลอง

5.1 ชุดข้อมูลที่ใช้ในการทดลอง

ชุดข้อมูลที่ใช้ในการทดลองมีด้วยกัน 2 ชุดด้วยกัน คือ ชุดข้อมูลสควอด และชุดข้อมูลสัพเพเหระ

5.1.1 ชุดข้อมูลสวดอด (Stanford Question Answering Dataset; SQuAD)

Pranav Rajpurkar และคณะ [6] ได้เสนอชุดคำถามจากการอ่านบทความชื่อว่าสควอดออกมาในช่วงปลายปี 2016 โดยสควอดเป็นข้อมูลขนาดใหญ่ชุดแรกที่สร้างขึ้นจากการถาม-ตอบจากมนุษย์เจ้าของภาษาจริง ๆ ทำให้สามารถทดสอบการอ่านทำความเข้าใจได้ดีกว่าชุดข้อมูลอื่น ๆ ก่อนหน้า

ชุดข้อมูลสควอดถูกทำขึ้นโดยสุ่มข้อมูลมาจากบทความในวิกิพีเดียภาษาอังกฤษที่ได้รับความนิยมหนึ่งหมื่นลำดับแรกมาเป็นจำนวน 536 บทความ หลังจากนั้นจึงนำบทความเหล่านั้นไปให้มนุษย์ทำการถามตอบผ่านทาง Amazon Mechanical Turk ซึ่งเป็นเว็ปสำหรับการทำคราวซอสซิ่ง (Crow Sourcing) ทำให้ได้ข้อมูลมาดังรูป 5.1

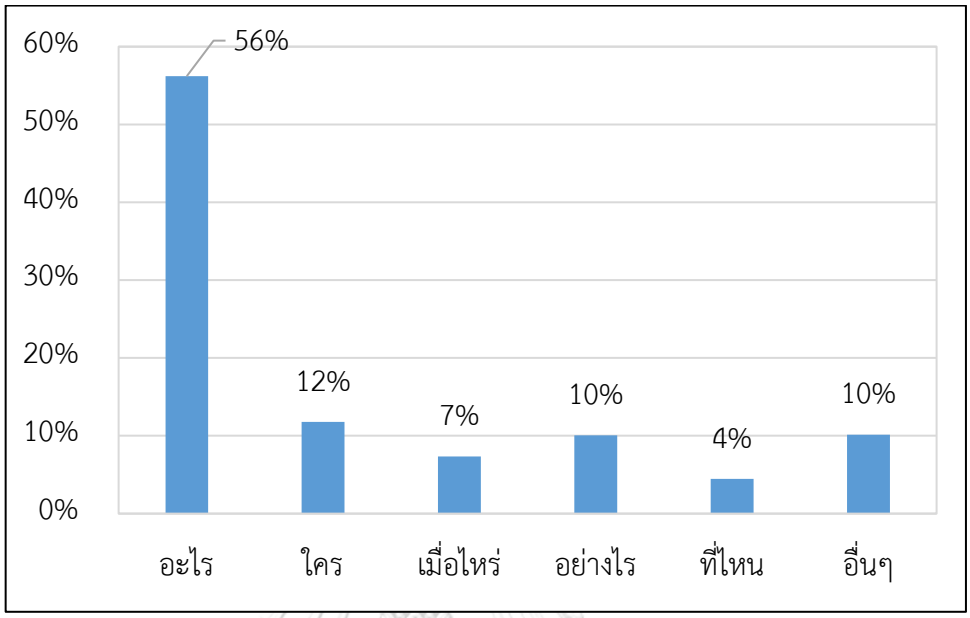
ชุดข้อมูลสควอดถูกแบ่งออกเป็นสามส่วนด้วยกันคือ ส่วนการเรียนรู้ (Training set) เป็นจำนวน 87,599 คำถาม ส่วนการพัฒนา (Development Set) เป็นจำนวน 10,570 คำถาม และส่วนการทดสอบ (Test Set) ซึ่งถูกเก็บเป็นความลับ แต่มีจำนวนที่ใกล้เคียงกับส่วนการพัฒนา

บทความ : At the end of this speech, Luther raised his arm "in the traditional salute of a knight winning a bout." Michael Mullett considers this speech as a "world classic of epoch-making oratory."
คำถาม : What did Luther do at the end of his speech?
คำตอบ : 9 11 (raised his arm)

รูป 5.1 ตัวอย่างข้อมูลสควอด ข้อมูลขาเข้าคือบทความและคำถาม โดยมีผลลัพธ์เป็นตำแหน่งของคำตอบจากบทความ

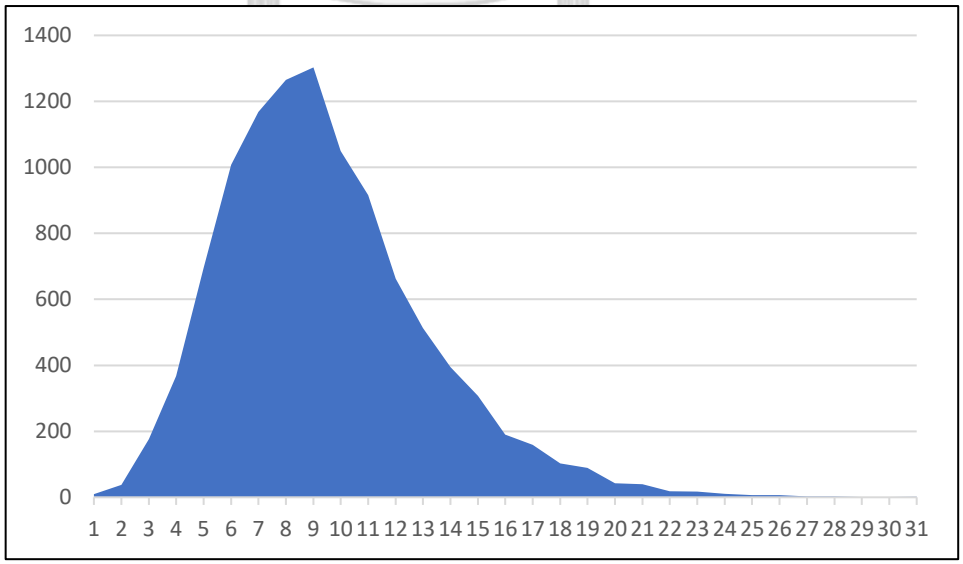
ต่อไปจะเป็นการดูสถิติของข้อมูลโดยอ้างอิงจากชุดข้อมูลพัฒนาบนสควอด โดยจะแบ่งประเภทของคำถามในชุดข้อมูลสควอดออกเป็น คำถามกลุ่มอะไร (What) คำถามกลุ่มใคร (Who) คำถามกลุ่มเมื่อไหร่ (When) คำถามกลุ่มอย่างไร (How) คำถามกลุ่มที่ไหน (Where) และคำถาม

กลุ่มอื่น ๆ (Others) รวม 6 ประเภทด้วยกัน โดยที่คำถามกลุ่มอะไรจะมีมากที่สุด (56%) และมีกลุ่มคำถามเกี่ยวกับสถานที่น้อยที่สุด (4%) สัดส่วนดังกล่าวแสดงอยู่ในรูป 5.2



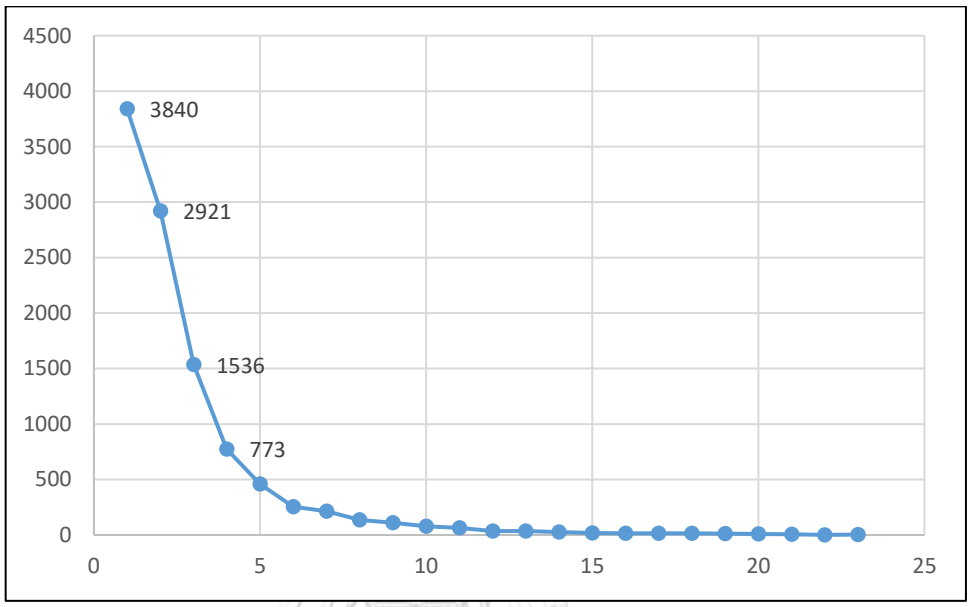
รูป 5.2 สัดส่วน (%) ประเภทคำถามบนชุดข้อมูลพัฒนาสควอด

ความยาวของคำถามในชุดข้อมูลสควอดมีตั้งแต่ 3 ถึง 34 คำ โดยคำถามส่วนมากจะมีความยาวในช่วง 5 ถึง 12 คำ และมีความยาวเฉลี่ยที่ 11.42 คำ ดังที่แสดงในรูป 5.3



รูป 5.3 จำนวนต่อความยาวคำถามบนชุดข้อมูลพัฒนาสควอด

ความยาวของคำตอบในชุดข้อมูลสควอด มีตั้งแต่ 1 ถึง 23 คำ โดย คำตอบส่วนมากจะมีความยาวอยู่ในช่วง 1 ถึง 3 คำ ดังที่แสดงในรูป 5.4 ปริมาณของคำตอบที่มีความยาวเป็น 1 มีมากที่สุดอยู่ที่ 36% ของคำตอบทั้งหมด



รูป 5.4 จำนวนต่อความยาวคำตอบบนชุดข้อมูลพัฒนาสควอด

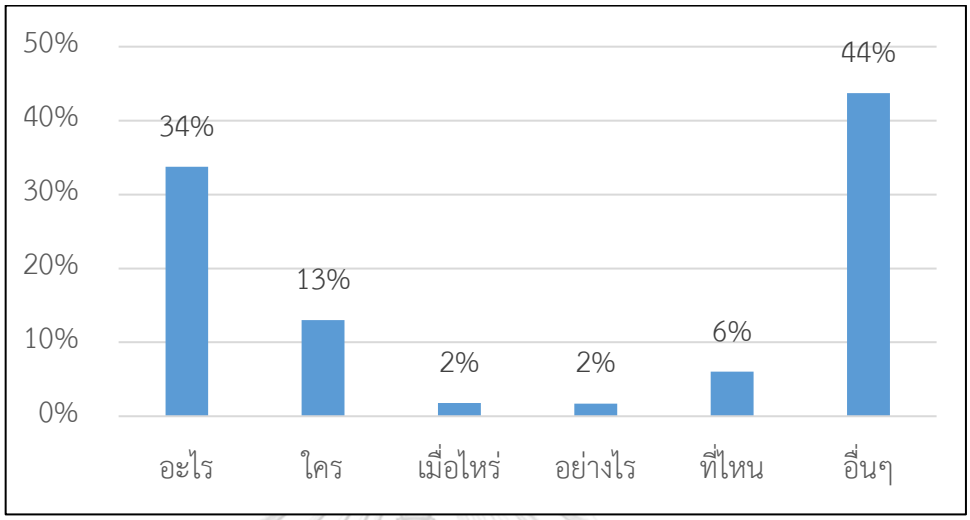
5.1.2 ชุดข้อมูลสัพเพเหระ (TriviaQA)

ในปี 2017 Mandar Joshi และคณะ [5] เล็งเห็นว่าชุดคำถามสควอดนั้นมีไบแอสที่เกิดจากมนุษย์ เนื่องจากคนที่ถามคำถามจะได้เห็นบทความที่จะนำมาใช้เพื่อตอบคำถาม ทำให้คำถามและคำตอบมีความคล้ายกันมาก ดังนั้น Mandar Joshi และคณะจึงเสนอชุดข้อมูลใหม่ที่ใช้ข้อมูลคำถามสัพเพเหระ (Trivial Quiz) จากเว็บไซต์ต่าง ๆ มาใช้ แล้วค่อยค้นหาบทความมาประกอบด้วยเครื่องมือค้นหาในภายหลัง ทำให้ได้ชุดข้อมูลที่มีไบแอสน้อยกว่าสควอด

ข้อมูลสัพเพเหระแบ่งได้ออกเป็นสองกลุ่มใหญ่คือ กลุ่มบทความจากวิกิพีเดีย และกลุ่มบทความจากอินเตอร์เน็ต สำหรับในงานวิจัยนี้ได้เลือกกลุ่มบทความจากวิกิพีเดียมาใช้ โดยจะแบ่งชุดข้อมูลได้เป็น ส่วนการเรียนรู้เป็นจำนวน 110,648 บทความ ส่วนการพัฒนาเป็นจำนวน 14,229 บทความ และส่วนการทดสอบที่ถูกซ่อนไว้เป็นจำนวน 13,661 บทความ

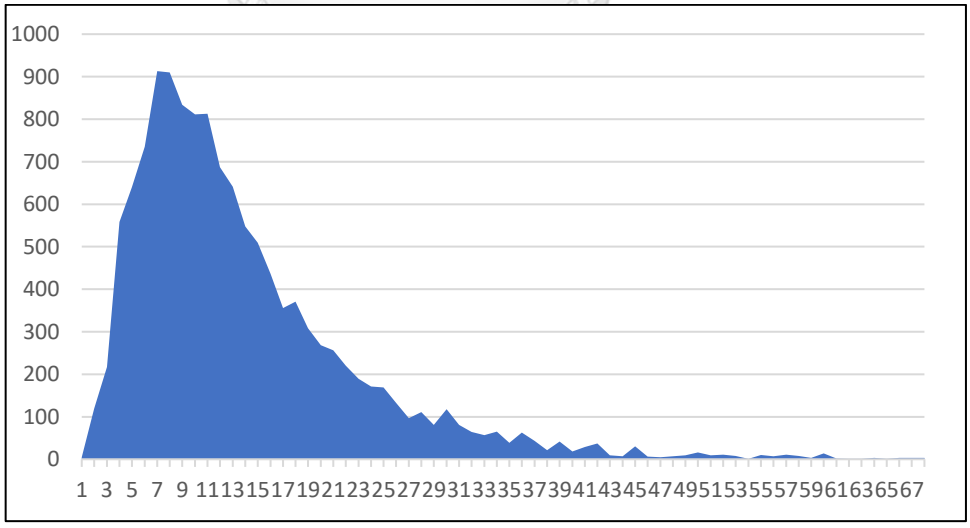
คำถามในข้อมูลสัพเพเหระชุดพัฒนาสามารถแบ่งออกได้เป็นหลายประเภท เช่นเดียวกับในชุดคำถามสควอด ทั้งนี้คำถามในข้อมูลสัพเพเหระจะมีกลุ่มคำถามประเภทอื่น ๆ ที่สูงที่สุด เนื่องจากลักษณะของคำถามมีความหลากหลายและยากต่อการจำแนก เช่น ‘Which Lloyd Webber musical premiered in the US on 10th December 1993 ?’ ประเภทคำถามถัดมาที่มี

มากที่สุดคือคำถามกลุ่มอะไร (34%) โดยคำถามกลุ่มที่มีน้อยที่สุด ได้แก่คำถามกลุ่มเมื่อไหร่และคำถามกลุ่มอย่างไร ดังที่แสดงไว้ในรูป 5.5



รูป 5.5 สัดส่วน (%) ประเภทคำถามบนชุดข้อมูลสัพเพเหระชุดพัฒนา

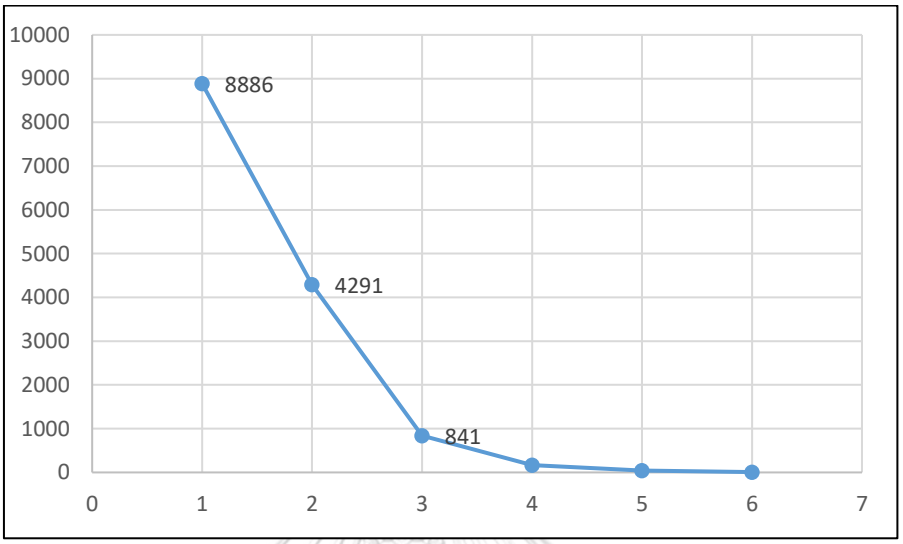
ความยาวของคำถามในชุดข้อมูลพัฒนาบนชุดข้อมูลสัพเพเหระ มีความยาวตั้งแต่ 5 ถึง 15 คำ โดยคำถามส่วนมากจะมีความยาวในช่วง 8 ถึง 18 คำ ดังที่แสดงในรูป 5.6 ซึ่งเมื่อเปรียบเทียบกับชุดข้อมูลสควอดแล้วจะมีความยาวที่มากกว่า



รูป 5.6 จำนวนต่อความยาวคำถามบนชุดข้อมูลสัพเพเหระชุดพัฒนา

ความยาวของคำตอบในชุดข้อมูลพัฒนาบนชุดข้อมูลสัพเพเหระ มีตั้งแต่ 1 ถึง 6 คำ คำตอบโดยมากจะมีความยาวอยู่ในช่วง 1 ถึง 3 คำ ดังที่แสดงในรูป 5.7 สังเกตเห็นได้ชัดว่าความยาว

คำตอบของชุดข้อมูลสัพเพเหระจะสั้นกว่าชุดข้อมูลสควอดมาก โดยคำตอบที่มีความยาวเป็น 1 มีมากถึง 62%



รูป 5.7 จำนวนต่อความยาวคำตอบบนชุดข้อมูลสัพเพเหระชุดพัฒนา

5.1.3 การแบ่งคำถามที่มีหลายความสัมพันธ์

จากแนวคิดที่ได้กล่าวไว้ใน 2.1 เราสามารถที่จะแบ่งกลุ่มของคำถามประเภทที่มีหลายความสัมพันธ์ออกมาได้ โดยดูว่าคำสำคัญในคำถามมีการเชื่อมไปยังคำในประโยคที่มีคำตอบอยู่หรือไม่ (multiple sentences) อีกวิธีหนึ่งคือดูว่าคำสำคัญในคำถามอยู่ห่างจากคำตอบเกินกว่าเกณฑ์ที่กำหนดหรือไม่ (keyword-answer distance; KA distance)

ในการแบ่งคำถามที่มีหลายความสัมพันธ์ เราจะตัดคำที่ไม่เกี่ยวข้อง (stop words) ออกจากคำถามเพื่อให้เหลือแต่คำสำคัญ แล้วค่อยนำไปคัดแยกประเภทคำถาม ทั้งนี้การแบ่งโดยใช้ระยะห่างระหว่างคำสำคัญกับคำตอบ จะถือว่าระยะห่างคือคำตรงกลางของคำตอบไปยังคำสำคัญที่ใกล้ที่สุด ซึ่งในที่นี้ถ้าหากระยะห่างมีค่าเกินกว่า 8 คำ เราจะให้คำถามนั้นเป็นประเภทมีหลายความสัมพันธ์

ตารางที่ 5.1 สัดส่วนคำถามที่มีหลายความสัมพันธ์บนชุดข้อมูลทดสอบ

	แบ่งโดยดูจากการเชื่อมประโยค	แบ่งโดยระยะห่างระหว่างคำสำคัญกับคำตอบ	สัดส่วนคำถามที่มีหลายความสัมพันธ์จากตารางที่ 1.1
ชุดข้อมูลสควอด	2.7%	13.5%	14%
ชุดข้อมูลสัพเพเหระ	17.2%	36.5%	40%

5.2 ระบบที่ใช้ทดลอง

ในส่วนนี้จะอธิบายถึงสภาพแวดล้อมที่ใช้ในการทดลอง ได้แก่ การแบ่งชุดข้อมูล การตั้งค่าเริ่มต้นต่าง ๆ และวิธีการสอนนิวรอลเน็ตเวิร์กเชิงลึก

ชุดข้อมูลจากสควอดและชุดข้อมูลสัพเพเหระ เราจะนำมาใช้เฉพาะชุดข้อมูลฝึกสอน (Training Set) และ ชุดข้อมูลพัฒนา (Development Set) เท่านั้น เนื่องจากชุดข้อมูลทดสอบถูกซ่อนไว้ทำให้ไม่สามารถนำเอามาใช้เองได้ ทั้งนี้เราจะแบ่งชุดข้อมูลฝึกสอนที่ได้ออกเป็น 2 ส่วน ได้แก่ ชุดข้อมูลฝึกสอนกับชุดข้อมูลตรวจสอบ (Validation Set) โดยจะใช้ชุดข้อมูลพัฒนาเป็นชุดข้อมูลสำหรับทดสอบ (Test Set) แทน โดยการแบ่งชุดข้อมูลได้ใช้วิธีการแบ่งตามประเภทของคำถาม (Stratify Sampling) ที่กล่าวไว้ในบทที่ 5.1

การตั้งค่าเริ่มต้นเราได้ใช้ Glove 840B Common Crawl ที่มีเวกเตอร์ของคำขนาด 300 มิติ สำหรับเน็ตเวิร์กคอนโวลูชันที่นำมาใช้ฝังคำในระดับอักขระ จะทำกับเวกเตอร์อักขระขนาด 100 มิติ จะมีจำนวนตัวกรองขนาด 5x100 อยู่ 100 ตัว สำหรับขนาดของชั้นลับที่ใช้ใน Bi-GRU ทุกตัวจะอยู่ที่ 150 และมีการใช้ดรอปเอาท์โดยตั้งค่าไว้ที่ 0.2 ในทุกชั้นของ Bi-GRU และนิวรอลเน็ตเวิร์กแบบไปข้างหน้า

ระบบมีการนำเข้าข้อมูลสำหรับฝึกสอนพร้อมกัน 64 ชุด (batch size) อัตราการเรียนรู้ (learning rate) เริ่มต้นที่ 0.0004 ในขณะสอนหากค่าเอฟวันบนชุดข้อมูลตรวจสอบไม่ดีขึ้นจะมีการหารค่าอัตราการเรียนรู้ออกครึ่งหนึ่งทุก 5 รอบการสอน (Epoch) ในการฝึกสอนจะใช้ Adam เป็นตัวช่วยปรับค่าเกรเดียนในการเรียนรู้ โดยผลลัพธ์สุดท้ายที่รายงานบนข้อมูลทดสอบจะเลือกโดยดูจากโมเดลที่มีค่าเอฟวันสูงสุดบนชุดข้อมูลตรวจสอบ

ตารางที่ 5.2 ปริมาณข้อมูลคำถาม-บทความในแต่ละชุดข้อมูลที่ใช้ในการทดลอง

	ชุดข้อมูลฝึกสอน (Training Set)	ชุดข้อมูลตรวจสอบ (Validation Set)	ชุดข้อมูลทดสอบ (Test Set)
ชุดข้อมูลสควอด	80,000	7,599	10,570
ชุดข้อมูลสัพเพเหระ	97,951	12,697	14,229

5.3 ผลการทดลอง

ในส่วนนี้จะรายงานการทดลองและทำการวิเคราะห์เปรียบเทียบโมเดลและวิธีการที่เสนอไปในบทที่ 4.2 ได้แก่ (1) การใช้เวกเตอร์คำอ้างอิง (Coreference Vector) (2) การตอบคำถามแบบสองทิศทาง (Bidirectional Answer) (3) ฟังก์ชันต้นทุนจากความยาวของคำตอบ (Answer Length Loss Function) ทั้งนี้รายการของโมเดลที่ใช้ในงานทดลองได้มีการอธิบายไว้ในตารางที่ 5.3

ตารางที่ 5.3 อธิบายโมเดลที่นำมาใช้ในการทดลอง

โมเดล	คำอธิบาย
Mnemonic Reader	โมเดลการอ่านด้วยการจำ เป็นโมเดลมาตรฐานที่ใช้สภาพแวดล้อมเดียวกับโมเดลของเราในการทดลอง โดยนำมาจาก[1] ซึ่งให้ผลลัพธ์ที่ใกล้เคียงกับที่รายงานในงานวิจัยต้นฉบับ
Base Model	โมเดลพื้นฐานที่ใช้ที่ได้อธิบายไปในตารางที่ 3.1 โดยยังไม่ได้มีการนำวิธีที่เสนอในบทที่ 4.2 ไปใช้งาน
Full Model [All proposed methods]	โมเดลของเรา ที่ได้อธิบายไปในบทที่ 4.3 ที่นำทุกวิธีการในบทที่ 4.2 ไปใช้งาน
Full Model w/o Coreferece Vector [All proposed methods – (1)]	Full Model ที่ตัดเวกเตอร์อ้างอิงออกไป
Full Model w/o Bidirectional Answer [All proposed methods – (2)]	Full Model ที่ตัดวิธีการตอบแบบสองทางออกไป
Full Model w/o Length Loss Function [All proposed methods – (3)]	Full Model ที่ไม่ได้ใช้ค่าฟังก์ชันต้นทุนจากความยาวของคำตอบมาใช้ในการสอน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

5.3.1 เปรียบเทียบผลการทดลองโดยรวม

การทดลองนี้ จะทำการเปรียบเทียบโมเดลที่ได้ทำการเสนอขึ้นมากับโมเดลที่มีอยู่ก่อนหน้าคือโมเดลการอ่านด้วยการจำ โดยจะทำการเปรียบเทียบโดยใช้โมเดลพื้นฐานและโมเดลที่ใช้ทุกวิธีการวิธีการที่นำเสนอ หลังจากนั้นจะทำการวิเคราะห์เปรียบเทียบวิธีการที่เสนอแบบแยกเป็นส่วนประกอบ โดยจะทำการตัดออกทีละส่วน (Ablation Test)

จากผลการทดลองในตารางที่ 5.4 ผลปรากฏว่าโมเดลที่ใช้ทุกวิธีการที่เสนอนั้น ให้ประสิทธิภาพเอฟวันและความถูกต้อง (Exact Match) ที่ดีที่สุดบนทั้งสองชุดข้อมูล โดยมีค่าเอฟวันเพิ่มขึ้นเป็น 0.7 และ 3.1 เมื่อเทียบกับโมเดลพื้นฐาน สำหรับชุดข้อมูลสควอดและชุดข้อมูลสัพเพเหระตามลำดับ จะพบว่าวิธีการที่นำเสนอนั้นสามารถเพิ่มประสิทธิภาพบนชุดข้อมูลสัพเพเหระได้มากกว่าบนสควอด

[1] https://github.com/HKUST-KnowComp/MnemonicReader

ตารางที่ 5.4 แสดงผลการทดลองในภาพรวม

โมเดล	ชุดข้อมูลสควอด		ชุดข้อมูลสัพเพเหระ	
	F1	EM	F1	EM
Mnemonic Reader	76.8	67.4	66.2	60.9
Base Model	76.9	66.9	66.3	61.8
Full Model	**77.6**	**67.8**	**69.4**	**64.0**

ผลการทดลองจากตารางที่ 5.5 แสดงให้เห็นว่า วิธีการตอบแบบสองทางสามารถเพิ่มประสิทธิภาพได้ดีที่สุดในชุดข้อมูลสควอด โดยเมื่อตัดวิธีการดังกล่าวออกทำให้ค่าเอฟวันลดลงถึง 0.5 และค่าความถูกต้องลดลง 0.6 จุดที่น่าสนใจอีกอย่างบนชุดข้อมูลสควอดคือ การตัดฟังก์ชันต้นทุนความยาวคำตอบออกทำให้ค่าความถูกต้องลดลงไปมากถึง 0.7 เมื่อเทียบกับค่าเอฟวันที่ลดไปเพียง 0.2 คาดว่าน่าจะเกิดจากการที่ฟังก์ชันต้นทุนดังกล่าวมีส่วนช่วยในการทำให้คำตอบกระชับ ทำให้มีความครบถ้วนถูกต้องขึ้น

บนชุดข้อมูลสัพเพเหระพบว่าวิธีการที่ดีที่สุดคือการเพิ่มเวกเตอร์อ้างอิง ซึ่งเมื่อตัดออกทำให้ค่าเอฟวันตกไป 1.0 ทั้งนี้วิธีที่เสนออีกสองวิธีเมื่อตัดออกไปแล้วไม่ได้ทำให้ค่าเอฟวันตกลงในทางกลับกันกลับทำให้ค่าความถูกต้องเพิ่มขึ้น คาดว่าสาเหตุน่าจะมาจากการที่ชุดข้อมูลสัพเพเหระมีสัดส่วนคำถามที่คำตอบมีเพียงคำเดียวมากถึง 62% ในขณะที่คำตอบที่มีคำเดียวในสควอดมีเพียง 36% ตามที่ได้ศึกษาไปในบทที่ 5.1.1 และ 5.1.2 ทำให้การตอบแบบสองทางและฟังก์ชันต้นทุนไม่มีผลมากเท่าบนชุดข้อมูลสควอด

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 5.5 แสดงผลการทดลองโดยตัดวิธีการที่นำเสนอออกทีละส่วนจากโมเดลที่สมบูรณ์

โมเดล	ชุดข้อมูลสควอด		ชุดข้อมูลสัพเพเหระ	
	F1	EM	F1	EM
Full Model	**77.6**	**67.8**	**69.4**	64.0
Full Model w/o Coreferece Vector	77.4	67.5	68.4	63.7
Full Model w/o Bidirectional Answer	77.1	67.2	**69.4**	**64.8**
Full Model w/o Length Loss Function	77.4	67.1	**69.4**	64.6

5.3.2 เปรียบเทียบผลการใส่เวกเตอร์อ้างอิงกับข้อมูลที่มีหลายความสัมพันธ์

ในการทดลองนี้จะดูผลของการใส่เวกเตอร์อ้างอิงกับคำถามที่มีหลายความสัมพันธ์ ว่าสามารถเพิ่มประสิทธิภาพได้จริงหรือไม่โดยจะดูจากค่าเอฟวันเป็นหลัก ผลการทดลองจะอยู่ใน ตารางที่ 5.6 และ ตารางที่ 5.7 ซึ่งใช้วิธีแบ่งโดยดูจากระยะห่างของคำสำคัญกับคำตอบ และแบ่งจากการเชื่อมประโยค ตามลำดับ โดยสามารถดูเกณฑ์ในการแบ่งประเภทและปริมาณสัดส่วนได้ในบทที่ 5.1.3

จากตารางที่ 5.6 พบว่าการเพิ่มเวกเตอร์อ้างอิงทำให้การตอบคำถามที่มีหลายความสัมพันธ์ทำได้ดีขึ้นทุกชุดข้อมูล โดยเฉพาะบนชุดข้อมูลสควอดที่เมื่อตัดเอาเวกเตอร์อ้างอิงออก พบว่า ค่าเอฟวันของคำถามที่มีหลายความสัมพันธ์จะตกไป 1.1 แต่ประสิทธิภาพบนคำถามที่มีความสัมพันธ์เดียวแทบจะไม่เปลี่ยนแปลง

ตารางที่ 5.6 แสดงค่าเอฟวันบนชุดข้อมูลสควอดและชุดข้อมูลสัพเพเหระ โดยแบ่งประเภทคำถามที่มีหลายความสัมพันธ์ด้วยระยะห่างระหว่างคำสำคัญกับคำตอบ

โมเดล	คำถามที่มีความสัมพันธ์เดียว		คำถามที่มีหลายความสัมพันธ์	
	ชุดข้อมูลสควอด	ชุดข้อมูลสัพเพเหระ	ชุดข้อมูลสควอด	ชุดข้อมูลสัพเพเหระ
Base Model	79.6	68.5	59.4	62.1
Full Model w/o Coreferece Vector	**80.3**	71.4	59.0	63.2
Full Model	**80.3**	**71.8**	**60.1**	**65.1**

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ในส่วนของตารางที่ 5.7 ซึ่งแบ่งคำถามจากการเชื่อมประโยค ผลปรากฏว่าการถอดเวกเตอร์อ้างอิงออกยังคงทำให้ค่าเอฟวันในคำถามกลุ่มที่มีหลายความสัมพันธ์ตกไปมาก โดยเอฟวันที่ลดลงเป็น 0.6 และ 1.4 บนชุดข้อมูลสควอดและชุดข้อมูลสัพเพเหระตามลำดับ

โดยภาพรวมแล้วการเพิ่มเวกเตอร์อ้างอิงสามารถช่วยเพิ่มประสิทธิภาพบนชุดข้อมูลที่มีหลายความสัมพันธ์ที่ใช้วิธีการแบ่งทั้งสองแบบ โดยเฉพาะบนชุดข้อมูลสัพเพเหระที่มีสัดส่วนของข้อมูลที่มีหลายความสัมพันธ์มากกว่าของชุดข้อมูลสควอด

ตารางที่ 5.7 แสดงค่า F1 บนชุดข้อมูลสควอดและชุดข้อมูลสัพเพเหระ โดยแบ่งประเภทคำถามที่มีหลายความสัมพันธ์จากการเชื่อมประโยค

โมเดล	คำถามที่มีความสัมพันธ์เดียว		คำถามที่มีหลายความสัมพันธ์	
	ชุดข้อมูลสควอด	ชุดข้อมูลสัพเพเหระ	ชุดข้อมูลสควอด	ชุดข้อมูลสัพเพเหระ
Base Model	77.7	67.1	47.7	62.6
Full Model w/o Coreferece Vector	78.2	69.6	47.1	62.6
Full Model	**78.4**	**70.5**	**47.7**	**64.0**

5.3.3 เปรียบเทียบผลการทดลองโดยความยาวของคำตอบ

การทดลองนี้ต้องการจะดูผลของการใช้ การตอบแบบสองทาง และฟังก์ชันต้นทุนความยาวคำตอบ โดยจะทำการเปรียบเทียบบนเฉพาะชุดข้อมูลสควอด เนื่องจากความยาวของคำตอบมีความหลากหลายมากกว่า ทำให้เห็นผลได้ชัดเจนกว่า

จากในตารางที่ 5.8 ผลปรากฏว่าการตอบแบบสองทาง และ ฟังก์ชันต้นทุนความยาวมีผลช่วยให้คำตอบกระชับมากขึ้น โดยเฉพาะเมื่อไม่ใช้ฟังก์ชันต้นทุนความยาวในขณะสอนโมเดล จะทำให้ความยาวเฉลี่ยของคำตอบเพิ่มขึ้น 0.12 คำ และความต่างเฉลี่ยเทียบกับผลเฉลย เพิ่มขึ้นถึง 0.26 หรือคิดเป็น 18% เทียบกับโมเดลที่ใช้ฟังก์ชันต้นทุนความยาว ส่วนการตอบแบบสองทางพบว่าไม่ได้ช่วยลดความยาวเฉลี่ยโดยตรง แต่ช่วยในเรื่องของการปรับให้คำตอบมีความยาวใกล้เคียงกับผลเฉลย

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 5.8 แสดงผลของความยาวของคำตอบ ความต่างหมายถึงค่าความต่างสัมบูรณ์ (Absolute Difference) ระหว่างความยาวคำตอบของโมเดลกับผลเฉลย

โมเดล	ชุดข้อมูลสควอด	
	ความยาวเฉลี่ยของคำตอบ	ความต่างเฉลี่ยของความยาวคำตอบ
Ground Truth	2.75	-
Base Model	3.58	1.73
Full Model w/o Length Loss Function	3.65	1.70
Full Model w/o Bidirectional Answer	3.56	1.64
Full Model	**3.53**	**1.44**

5.3.4 ผลการทดลองโดยดูจากนัยสำคัญทางสถิติ (Statistical Significance)

ในการทดลองนี้จะทำการทดสอบวิธีการที่เสนอว่าสามารถเพิ่มประสิทธิภาพให้โมเดลได้อย่างมีนัยสำคัญทางสถิติหรือไม่ ด้วยวิธีทดสอบ 2 วิธีการได้แก่วิธี การทดสอบคู่แบบที และ การทดสอบแม็คนมาร์

5.3.4.1 การทดสอบคู่แบบที (Paired t-Test)

การทดสอบนี้จะทำบนข้อมูลที่ทำการบูทสแตร็ป (Bootstrap) โดยสุ่มแบบแบ่งประเภท (Stratify Sampling) เฉพาะชุดข้อมูลที่ใช้ทดสอบเป็น 50 ชุดด้วยกัน แล้วนำผลการทดลองมาทำการทดสอบคู่แบบทีระหว่างโมเดลตัวเต็มกับโมเดลที่ตัดส่วนประกอบที่เสนอออก เพื่อหาค่าพี (P Value) โดยวิธีการทำบูทสแตร็ปบนข้อมูลทดสอบมีการใช้อยู่ในงานวิจัยเกี่ยวกับการเรียนรู้ด้วยคอมพิวเตอร์ [34-38] เนื่องจากข้อจำกัดในด้านเวลาที่ใช้สอน

จากตารางที่ 5.9 พบว่าบนชุดข้อมูลสควอดทุกวิธีการที่เสนอสามารถเพิ่มประสิทธิภาพได้อย่างมีนัยสำคัญ ส่วนบนชุดข้อมูลสัพเพเหระพบว่า ยกเว้นเฉพาะโมเดลที่ใช้ฟังก์ชันต้นทุนความยาว วิธีการอื่นที่เสนอสามารถเพิ่มประสิทธิภาพได้อย่างมีนัยสำคัญ

ตารางที่ 5.9 แสดงผลการทำ Paired t-Test กับข้อมูลทดสอบจากการทำบูทสแตร็ป 50 ชุด

โมเดลที่นำมาเปรียบเทียบกับ Full Model	**ชุดข้อมูลสควอด**	**ชุดข้อมูลสัพเพเหระ**
	ค่าพี (P Value)	**ค่าพี (P Value)**
Full Model w/o Coref	0.00018	**<.00001**
Full Model w/o Bidirectional Answer	**< .00001**	**<.00001**
Full Model w/o Length Loss Function	**< .00001**	0.14140

เมื่อดูตารางที่ 5.10 พบว่าโมเดลที่ใส่ทุกวิธีที่นำเสนอให้ค่าเอฟวันเฉลี่ยที่ดีที่สุดบนทุกชุดข้อมูล โดยวิธีการที่เพิ่มประสิทธิภาพได้มากที่สุดบนชุดข้อมูลสควอดคือการตอบแบบสองทาง ส่วนบนชุดข้อมูลสัพเพเหระการใส่เวกเตอร์อ้างอิงจะเพิ่มค่าเอฟวันเฉลี่ยได้ดีที่สุด

ตารางที่ 5.10 แสดงค่าเอฟวันเฉลี่ยและค่า SD บนชุดข้อมูลทดสอบที่ทำการบูทสแตร็ป

โมเดล	**ชุดข้อมูลสควอด**		**ชุดข้อมูลสัพเพเหระ**	
	Mean F1	**SD**	**Mean F1**	**SD**
Full Model	**77.5**	0.34	**69.4**	0.49
Full Model w/o Coreferece Vector	77.4	0.35	68.4	0.50
Full Model w/o Bidirectional Answer	77.1	0.37	69.0	0.47
Full Model w/o Length Loss Function	77.3	0.34	69.3	0.52

5.3.4.2 การทดสอบแม็คนมาร์ (Mcnemar Test)

การทดสอบนี้จะทำการเปรียบเทียบผลลัพธ์ระหว่าง โมเดลแบบเต็ม กับ โมเดลแบบเต็มที่ได้ทำการตัดส่วนประกอบแต่ละชิ้นออกไป ว่าผลลัพธ์คำตอบมีความแตกต่างกันหรือไม่ โดยจะวัดค่าความสำคัญทางสถิติโดยใช้ค่าพีบนชุดข้อมูลทดสอบ (ไม่ได้ทำการบูทสแตร๊ป)

จากตารางที่ 5.11 พบว่าบนชุดข้อมูลสควอดคำตอบที่ได้จากโมเดลแบบเต็มกับโมเดลแบบเต็มที่ตัดชิ้นส่วนต่าง ๆ ออก ไม่ได้มีความแตกต่างอย่างมีนัยสำคัญ โดยโมเดลแบบเต็มที่ตัดการตอบแบบสองทางออกมีค่าพีที่ดีที่สุดที่ 0.199 ส่วนการทดสอบบนชุดข้อมูลสัพเพเหระ โมเดลแบบเต็มที่ตัดการใช้เวกเตอร์อ้างอิงออก ให้ผลค่าพีที่ดีที่สุดคือ 0.054 ซึ่งเกือบจะแตกต่างอย่างมีนัยสำคัญทางสถิติ

ตารางที่ 5.11 แสดงผลการทดสอบแม็คนมาร์ของการตัดชิ้นส่วนที่นำเสนอออกจากโมเดล

โมเดลที่นำมาเปรียบเทียบกับ Full Model	**ชุดข้อมูลสควอด**	**ชุดข้อมูลสัพเพเหระ**
	ค่าพี (P Value)	**ค่าพี (P Value)**
Full Model w/o Coref	0.346	**0.054**
Full Model w/o Bidirectional Answer	**0.199**	0.475
Full Model w/o Length Loss Function	0.442	0.976

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

บทที่ 6
สรุปการวิจัยและแนวทางการวิจัยในขั้นถัดไป

6.1 สรุปผลการทดลอง

วิทยานิพนธ์ชิ้นนี้ ได้เสนอวิธีการเพิ่มประสิทธิภาพในการตอบคำถามของการอ่านด้วยเครื่องสามวิธีด้วยกันได้แก่ (1) การใช้เวกเตอร์ของคำอ้างอิงเพื่อเพิ่มประสิทธิภาพของคำถามในกลุ่มที่มีหลายความสัมพันธ์ (2) การตอบคำถามแบบสองทางเพื่อลดความผิดพลาดที่เกิดจากการตอบผิดในขั้นแรกของการตอบแบบทางเดียว (3) ฟังก์ชันต้นทุนจากความยาวของคำตอบที่ช่วยให้คำตอบกระชับ

จากการทดลองแล้วพบว่าวิธีการตอบแบบสองทางและฟังก์ชันต้นทุนจากความยาว ช่วยเพิ่มประสิทธิภาพบนชุดข้อมูลสควอดที่คำตอบยาวได้ดี ส่วนการใช้เวกเตอร์อ้างอิงจะเหมาะกับชุดข้อมูลสัพเพเหระที่มีคำถามประเภทหลายความสัมพันธ์เป็นจำนวนมาก โดยโมเดลที่ใช้วิธีการที่เสนอทั้งสามวิธีให้ค่าเอฟวัน (F1) และค่าความถูกต้อง (Exact Match) ที่ดีที่สุดบนทุกชุดข้อมูลที่ใช้ทดสอบโดยมีค่าเอฟวันเพิ่มขึ้นจาก Mnemonic Reader ซึ่งเป็นโมเดลมาตรฐาน อยู่ 1.0% และ 4.8% บนชุดข้อมูลสควอดและชุดข้อมูลสัพเพเหระตามลำดับ

นอกจากนี้ในวิทยานิพนธ์ฉบับนี้ยังได้มีการวิเคราะห์ชุดข้อมูลในหลายแง่มุม รวมทั้งมีการทดลองที่ประเมินผลของวิธีการที่นำเสนอแยกทีละส่วนประกอบ พร้อมทั้งวัดผลนัยสำคัญทางสถิติสองวิธี โดยผลการทดสอบแม็คนมาร์ (Mcnemar Test) พบว่านอกจากการใช้เวกเตอร์อ้างอิงบนชุดข้อมูลสัพเพเหระที่เกือบจะแตกต่างอย่างมีนัยสำคัญแล้ว วิธีการอื่นที่เสนอไม่ได้มีความแตกต่างอย่างมีนัยสำคัญ ส่วนผลการทดสอบด้วยการทดสอบคู่แบบที (Paired-t Test) พบว่าทุกวิธีที่นำเสนอสามารถเพิ่มประสิทธิภาพได้อย่างมีนัยสำคัญ ยกเว้นฟังก์ชันต้นทุนความยาวบนชุดข้อมูลสัพเพเหระ

6.2 แนวทางการวิจัยถัดไป

การสกัดคำอ้างอิงเป็นขั้นตอนที่สำคัญมากในการนำเอาคำอ้างอิงไปใช้งาน โดยหากสามารถสร้างโมเดลที่สามารถเรียนรู้การสกัดคำอ้างอิงไปพร้อมกับการหาคำตอบจะสามารถช่วยเพิ่มประสิทธิภาพของโมเดลได้

ทั้งนี้การสร้างเวกเตอร์จากคำอ้างอิงด้วยหลักการทางภาษาศาสตร์ให้เชื่อมโยงคำได้ดีขึ้นก็เป็นอีกหัวข้อที่มีความเป็นไปได้ที่จะเพิ่มประสิทธิภาพให้กับตัวโมเดลให้สูงยิ่งขึ้นได้

แนวทางการวิจัยที่น่าสนใจอย่างสุดท้ายคือการพัฒนาโมเดลให้สามารถเรียนรู้ที่จะตอบว่าไม่รู้ได้ ถ้าคำถามนั้นไม่สามารถตอบได้ด้วยเอกสารที่มีอยู่ ซึ่งจะมีประโยชน์มากต่อการนำไปใช้งานจริง

รายการอ้างอิง

1. Riloff, E. and M. Thelen. *A rule-based question answering system for reading comprehension tests*. in *Proceedings of the 2000 ANLP/NAACL Workshop on Reading comprehension tests Volume 6*. 2000. Association for Computational Linguistics.
2. Poon, H., et al. *Machine reading at the university of washington*. in *Proceedings of the NAACL HLT 2010* 2010. Association for Computational Linguistics.
3. Hermann, K.M., et al. *Teaching machines to read and comprehend*. in *Advances in Neural Information Processing Systems*. 2015.
4. Hill, F., et al., *The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations.* arXiv preprint arXiv:1511.02301, 2015.
5. Joshi, M., et al., *TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension.* arXiv preprint arXiv:1705.03551, 2017.
6. Rajpurkar, P., et al., *Squad: 100,000+ questions for machine comprehension of text.* arXiv preprint arXiv:1606.05250, 2016.
7. Weston, J., et al., *Towards ai-complete question answering: A set of prerequisite toy tasks.* arXiv preprint arXiv:1502.05698, 2015.
8. Peng, B., et al., *Towards neural network-based reasoning.* arXiv preprint arXiv:1508.05508, 2015.
9. Henaff, M., et al., *Tracking the world state with recurrent entity networks.* arXiv preprint arXiv:1612.03969, 2016.
10. Tretasayuth, N., P. Vateekul, and P. Boonkwan, *End-to-End Memory Network for QA with Multiple Relationships*, in *The 14th International Joint Conference on Computer Science and Software Engineering (JCSSE)*. 2017.
11. Collobert, R., et al., *Natural language processing (almost) from scratch.* Journal of Machine Learning Research, 2011. **12**(Aug): p. 2493-2537.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

12. Pennington, J., R. Socher, and C. Manning. *Glove: Global vectors for word representation*. in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014.
13. Kim, Y., *Convolutional neural networks for sentence classification.* arXiv preprint arXiv:1408.5882, 2014.
14. Srivastava, N., et al., *Dropout: A simple way to prevent neural networks from overfitting.* The Journal of Machine Learning Research, 2014. **15**(1): p. 1929-1958.
15. Gal, Y. and Z. Ghahramani. *A theoretically grounded application of dropout in recurrent neural networks*. in *Advances in neural information processing systems*. 2016.
16. Hu, M., Y. Peng, and X. Qiu, *Mnemonic Reader for Machine Comprehension.* arXiv preprint arXiv:1705.02798, 2017.
17. LeCun, Y., Y. Bengio, and G. Hinton, *Deep learning.* Nature, 2015. **521**(7553): p. 436-444.
18. Hochreiter, S. and J. Schmidhuber, *Long short-term memory.* Neural computation, 1997.
19. Cho, K., et al., *Learning phrase representations using RNN encoder-decoder for statistical machine translation.* arXiv preprint arXiv:1406.1078, 2014.
20. Bahdanau, D., K. Cho, and Y. Bengio, *Neural machine translation by jointly learning to align and translate.* arXiv preprint arXiv:1409.0473, 2014.
21. Luong, M.-T., H. Pham, and C.D. Manning, *Effective approaches to attention-based neural machine translation.* arXiv preprint arXiv:1508.04025, 2015.
22. Cheng, J., L. Dong, and M. Lapata, *Long short-term memory-networks for machine reading.* arXiv preprint arXiv:1601.06733, 2016.
23. Xiong, C., V. Zhong, and R. Socher, *Dynamic coattention networks for question answering.* arXiv preprint arXiv:1611.01604, 2016.
24. Lu, J., et al. *Hierarchical question-image co-attention for visual question answering*. in *Advances In Neural Information Processing Systems*. 2016.
25. Sukhbaatar, S., J. Weston, and R. Fergus. *End-to-end memory networks*. in *Advances in neural information processing systems*. 2015.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

26. Seo, M., et al., *Bidirectional attention flow for machine comprehension.* arXiv preprint arXiv:1611.01603, 2016.
27. Shen, Y., et al. *Reasonet: Learning to stop reading in machine comprehension*. in *Proceedings of the 23rd ACM SIGKDD*. 2017. ACM.
28. Wang, S. and J. Jiang, *Machine comprehension using match-lstm and answer pointer.* arXiv preprint arXiv:1608.07905, 2016.
29. Wang, W., et al. *Gated self-matching networks for reading comprehension and question answering*. in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2017.
30. Wang, Z., et al., *Multi-perspective context matching for machine comprehension.* arXiv preprint arXiv:1612.04211, 2016.
31. Lee, K., et al., *End-to-end neural coreference resolution.* arXiv preprint arXiv:1707.07045, 2017.
32. Clark, K. and C.D. Manning, *Deep reinforcement learning for mention-ranking coreference models.* arXiv preprint arXiv:1609.08667, 2016.
33. Mitkov, R., et al. *Coreference Resolution: To What Extent Does It Help NLP Applications?* in *International Conference on Text, Speech and Dialogue*. 2012. Springer.
34. Litjens, G., et al., *Deep learning as a tool for increased accuracy and efficiency of histopathological diagnosis.* Scientific reports, 2016. **6**: p. 26286.
35. Rajkomar, A., et al., *Scalable and accurate deep learning with electronic health records.* npj Digital Medicine, 2018. **1**(1): p. 18.
36. Henderson, P., et al., *Deep reinforcement learning that matters.* arXiv preprint arXiv:1709.06560, 2017.
37. Kooi, T., et al., *Large scale deep learning for computer aided detection of mammographic lesions.* Medical image analysis, 2017. **35**: p. 303-312.
38. Gubern-Mérida, A., et al., *Automated localization of breast cancer in DCE-MRI.* Medical image analysis, 2015. **20**(1): p. 265-274.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาคผนวก ก.
ตัวอย่างผลการตอบคำถาม

ตัวอย่างการตอบคำถาม เมื่อเทียบโมเดลเต็ม (Full Model) กับโมเดลเต็มที่ไม่ได้ใช้เวกเตอร์อ้างอิง (Full Model w/o Coreference Vector)

บทความ (เลือก)
The most frequent musical contributor during the first 15 years was Dudley Simpson, who is also well known for his theme and incidental music for Blake's 7, and for his haunting theme music and score for the original 1970s version of The Tomorrow People. Simpson's first Doctor Who score was Planet of Giants (1964) and he went on to write music for many adventures of the **1960s and 1970s**, including most of the stories of the Jon Pertwee/Tom Baker periods, ending with The Horns of Nimon (1979). He also made a cameo appearance in The Talons of Weng-Chiang (as a Music hall conductor).
คำถาม : In what decades was Dudley Simpson most active in contributing to Doctor Who?
คำตอบจากโมเดลเต็มที่ไม่ได้ใช้เวกเตอร์อ้างอิง : 15 years
คำตอบจากโมเดลเต็ม : 1960s and 1970s
คำตอบที่ถูกต้อง : 1960s and 1970s

บทความ (สุ่ม)
Mean monthly temperatures range from around 53 F in January to 82 F in July. High temperatures average 64 to 92 °F (18 to 33 °C) throughout the year. High heat indices are common for the summer months in the area, with indices above 110 °F (43.3 °C) possible. The highest temperature recorded was 104 °F (40 °C) on July 11, 1879 and July 28, 1872. It is common for thunderstorms to erupt during a typical summer afternoon. These are caused by the rapid heating of the land relative to the water, combined with extremely high humidity.
คำถาม What is a common occurrence during summer days?
คำตอบจากโมเดลเต็มที่ไม่ได้ใช้เวกเตอร์อ้างอิง : thunderstorms
คำตอบจากโมเดลเต็ม : thunderstorms
คำตอบที่ถูกต้อง : thunderstorms to erupt

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตัวอย่างการตอบคำถาม เมื่อเทียบโมเดลเต็ม (Full Model) กับโมเดลเต็มที่ไม่ได้ใช้การตอบแบบสองทาง (Full Model w/o Bidirectional Answer)

บทความ (เลือก) The Chinese medical tradition of the Yuan had "Four Great Schools" that the Yuan inherited from the Jin dynasty. All four schools were based on the same intellectual foundation, but advocated different theoretical approaches toward medicine. Under the Mongols, the practice of Chinese medicine spread to other parts of the empire. Chinese physicians were brought along military campaigns by the Mongols as they expanded towards the west. Chinese ....
คำถาม : How did Chinese medicine spread?
คำตอบจากโมเดลเต็มที่ไม่ได้ใช้การตอบแบบสองทาง: empire. Chinese physicians were brought along military campaigns by the Mongols as they expanded towards the west. Chinese ....
คำตอบจากโมเดลเต็ม : to other parts of the empire.
คำตอบที่ถูกต้อง : to other parts of the empire.

บทความ (สุ่ม) The LOC included Launch Complex 39, a Launch Control Center, and a 130 million cubic foot (3.7 million cubic meter) Vertical Assembly Building (VAB) in which the space vehicle (launch vehicle and spacecraft) would be assembled on a Mobile Launcher Platform and then moved by a transporter to one of several launch pads. Although at least three pads were planned, only two, designated A and B, were completed in October 1965. The LOC also included an Operations and Checkout Building (OCB) to which Gemini and Apollo spacecraft were initially received prior to being mated to their launch vehicles. The Apollo spacecraft could be tested in two vacuum chambers capable of simulating atmospheric pressure at altitudes up to 250,000 feet (76 km), which is nearly a vacuum.
คำถาม : How many launch pads were originally planned?
คำตอบจากโมเดลเต็มที่ไม่ได้ใช้การตอบแบบสองทาง: three
คำตอบจากโมเดลเต็ม : three
คำตอบที่ถูกต้อง : three

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตัวอย่างการตอบคำถาม เมื่อเทียบโมเดลเต็ม (Full Model) กับโมเดลเต็มที่ไม่ได้ใช้ฟังก์ชันต้นทุนจากความยาวของคำตอบ (Full Model w/o Length Loss Function)

บทความ (เลือก) A president is elected by the judges for three years. Under TEU article 19(3) is to be the ultimate court to interpret questions of EU law. In fact, most EU law is applied by member state courts (the English Court of Appeal, the German Bundesgerichtshof, the Belgian Cour du travail, etc.) but they can refer questions to the EU court for a preliminary ruling. The CJEU's duty is to "ensure that in the interpretation and application of the Treaties the law is observed", although realistically it has the ability to expand and develop the law according to the principles it deems to be appropriate. Arguably this has been done through both seminal and controversial judgments, including Van Gend en Loos, Mangold v Helm, and Kadi v Commission.
คำถาม : Under which courts is most EU law applied?
คำตอบจากโมเดลเต็มที่ไม่ได้ใช้ฟังก์ชันต้นทุนจากความยาวของคำตอบ: member state courts (the English Court of Appeal, the German Bundesgerichtshof, the Belgian Cour du travail, etc.)
คำตอบจากโมเดลเต็ม : member state courts
คำตอบที่ถูกต้อง : member state courts

บทความ (สุ่ม) Effects of inequality researchers have found include higher rates of health and social problems, and lower rates of social goods, a lower level of economic utility in society from resources devoted on high-end consumption, and even a lower level of economic growth when human capital is neglected for high-end consumption. For the top 21 industrialized countries, counting each person equally, life expectancy is lower in more unequal countries (r = -.907). A similar relationship exists among US states (r = -.620).
คำถาม : Why does a lower level of economic growth occur due to high-end consumption?
คำตอบจากโมเดลเต็มที่ไม่ได้ใช้ฟังก์ชันต้นทุนจากความยาวของคำตอบ: goods, a lower level of economic utility in society from resources devoted on high-end consumption, and even a lower level of economic growth when human
คำตอบจากโมเดลเต็ม : lower rates of social goods
คำตอบที่ถูกต้อง : lower level of economic growth

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ภาคผนวก ข. ผลการทดลองบนชุดข้อมูลทดสอบที่ทำบูทสแตร๊ป 50 ครั้ง

ผลการทดลองบนชุดข้อมูลสควอด

โมเดล	ชุด1	ชุด2	ชุด3	ชุด4	ชุด5	ชุด6	ชุด7	ชุด8	ชุด9	ชุด10	ชุด11	ชุด12	ชุด13
Full Model	**77.9**	**77.2**	77.6	77.1	**77.6**	**77.3**	**77.8**	**77.6**	**77.7**	77.6	**77.2**	**77.7**	**77.9**
Full Model w/o Coreferece Vector	77.4	77.1	**78.1**	**77.4**	77.1	77.1	77.3	77.4	77.6	**77.8**	76.8	77.5	77.8
Full Model w/o Bidirectional Answer	77.3	76.7	77.3	76.9	77.6	76.7	77.4	77.3	77.2	76.8	76.7	77.5	77.4
Full Model w/o Length Loss Function	77.4	77.3	77.6	**77.4**	77.4	77.2	77.6	77.3	77.1	77.4	77.0	77.0	77.4

ชุด14	ชุด15	ชุด16	ชุด17	ชุด18	ชุด19	ชุด20	ชุด21	ชุด22	ชุด23	ชุด24	ชุด25	ชุด26	ชุด27	ชุด28	ชุด29	ชุด30	ชุด31	ชุด32
78.1	76.6	**77.4**	77.1	77.6	**77.6**	**77.4**	**77.4**	**77.7**	77.7	**76.9**	**77.9**	**77.9**	77.9	77.4	77.4	**77.5**	**78.4**	**77.7**
78.1	**77.3**	77.1	**77.2**	77.5	77.4	77.0	77.2	77.3	77.8	77	77.3	77.6	**78.2**	**77.5**	77.4	76.9	77.6	77.3
77.3	76.4	76.7	76.7	77.3	77.1	76.8	76.8	77.5	77.3	76.3	77.3	77.1	77.8	77.0	77.2	77.4	78.1	77.1
77.5	76.8	77.2	76.8	**77.8**	77.5	76.8	76.8	77.4	**78.0**	76.7	77.2	77.6	77.7	**77.5**	**77.5**	77.0	78.0	77.4

ชุด33	ชุด34	ชุด35	ชุด36	ชุด37	ชุด38	ชุด39	ชุด40	ชุด41	ชุด42	ชุด43	ชุด44	ชุด45	ชุด46	ชุด47	ชุด48	ชุด49	ชุด50
77.9	**77.3**	**77.9**	77.5	77.5	**77.5**	**77.6**	**78.0**	**76.7**	77.4	**77.3**	**77.5**	**77.2**	**77.9**	**77.5**	**77.4**	77.4	77.5
77.6	76.9	77.8	**77.6**	77.5	77.1	77.6	**78.0**	76.4	**77.5**	77.0	77.3	77.1	77.3	77.1	**77.4**	**77.5**	77.2
77.6	76.9	77.2	77.2	76.9	76.8	77.4	76.9	76.6	76.4	77.0	77.0	76.9	77.4	76.6	77.3	77.1	77.1
77.4	76.7	**77.9**	77.5	77.5	76.9	77.3	77.6	76.6	77.4	77.2	77.1	77.1	77.5	76.9	77.0	77.2	**77.6**

ผลการทดลองบนชุดข้อมูลสัพเพเหระ

โมเดล	ชุด1	ชุด2	ชุด3	ชุด4	ชุด5	ชุด6	ชุด7	ชุด8	ชุด9	ชุด10	ชุด11	ชุด12
Full Model	**69.2**	**69.1**	**69.4**	**69.5**	**69.3**	**69.3**	69.5	**70.6**	69.0	69.3	69.5	**69.2**
Full Model w/o Coreferece Vector	68.2	67.7	68.7	68.9	68.0	68.2	68.5	69.5	68.5	68.2	69.0	68.2
Full Model w/o Bidirectional Answer	68.8	68.9	68.8	68.7	68.8	68.7	**69.6**	69.6	68.7	68.8	69.3	68.8
Full Model w/o Length Loss Function	68.9	**69.1**	68.8	**69.5**	68.9	68.8	**69.6**	70.1	**69.2**	**69.7**	**69.6**	68.9

ชุด13	ชุด14	ชุด15	ชุด16	ชุด17	ชุด18	ชุด19	ชุด20	ชุด21	ชุด22	ชุด23	ชุด24	ชุด25	ชุด26	ชุด27	ชุด28	ชุด29	ชุด30	ชุด31
69.0	**69.4**	**69.9**	69.4	**70.0**	68.4	69.4	68.8	69.6	70.1	68.7	**69.0**	**70.2**	69.4	**69.2**	**68.7**	**69.7**	**69.8**	**69.4**
68.5	68.0	68.5	68.3	68.6	67.6	67.9	67.7	68.5	69.1	67.9	68.2	69.1	67.8	69.1	68.1	68.5	68.6	69.0
68.8	69.0	69.1	**69.5**	69.1	**68.7**	**69.6**	68.5	69.3	70.0	68.2	**69.0**	69.1	69.1	69.1	68.2	69.1	69.0	**69.4**
68.8	68.6	69.1	**69.5**	69.3	**68.7**	69.3	**69.2**	**70.5**	**70.3**	**69.3**	68.9	69.9	**69.8**	69.0	68.4	69.6	69.6	69.0

ชุด32	ชุด33	ชุด34	ชุด35	ชุด36	ชุด37	ชุด38	ชุด39	ชุด40	ชุด41	ชุด42	ชุด43	ชุด44	ชุด45	ชุด46	ชุด47	ชุด48	ชุด49	ชุด50
68.8	69.3	**69.9**	**70.6**	69.4	**69.3**	69.5	69.3	68.8	69.4	69.3	**69.3**	**70.1**	**69.1**	69.5	68.2	**70.2**	69.1	**69.7**
67.5	68.5	68.7	69.3	68.7	67.8	68.0	68.2	68.1	68.3	68.8	68.1	69.4	68.0	68.6	67.4	69.0	68.3	68.8
68.5	68.9	69.2	69.7	69.6	68.9	68.7	68.8	68.6	70.0	68.8	67.8	70.0	69.0	68.8	68.2	69.6	69.2	69.1
68.3	**69.6**	69.2	70.6	**70.0**	68.8	**69.6**	**68.8**	**69.3**	**70.1**	**69.4**	68.2	70.0	68.4	**69.6**	**68.5**	69.7	**69.7**	69.2

ผลการเปรียบเทียบจำนวนครั้งที่ชนะบนชุดข้อมูลสควอดที่ทำการบูทสแตร๊ป 50 ครั้ง

โมเดลที่ใช้เปรียบเทียบกับ Full Model	จำนวนครั้งที่ Full Model ชนะ	จำนวนครั้งที่ Full Model แพ้	จำนวนครั้งที่ Full Model เสมอ
Full Model w/o Coreferece Vector	32	13	2
Full Model w/o Bidirectional Answer	49	0	1
Full Model w/o Length Loss Function	37	8	5

ผลการเปรียบเทียบจำนวนครั้งที่ชนะบนชุดข้อมูลสัพเพเหระที่ทำการบูทสแตร๊ป 50 ครั้ง

โมเดลที่ใช้เปรียบเทียบกับ Full Model	จำนวนครั้งที่ Full Model ชนะ	จำนวนครั้งที่ Full Model แพ้	จำนวนครั้งที่ Full Model เสมอ
Full Model w/o Coreferece Vector	50	0	0
Full Model w/o Bidirectional Answer	40	7	3
Full Model w/o Length Loss Function	20	20	10

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ประวัติผู้เขียนวิทยานิพนธ์

นายณัฐชัย ตรีทศายุธ เกิดเมื่อวันที่ 19 ตุลาคม พ.ศ. 2533 ที่กรุงเทพมหานคร สำเร็จการศึกษาระดับปริญญาตรีหลักสูตรวิศวกรรมศาสตร์บัณฑิต (เกียรตินิยมอันดับ 2) สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2555 และเข้าศึกษาในหลักสูตรวิศวกรรมคอมพิวเตอร์มหาบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2559

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY