บทความวิจัย

การลดความผิดพลาดในการพิมพ์ภาษาไทยบนแป้นพิมพ์เสมือนด้วยวิธีปรับพื้นที่ปุ่มเป้าหมายและการใช้แบบจำลองทางภาษา LSTM

วัชร สุวรรณโสภณ และ ฐิติรัตน์ ศิริบวรรัตนกุล*
สาขาวิชาวิทยาการข้อมูล คณะสถิติประยุกต์ สถาบันบัณฑิตพัฒนบริหารศาสตร์

* ผู้นิพนธ์ประสานงาน โทรศัพท์ 0 2727 3067 อีเมล: thitirat@as.nida.ac.th
DOI: 10.14416/j.kmutnb.2021.09.001
รับเมื่อ 20 เมษายน 2564 แก้ไขเมื่อ 22 มิถุนายน 2564 ตอบรับเมื่อ 9 กรกฎาคม 2564 เผยแพร่ออนไลน์ 7 กันยายน 2564
© 2021 King Mongkut's University of Technology North Bangkok. All Rights Reserved.

บทคัดย่อ

จากความแพร่หลายของการใช้งานสมาร์ตโฟนในปัจจุบันทำให้แป้นพิมพ์เสมือนกลายเป็นหนึ่งในเครื่องมือหลักของการนำเข้าข้อมูลตัวอักษรสู่สมาร์ตโฟน แต่ด้วยขนาดปุ่มที่เล็กของแป้นพิมพ์เสมือนบนสมาร์ตโฟนจึงเป็นเหตุให้ผู้ใช้พิมพ์ผิดพลาดไปจากปุ่มที่ต้องการได้บ่อยครั้ง ซึ่งจากการทบทวนวรรณกรรมผู้วิจัยพบว่า มีการนำเสนอเทคนิคต่างๆ ในการแก้ปัญหานี้บนแป้นพิมพ์เสมือนภาษาต่างประเทศ แต่ยังไม่พบงานวิจัยลักษณะนี้สำหรับแป้นพิมพ์เสมือนภาษาไทย จึงเป็นที่มาของงานวิจัยชิ้นนี้ที่ผู้วิจัยทดลองนำคลังคำศัพท์มาตรฐานภาษาไทยมาเป็นชุดข้อมูลเพื่อฝึกสอน สร้าง และทดสอบแบบจำลองภาษา 2 ตัว ได้แก่ แบบจำลองภาษาที่ใช้การเรียนรู้ของเครื่องจักรชนิด Markov Chain และแบบจำลองภาษาที่ใช้การเรียนรู้เชิงลึกชนิด LSTM โดยหน้าที่ของแบบจำลองภาษาแต่ละตัวคือคาดเดาปุ่มบนแป้นพิมพ์เสมือนภาษาไทยตัวถัดไปที่ผู้ใช้น่าจะกดเพื่อให้สมาร์ตโฟนสามารถทำการขยายขนาดปุ่มเป้าหมายให้ใหญ่ขึ้นรอไว้ล่วงหน้าได้ ทั้งนี้โดยมีเป้าหมายเพื่อช่วยลดอัตราการพิมพ์ผิดของผู้ใช้บนแป้นพิมพ์เสมือนภาษาไทย ผลการทดลองพบว่า แบบจำลองภาษาชนิด LSTM ให้ผลลัพธ์การพยากรณ์ที่ดีกว่าชนิด Markov Chain และเมื่อนำแบบจำลองภาษา LSTM ตัวดังกล่าวไปทดลองกับข้อมูลการพิมพ์ของผู้เข้าร่วมการทดลอง 7 คน ซึ่งเก็บรวบรวมมาโดยแอปพลิเคชันต้นแบบบนสมาร์ตโฟนที่ผู้วิจัยพัฒนาขึ้นพบว่า การใช้แบบจำลองภาษา LSTM ร่วมกับการขยายขนาดปุ่มล่วงหน้าสามารถช่วยลดความผิดพลาดในการพิมพ์ได้จริง โดยสามารถลดการพิมพ์ผิดได้ทั้งหมด 13 ครั้ง จากการพิมพ์ผิดทั้งหมด 257 ครั้ง หรือคิดเป็น 5.05%

คำสำคัญ: ปัญญาประดิษฐ์ การเรียนรู้ของเครื่องจักร การเรียนรู้เชิงลึก แบบจำลองภาษา แบบจำลอง LSTM แป้นพิมพ์เสมือนภาษาไทย สมาร์ตโฟน

การอ้างอิงบทความ: วัชร สุวรรณโสภณ และ ฐิติรัตน์ ศิริบวรรัตนกุล, "การลดความผิดพลาดในการพิมพ์ภาษาไทยบนแป้นพิมพ์เสมือนด้วยวิธีปรับพื้นที่ปุ่มเป้าหมายและการใช้แบบจำลองทางภาษา LSTM," *วารสารวิชาการพระจอมเกล้าพระนครเหนือ*, 2564, doi: 10.14416/j.kmutnb.2021.09.001.

วารสารวิชาการพระจอมเกล้าพระนครเหนือ, 2564
The Journal of KMUTNB., 2021

Research Article

Combining Key-target Resizing and LSTM Language Model to Reduce Typos in Thai Soft Keyboard

Vachara Suwansophon and Thitirat Siriborvornratanakul*
Data Science, School of Applied Statistics, National Institute of Development Administration, Bangkok, Thailand

* Corresponding Author, Tel. 0 2727 3067, E–mail: thitirat@as.nida.ac.th
DOI: 10.14416/j.kmutnb.2021.09.001
Received 20 April 2021; Revised 22 June 2021; Accepted 9 July 2021; Published online: 7 September 2021
© 2021 King Mongkut's University of Technology North Bangkok. All Rights Reserved.

Abstract

Recently virtual keyboard has become one of the main user interfaces for entering textual data to a smartphone. For virtual keyboards in foreign languages, there are many researches that study how to reduce typos caused by the small size of each button in the virtual keyboard. Nevertheless, as we do not find this kind of researches for Thai virtual keyboard, we propose our work that experiments and evaluates feasibility of using a combination of language model and key-target resizing technique to reduce typos on Thai virtual keyboard. Our work starts by using standard Thai vocabulary corpuses to train two language models (i.e., Markov Chain and LSTM) in order to predict the most likely buttons that a user will press next. Then, we collect typing data on Thai virtual keyboard from seven users using our prototype system. Finally, we analyze the collected data in conjunction with predicted results from our language model. According to our experimental results, the LSTM based language model performs better than the Markov Chain based language model in predicting the next Thai's character buttons. When this LSTM language model is used to enlarge six buttons with highest predicted probabilities in advance, results show that it helps reduce typos by 5.05%. More specifically, the number of typos is reduced by 13 out of 257 typos.

Keywords: Artificial Intelligence, Machine Learning, Deep Learning, Language Model, LSTM, Thai Soft Keyboard, Smartphone

Please cite this article in press as: V. Suwansophon and T. Siriborvornratanakul, "Combining key-target resizing and LSTM language model to reduce typos in Thai soft keyboard," *The Journal of KMUTNB*, 2021 (in Thai), doi: 10.14416/j.kmutnb.2021.09.001.

1. บทนำ

อ้างอิงจากข้อมูลของบริษัทแอดวานซ์ อินโฟร์ เซอร์วิส (AIS) [1] ที่ระบุว่าเวลาที่ผู้ใช้ใช้โทรเฉลี่ยต่อเลขหมายต่อเดือนนั้น มีแนวโน้มลดลงติดต่อกันมาเป็นปีที่ 4 แล้ว ในขณะที่อัตราการใช้งานอินเทอร์เน็ตเฉลี่ยต่อเลขหมายต่อเดือนกลับเพิ่มสูงขึ้นทุกปี ข้อมูลนี้ชี้ให้เห็นถึงรูปแบบการสื่อสารผ่านโทรศัพท์มือถือที่เปลี่ยนแปลงไปของคนไทย จากเดิมที่เป็นการสื่อสารแบบพูดคุยด้วยเสียง กลายมาเป็นการสื่อสารแบบสื่อผสม (Multimedia) ที่มีทั้งภาพ เสียง และวิดีโอผ่านการใช้งานสมาร์ตโฟน (Smartphone) ภายใต้วิถีการใช้งานใหม่นี้ แป้นพิมพ์เสมือนหรือคีย์บอร์ดเสมือน (Virtual Keyboard) ซึ่งเป็นหนึ่งในวิธีหลักในการนำเข้าข้อมูลการพิมพ์บนสมาร์ตโฟน จึงมีบทบาทความสำคัญเพิ่มมากขึ้น

ข้อดีของแป้นพิมพ์เสมือนคือ ความยืดหยุ่นในการใช้งาน โดยผู้ใช้สามารถปรับแต่งขนาด สี รวมถึงรูปแบบการจัดวางปุ่มให้เหมาะกับตนเองได้ แต่ข้อเสียของแป้นพิมพ์เสมือนคือไม่ให้ความรู้สึกในการสัมผัส (Tactile Feedback) และขาดความแม่นยำเมื่อใช้นิ้วเป็นสื่อสัมผัส นำไปสู่การพิมพ์ผิดจากความไม่แม่นยำ (Ambiguous Key Typos) ที่สูงขึ้นโดยเฉพาะกับขนาดพื้นที่การพิมพ์ที่เล็กลง [2] ทั้งนี้ในแป้นพิมพ์เสมือนภาษาต่างประเทศมีความพยายามในการเพิ่มพูนประสบการณ์การใช้งาน (User Experience) ของการพิมพ์บนแป้นพิมพ์เสมือนอยู่หลายวิธี อาทิ ลดการพิมพ์ผิดด้วยการแก้ไขคำที่ผิดอัตโนมัติ (Autocorrection) คาดเดาคำอัตโนมัติ (Autocomplete) หรือนำการเรียนรู้ของเครื่องจักร (Machine Learning) มาใช้เพื่อช่วยเพิ่มความแม่นยำในการพิมพ์ โดยกระบวนวิธีการแก้ปัญหานั้นสามารถแบ่งออกได้เป็น 2 ประเภทใหญ่ [3] คือ 1) การสร้างตัวแบบเพื่อลดความคลาดเคลื่อนจากจุดสัมผัสหรือ Touch Point Modeling [4] และ 2) การใช้ตัวแบบทางภาษา (Language Model) เพื่อปรับขนาดของปุ่มเป้าหมายหรือ Key-target Resizing [3]

ใน พ.ศ. 2561 ประเทศไทยมีจำนวนผู้ใช้งานโทรศัพท์มือถืออยู่ที่ 56.7 ล้านคน หรือคิดเป็นร้อยละ 89.6 ของจำนวนประชากร [5] ซึ่งถือเป็นตัวเลขที่สูงมาก (ประมาณอันดับที่ 20 ของโลก) แต่เป็นที่น่าสังเกตว่ายังไม่พบการศึกษาวิจัยเพื่อนำการเรียนรู้ของเครื่องจักร และแบบจำลองภาษามาใช้ทดลองกับแป้นพิมพ์เสมือนภาษาไทย จากเหตุดังกล่าวนี้ทำให้ผู้วิจัยมีความสนใจจะทำการค้นคว้าวิจัยถึงความเป็นไปได้ในการนำแบบจำลองทางภาษามาสร้างตัวแบบการปรับขนาดของปุ่มเป้าหมาย เพื่อลดการพิมพ์ผิดจากความไม่แม่นยำบนแป้นพิมพ์เสมือนภาษาไทย

2. วัสดุ อุปกรณ์และวิธีการวิจัย

2.1 การเตรียมข้อมูล

งานวิจัยนี้ใช้ชุดคำจากคลังศัพท์ภาษาไทย 3 ชุด ได้แก่

2.1.1 คลังศัพท์ของโครงการคลังข้อมูลภาษาไทยแห่งชาติ ในพระราชูปถัมภ์ของสมเด็จพระเทพรัตนราชสุดาฯ สยามบรมราชกุมารี (Thai National Corpus) [6] โดยคลังศัพท์นี้มีจำนวนคำเอกลักษณ์ (Unique Words) รวม 53,894 คำ พร้อมค่าความถี่ ซึ่งผู้วิจัยได้ทำการเลือกมาเฉพาะคำที่มีแต่พยัญชนะ สระ วรรณยุกต์ เครื่องหมายวรรคตอนภาษาไทย และที่ว่าง คงเหลือเป็นข้อมูลคลังศัพท์ภาษาไทยจำนวน 281,109 ตัวอักษร ซึ่งจะถูกใช้ในการฝึกสอน (Train) แบบจำลองทางภาษา

2.1.2 คลังศัพท์ BEST I Corpus จาก NECTEC [7] ซึ่งเป็นคลังศัพท์สำหรับฝึกตัดคำด้วยคอมพิวเตอร์ โดยผู้วิจัยนำข้อมูลนี้มาผ่านการจัดเตรียมนำอักขระที่ไม่ใช่ภาษาไทยออก คงเหลือไว้เพียงพยัญชนะ สระ วรรณยุกต์ เครื่องหมายวรรคตอนภาษาไทย และที่ว่างเพื่อคั่นประโยคเท่านั้น จากนั้นผู้วิจัยจึงนำข้อมูลมาจัดเรียงต่อกันได้เป็นชุดข้อมูลคลังศัพท์ภาษาไทยจำนวน 19,994,984 ตัวอักษร ซึ่งจะถูกใช้ในการฝึกสอนแบบจำลองทางภาษา

2.1.3 คลังประโยคจาก GitHub ของโครงการ Mozilla Common Voice [8] ซึ่งผู้วิจัยนำมาผ่านกระบวนการทำความสะอาดข้อมูล (Data Cleansing) เช่นเดียวกันกับ BEST I Corpus ทำให้เหลือประโยคที่ใช้งานได้อยู่ 39 ประโยค คิดเป็น 1,463 ตัวอักษร โดยข้อมูลทั้งหมดนี้จะถูกนำไปใช้เพื่อการทดสอบ (Test) การใช้งานแบบจำลองภาษาเท่านั้น

ทั้งนี้การใช้คลังศัพท์ที่ประกอบด้วยกลุ่มคำที่ตรงกับ

หมวดหมู่การใช้งานจะทำให้ได้ผลลัพธ์ที่ดีกว่าการใช้คลังศัพท์แบบรวม ตัวอย่างเช่น หากเป้าหมายคือการแก้ไขการพิมพ์ผิดของผู้ใช้เครือข่ายสังคมออนไลน์ (Social Media) บนสมาร์ตโฟน ก็ควรจะใช้คลังศัพท์ของการพิมพ์ภาษาไทยบนสมาร์ตโฟนของสื่อสังคมออนไลน์โดยเฉพาะ อย่างไรก็ตาม เนื่องจากคำที่ใช้ในสื่อสังคมออนไลน์มีความแปรผันในรูปแบบของคำสูง เช่น คำว่า “ครับ” อาจถูกย่อเหลือเพียง “คับ” ซึ่งความแปรผันนี้เป็นอีกหนึ่งตัวแปรที่สามารถกระทบต่อผลลัพธ์งานวิจัยได้ อีกทั้งผู้วิจัยยังไม่พบว่า มีการจัดทำคลังศัพท์ภาษาไทยที่รวบรวมจากสื่อสังคมออนไลน์แต่อย่างใด ดังนั้นในงานวิจัยต้นแบบนี้ เพื่อทดสอบสมมติฐานเบื้องต้นให้ได้ลุล่วงก่อน ผู้วิจัยจึงเลือกใช้คลังศัพท์ภาษาไทย 3 ชุด ดังที่กล่าวไปข้างต้น เพราะเป็นคลังศัพท์ภาษาไทยที่ได้มาตรฐานและเป็นที่นิยมใช้เพื่อทดสอบแบบจำลองการเรียนรู้ของเครื่องจักร

2.2 แบบจำลองทางภาษา

ในงานวิจัยนี้ผู้วิจัยทำการทดลองสร้างแบบจำลองทางภาษาโดยใช้เทคนิคที่แตกต่างกัน 2 ชนิด ได้แก่ เทคนิคการเรียนรู้ด้วยเครื่องจักรโดยใช้ Markov Chain และเทคนิคการเรียนรู้เชิงลึก (Deep Learning) โดยใช้ LSTM (Long Short-term Memory)

2.2.1 Markov Chain

ในการสร้างแบบจำลองภาษาด้วย Markov Chain ผู้วิจัยใช้ข้อมูลทั้งหมดจากคลังศัพท์ของโครงการคลังข้อมูลภาษาไทยแห่งชาติ ในพระราชูปถัมภ์ของสมเด็จพระเทพรัตนราชสุดาฯ สยามบรมราชกุมารี โดยนำข้อมูลไปสร้างเป็น Character-level 3-grams เหตุผลที่เลือกใช้ 3-grams นั้น อ้างอิงจากที่ผู้วิจัยได้ทำการทดลองเบื้องต้นเปรียบเทียบระหว่าง Unigram, Bigrams และ 3-grams โดยทำการฝึกสอนด้วยคลังศัพท์ BEST I Corpus และทำการทดสอบด้วยคลังศัพท์ Mozilla Common Voice Corpus ผลที่ได้พบว่า ความแม่นยำในการคาดเดาตัวอักษรถัดไปอยู่ที่ 16.49%, 2.82% และ 45.31% สำหรับ Unigram, Bi-grams และ 3-grams ตามลำดับ หรือก็คือ 3-grams ให้ผลการคาดเดาที่แม่นยำที่สุดจากการทดสอบเบื้องต้นนี้นั่นเอง

สำหรับการนำ Character-level 3-grams มาทดสอบนั้น ผู้วิจัยได้ทำการทดลองทั้งแบบที่ใช้และไม่ใช้ความถี่ของคำเป็นตัวถ่วงน้ำหนัก ในส่วนของแบบจำลองนั้นจะรับอินพุตเป็นตัวอักษร 1–2 ตัว และให้เอาต์พุตค่าความน่าจะเป็นของตัวอักษรถัดไป ทั้งนี้เมื่อผู้วิจัยทำการวัดผลโดยดูจากตัวอักษรที่มีความเป็นไปได้สูงสุดพบว่า สำหรับข้อมูลในชุดฝึกสอนของคลังศัพท์ Thai National Corpus ความแม่นยำของวิธีแบบถ่วงน้ำหนักและแบบไม่ถ่วงน้ำหนัก คือ 54.71% และ 49.15% ตามลำดับ ในขณะที่ความแม่นยำในชุดทดสอบของคลังศัพท์ Thai National Corpus อยู่ที่ 44.79% และ 40.56% ตามลำดับ

อนึ่ง เนื่องจากการทดลองในหัวข้อนี้เป็นการสร้าง Markov Chain จากชุดความถี่ของคำ และนำข้อมูลซึ่งอาจจะเป็นชุดเดียวกันหรือชุดใหม่มาดูความเป็นไปได้สูงสุดจาก Markov Chain ทีละตัวอักษร ด้วยวิธีการทดลองเช่นนี้ ผู้วิจัยจึงไม่ได้แบ่งชุดข้อมูลสำหรับฝึกสอนและชุดข้อมูลสำหรับทดสอบ แต่ใช้ข้อมูลทั้งหมดที่มีสร้าง Markov Chain ขึ้นมา และการวัดความแม่นยำก็วัดจากความน่าจะเป็นของตัวอักษรที่ Markov Chain แสดงอยู่ โดยหากตัวอักษรที่มีความน่าจะเป็นสูงสุด (1–6 อันดับแรก แล้วแต่การทดสอบ) ตรงกับตัวอักษรของข้อมูลชุดทดสอบก็จะถือว่าทำนายได้ถูกต้อง

นอกจากนี้ผู้วิจัยยังทดลองสร้าง Markov Chain แบบไม่ถ่วงน้ำหนักจากข้อมูลทั้งหมดในคลังศัพท์ BEST I Corpus ทั้งนี้เนื่องจากคลังศัพท์ BEST I Corpus ไม่มีการให้ค่าความถี่ของคำประกอบมาด้วย แต่ข้อมูลจะอยู่ในรูปของประโยคเต็มอยู่แล้ว (หรือก็คือ คำ 1 คำ สามารถมีปรากฏอยู่หลายที่) ดังนั้นการนำข้อมูลทั้งหมดนี้เข้าไปสร้าง Markov Chain จึงเปรียบเสมือนการถ่วงน้ำหนักตามความถี่ของคำที่ปรากฏในประโยคโดยอัตโนมัติ ผู้วิจัยจึงมิได้ทำการทดลองแบบถ่วงน้ำหนักกับคลังศัพท์ BEST I Corpus นี้ ในส่วนของการทดสอบนั้น เนื่องจากข้อมูลในคลังศัพท์ BEST I Corpus มีขนาดใหญ่เกินกว่า 2 ล้านบรรทัด ด้วยข้อจำกัดของสมรรถนะคอมพิวเตอร์ของผู้วิจัย ผู้วิจัยจึงไม่ได้ทำการทดสอบด้วย

คลังศัพท์ BEST I Corpus ทั้งหมด แต่ใช้การสุ่มเลือก (Sample without Replacement) ข้อมูล 1,000 ประโยค ขึ้นมาจำนวน 10 ครั้ง (รวมเป็นข้อมูลที่ใช้ทดสอบ 10,000 ประโยค) ผลลัพธ์ที่ได้มีความแม่นยำเฉลี่ยอยู่ที่ 50.05% นอกจากนี้ผู้วิจัยยังทำการทดลองเพิ่มเติมโดยใช้ชุดศัพท์ Mozilla Common Voice เป็นข้อมูลชุดทดสอบ ผลการทดลองได้ความแม่นยำ 45.31% แต่หากดูที่ความแม่นยำ 6 อันดับแรก (Top-6 accuracy) จะได้ความแม่นยำสูงถึง 79.04%

ในรายละเอียดของการสร้าง Character-level 3-grams สำหรับคลังศัพท์ Thai National Corpus และคลังศัพท์ BEST I Corpus นั้น มีวิธีการสร้างที่แตกต่างกัน โดยสำหรับคลังศัพท์ Thai National Corpus เนื่องจากข้อมูลอยู่ในรูปแบบของคำและความถี่ของคำ ผู้วิจัยจึงนำคำศัพท์ไปสร้าง 3-grams ได้โดยตรง และในการถ่วงน้ำหนักก็จะทำการ “ซ้ำ” คำดังกล่าวเข้าไปตามจำนวนที่ระบุด้วยค่าความถี่ ในขณะที่สำหรับคลังศัพท์ BEST I Corpus นั้น เนื่องจากข้อมูลอยู่ในรูปแบบของประโยคจากบทความ ผู้วิจัยจึงนำประโยคที่มีมาเรียงต่อกัน คั่นแต่ละประโยคด้วยช่องว่าง จากนั้นจึงนำสายของอักขระนี้ไปสร้างตาราง 3-grams ตามตัวอักษรที่ปรากฏได้เป็น Markov Chain ดังที่อธิบายไปข้างต้น

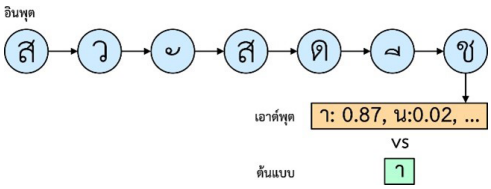
2.2.2 LSTM

ในการทดลองนี้ผู้วิจัยเลือกใช้ซอฟต์แวร์เฟรมเวิร์กการเรียนรู้เชิงลึกของ TensorFlow 2.1.0 เพื่อสร้างแบบจำลอง LSTM ขึ้นมาตามสถาปัตยกรรมที่ระบุในรูปที่ 1 โดยแบบจำลอง LSTM นี้ รับอินพุต คือ ชุดของตัวอักษรที่มีความยาวชุดละ 7 ตัวอักษร และให้เอาต์พุตคือ ค่าความน่าจะเป็นของตัวอักษรถัดไปดังตัวอย่างในรูปที่ 2

ในรายละเอียดแบบจำลองภาษา LSTM ตัวนี้เริ่มจากการนำอินพุตมาแปลงให้เป็นค่า Embedding หรือก็คือแปลงจากตัวอักษร 1 ตัว ให้กลายเป็นเวกเตอร์ของตัวเลขขนาด 73 ช่อง นั่นเอง โดยตัวเลข 73 นี้ผู้วิจัยอ้างอิงมาจากจำนวนรวมของพยัญชนะ สระ วรรณยุกต์ เครื่องหมายวรรคตอนภาษาไทย และที่ว่าง ซึ่งเลเยอร์ Embedding นี้ผู้วิจัยทำการฝึกฝนเองทั้งหมด (Train From Scratch) ไม่ได้มีการใช้

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 7, 73)             5329
_________________________________________________________________
lstm_1 (LSTM)                (None, 128)               103424
_________________________________________________________________
dense_1 (Dense)              (None, 73)                9417
=================================================================
Total params: 118,170
Trainable params: 118,170
Non-trainable params: 0
```

รูปที่ 1 สถาปัตยกรรมของแบบจำลองภาษา LSTM



รูปที่ 2 ตัวอย่างอินพุตและเอาต์พุตของแบบจำลองภาษา LSTM

Pre-trained Embedding ใดๆ โดยจากการทดลอง 6 ครั้ง ผลลัพธ์ที่ได้ไม่พบความแตกต่างอย่างมีนัยสำคัญ โดยผลลัพธ์ที่ดีที่สุดคือเลเยอร์ Embedding ที่มีสถาปัตยกรรมดังแสดงในรูปที่ 1 นั่นเอง เมื่อเสร็จจากการทำงานในเลเยอร์แรก ข้อมูลจะถูกส่งต่อไปให้กับชั้นของ LSTM ซึ่งใช้ Activation Function คือ Tanh และสุดท้ายผลลัพธ์จากชั้น LSTM ก็ถูกส่งต่อมาที่ชั้น Dense ซึ่งเป็นชั้นสุดท้ายของเอาต์พุต (Output Layer) ที่มีการใช้ Softmax Activation Function ในการพยากรณ์ความน่าจะเป็นของปุ่มถัดไปทั้งหมด 73 ปุ่ม ทั้งนี้สำหรับการฝึกสอนแบบจำลองภาษา LSTM ผู้วิจัยเลือกใช้ Loss Function ชนิด Categorical Cross-Entropy และใช้ Adam Optimizer

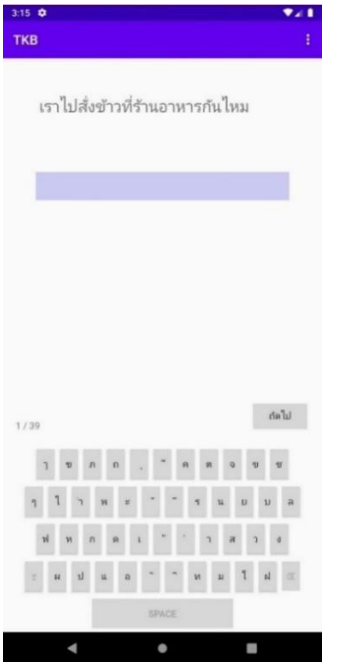
ในการทดลองฝึกสอนแบบจำลองภาษา LSTM ด้วยชุดข้อมูล Thai National Corpus ผู้วิจัยพบว่า ผลลัพธ์ที่ได้มีความแม่นยำเพียง 20% คาดว่าเป็นเพราะจำนวนข้อมูลฝึกสอนมีน้อยเกินไปสำหรับแบบจำลองการเรียนรู้เชิงลึก ดังนั้นผู้วิจัยจึงเปลี่ยนมาฝึกสอนแบบจำลองภาษา LSTM ด้วยชุดคำจากคลังศัพท์ BEST I Corpus จาก NECTEC แทน โดยนำข้อมูลทั้งหมดมาตัดออกเป็นชุดของตัวอักษรที่ความยาวชุดละ 8 ตัวอักษร หรือก็คือให้แบบจำลองภาษา LSTM ดูตัวอักษร 7 ตัว ก่อนหน้าเพื่อจะคาดเดาตัวอักษร

ลำดับที่ 8 นั่นเอง ผลการทดลองได้ความแม่นยำบนข้อมูลชุดฝึกสอน คือ 50.29% และได้ความแม่นยำเฉลี่ยจากการทำ K-folds Cross-validation (K=10) อยู่ที่ 50.37% จากนั้นผู้วิจัยนำแบบจำลองภาษา LSTM ที่ได้ไปทดลองกับข้อมูลชุดทดสอบของ Mozilla Common Voice ด้วย K-folds Cross-validation (K=10) เช่นเดิม ได้ความแม่นยำ (Top-1 Accuracy) เฉลี่ยแล้วที่ 59.66% อย่างไรก็ตาม เมื่อพิจารณาจากผังการวางปุ่มบนแป้นพิมพ์เสมือนดังตัวอย่างในรูปที่ 3 จะเห็นว่าปุ่มหนึ่งปุ่มจะมีปุ่มที่อยู่ล้อมรอบได้สูงสุดเพียง 6 ปุ่มเท่านั้น ดังนั้นผู้วิจัยจึงเพิ่มความสนใจให้ครอบคลุมปุ่มที่มีค่าความน่าจะเป็นจากการพยากรณ์สูงสุด 6 อันดับแรก โดยเมื่อทดสอบว่าตัวอักษรถัดไปปรากฏอยู่ใน 6 ลำดับของปุ่มดังกล่าวหรือไม่ (Top-6 Accuracy) ก็ได้ความแม่นยำอยู่ที่ 75.35% จากการทดสอบเพียงครั้งเดียว

จากผลการทดลองทั้งหมดที่กล่าวไปในหัวข้อ 2.2.1 และ 2.2.2 ผู้วิจัยได้ข้อสรุปว่าแบบจำลองภาษา LSTM ซึ่งมีผลลัพธ์ความแม่นยำ (Top-1 Accuracy) ที่ 59.66% นั้นมีความแม่นยำในการพยากรณ์ปุ่มถัดไปบนแป้นพิมพ์เสมือนภาษาไทยดีกว่าแบบจำลองภาษา Markov Chain ที่ให้ความแม่นยำเพียง 45.31% ดังนั้นขั้นตอนต่อจากนี้ไปผู้วิจัยจะนำแบบจำลองภาษา LSTM นี้ไปใช้ในการทดลอง

2.3 การเก็บข้อมูลการพิมพ์จากผู้เข้าร่วมการทดลอง

เนื่องจากหัวใจของงานวิจัยชิ้นนี้คือการขยายขอบเขตของตัวอักษร (ขยายขนาดปุ่ม) ให้รองรับ “จุดสัมผัส” ที่มากขึ้นบนแป้นพิมพ์เสมือนโดยอิงตามพฤติกรรมจริงของผู้ใช้ ในการนี้ผู้วิจัยจึงทำการเก็บข้อมูลจากอาสาสมัครผู้เข้าร่วมการทดลองทั้งสิ้น 7 คน (ผู้ชาย 2 คน และผู้หญิง 5 คน) โดยมีเกณฑ์การคัดเลือก ได้แก่ ผู้เข้าร่วมการทดลองต้องเป็นผู้ใช้โทรศัพท์มือถือแบบจอสัมผัส และใช้งานเครือข่ายสังคมเป็นประจำอยู่แล้ว เพื่อลดความผิดพลาดจากความไม่คุ้นชินในการใช้งานสมาร์ตโฟน อีกทั้งผู้เข้าร่วมการทดลองต้องมีอายุอยู่ระหว่าง 18–34 ปี และมี BMI อยู่ในช่วงที่กำหนด คือ 18.5–24.9 เพื่อหลีกเลี่ยงตัวแปรขนาดของนิ้วมือที่แตกต่างกันอย่างมีนัยสำคัญของผู้เข้าร่วมการทดลองแต่ละราย

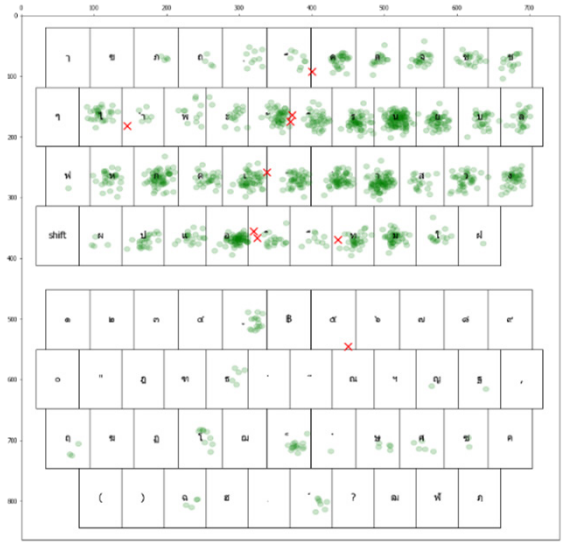


รูปที่ 3 แอปพลิเคชันต้นแบบบนสมาร์ตโฟนสำหรับให้ผู้เข้าร่วมทดลองพิมพ์ลงในกล่องข้อความสีฟ้าตามประโยคที่ปรากฏและกดปุ่มถัดไป ทำเช่นนี้ซ้ำจนครบ

ผู้เข้าร่วมการทดลองแต่ละคนจะได้รับคำสั่งให้พิมพ์ตามประโยคที่ผู้วิจัยได้กำหนดไว้ล่วงหน้าเหมือนกัน คิดเป็นจำนวนการพิมพ์ 39 ประโยค และ 1,463 ตัวอักษร สำหรับผู้เข้าร่วมการทดลองหนึ่งคน โดยข้อความสำหรับให้พิมพ์ทดสอบนี้ผู้วิจัยนำมาจากชุดข้อมูล Mozilla Common Voice ซึ่งรวมทุกประโยคแล้วจะมีจำนวนคำเอกลักษณ์อยู่ที่ 221 คำ การทดสอบทำบนสมาร์ตโฟนระบบปฏิบัติการแอนดรอยด์ Samsung Galaxy Note 8 เครื่องเดียวกันทั้งหมด ผ่านแอปพลิเคชันต้นแบบ (รูปที่ 3) และแป้นพิมพ์เสมือนซึ่งอ้างอิงมาจาก Samsung Keyboard (รูปที่ 4) ทั้งนี้เพื่อให้ระบบต้นแบบบันทึกตำแหน่งที่ผู้ใช้สัมผัสโดนจริงๆ บนแป้นพิมพ์เสมือนและนำไปสู่การวิเคราะห์การพิมพ์ผิดในลำดับต่อไปได้

รูปที่ 5 และ 6 แสดงจุดสัมผัสบนแป้นพิมพ์เสมือนที่ผู้วิจัยบันทึกได้จากผู้เข้าร่วมการทดลอง จากรูปจะเห็นว่าจุดที่พิมพ์ผิด (กากบาทสีแดง) นั้น มักอยู่ในตำแหน่งบริเวณ

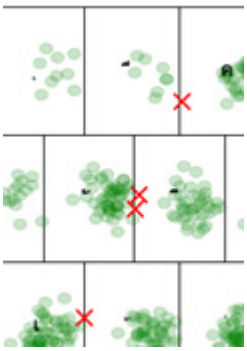
รูปที่ 4 Samsung Keyboard ต้นแบบของแป้นพิมพ์เสมือนที่ใช้ในการทดลองของงานวิจัยนี้



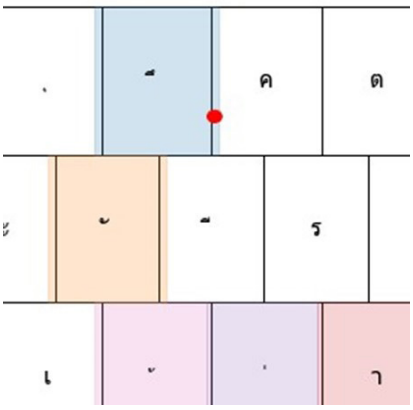
รูปที่ 5 ภาพแสดงตำแหน่งที่ผู้เข้าร่วมการทดลองสัมผัสจริงบนแป้นพิมพ์เสมือน จุดวงกลมสีเขียวคือจุดที่พิมพ์ถูก ในขณะที่กากบาทสีแดงคือพิมพ์ผิด

ขอบปุ่ม ดังนั้นหากผู้วิจัยนำแบบจำลองภาษา LSTM ที่ฝึกแล้วในหัวข้อที่ 2.2 มาช่วยคาดเดาปุ่มตัวอักษรถัดไปล่วงหน้า และทำการปรับขนาดพื้นที่เป้าหมายของตัวอักษรดังกล่าวให้ขยายกว้างขึ้นดังตัวอย่างในรูปที่ 7 ก็น่าจะช่วยลดอัตราการพิมพ์ผิดบริเวณขอบปุ่มเหล่านี้ได้

ข้อมูลการพิมพ์ที่ผู้วิจัยเก็บได้จากผู้เข้าร่วมการทดลองนั้น จะถูกทำความสะอาดก่อนนำไปใช้วิเคราะห์ในขั้นถัดไป การทำความสะอาดนี้เริ่มจากตัดการกดปุ่ม Shift ออกไป (บน



รูปที่ 6 ภาพขยายของรูปที่ 5 ซึ่งแสดงให้เห็นว่าจุดที่พิมพ์ผิดมักอยู่บริเวณขอบของปุ่ม



รูปที่ 7 วงกลมสีแดงคือจุดผิดพลาดที่สามารถแก้ไขให้ถูกต้องได้ด้วยการคาดเดาตัวอักษรและการขยายพื้นที่

แป้นพิมพ์เสมือนสำหรับการทดลอง จะไม่อนุญาตให้กดปุ่มอื่นที่ไม่เกี่ยวข้อง เช่น Backspace หรือ Spacebar) จากนั้นก็ตัดข้อมูลที่จำนวนครั้งการพิมพ์ไม่เท่ากับข้อมูลตัวอย่าง และข้อมูลที่พิมพ์ผิดมากกว่า 1 ครั้งติดกัน เนื่องจากการทดสอบมีวิธีการประมวลผลที่จำเป็นต้องรู้ว่าตัวอักษรที่ผู้ใช้ต้องการจะกดตัวต่อไปคืออะไร ทำให้ต้องตัดประโยคที่พิมพ์เกินหรือขาดออกไปจำนวนทั้งสิ้น 82 ประโยค จาก 273 ประโยค เหลือ 191 ประโยค คิดเป็นจำนวนครั้งการพิมพ์ที่ถูกตัดออกไป 3,106 ครั้ง เหลือเพียง 7,135 ครั้ง จาก 10,241 ครั้ง ดังแสดงในตารางที่ 1 และจากจำนวนครั้งการพิมพ์ที่เหลืออยู่นั้น มีตัวอักษรที่พิมพ์ถูกอยู่ที่ 6,878 ครั้ง คิดเป็น 96.39% และมีตัวอักษรที่พิมพ์ผิดอยู่ที่ 257 ครั้ง คิดเป็น 3.61%

ตารางที่ 1 ข้อมูลการพิมพ์ที่คงเหลือภายหลังการทำความสะอาดข้อมูล แยกตามผู้เข้าร่วมการทดลองแต่ละราย

Participant	Correct	Incorrect	Total
1	1,323	32	1,346
2	805	26	831
3	1,223	80	1,303
4	942	35	977
5	1,177	26	1,203
6	987	49	1,036
7	421	18	439
Total	**6,878**	**257**	**7,135**

ข้อมูลการพิมพ์ที่ผ่านการทำความสะอาดแล้วจะถูกนำไปประเมินด้วยวิธีการเดียวกับ Wassdahl และ Cho [9] ซึ่งแบ่งผลลัพธ์ออกเป็นตัวชี้วัด 4 ประเภท ได้แก่

1) พิมพ์ถูก (Correct) กรณีที่ผู้ใช้พิมพ์ตัวอักษรถูกต้องโดยไม่ว่าจะมีหรือไม่มีการปรับพื้นที่เป้าหมายมาช่วยเหลือผู้ใช้ก็ยังคงพิมพ์ถูก

2) พิมพ์ผิด (Incorrect) กรณีที่ผู้ใช้พิมพ์ตัวอักษรผิดโดยไม่ว่าจะมีหรือไม่มีการปรับพื้นที่เป้าหมายมาช่วยเหลือก็ยังคงพิมพ์ผิด นับรวมถึงจำนวนที่คลาดเคลื่อนด้วย

3) ช่วยเหลือ (Assisted) กรณีที่ผู้ใช้พิมพ์ตัวอักษรได้ถูกต้องเนื่องจากมีพื้นที่เป้าหมายที่ถูกขยายใหญ่ขึ้นมารองรับโดยหากไม่มีกระบวนการปรับพื้นที่เป้าหมายเข้ามาช่วยเหลือแล้ว ผลลัพธ์ที่ได้ก็จะผิด

4) คลาดเคลื่อน (Miss) กรณีที่ผู้ใช้พิมพ์ตัวอักษรผิดอันเนื่องมาจากพื้นที่เป้าหมายถูกปรับเปลี่ยนไปจากเดิม ซึ่งหากไม่ได้มีการปรับพื้นที่เป้าหมายแล้ว ตัวอักษรที่ได้ก็ควรจะถูก

ตัวชี้วัดทั้ง 4 นี้สามารถบ่งบอกถึงความสามารถในการแก้ไขการพิมพ์ผิดได้ด้วยการตั้งเป้าให้ผลรวมของ “พิมพ์ถูก” และ “ช่วยเหลือ” มีค่าสูงที่สุด ในขณะที่ “พิมพ์ผิด” น้อยที่สุด หรือก็คือผู้วิจัยต้องพยายามทำให้จำนวนผลรวมระหว่าง “พิมพ์ถูก” และ “ช่วยเหลือ” มีค่าสูงและห่างจากจำนวน “พิมพ์ผิด” ให้มากที่สุดตามสมการที่ (1) หรือดูจาก Net Gain ตามสมการที่ (2) ทั้งนี้ตัวแปร x ในสมการที่ (1) หมายถึงผลลัพธ์จากสมการ (correct + assisted) – incorrect หรือก็คือสมการนี้มีเป้าหมายคือการหา x ที่มีค่ามากที่สุดนั่นเอง

$$max(x) = (correct + assisted) - incorrect \quad (1)$$

$$Net\ Gain = assisted - miss \quad (2)$$

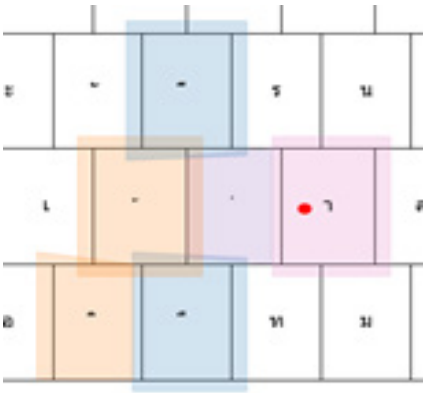
2.4 การขยายขนาดของปุ่ม

ในระบบปฏิบัติการแอนดรอยด์มีหน่วยวัดขนาดของหน้าจออยู่หลายแบบ แต่ในแอปพลิเคชันต้นแบบของงานวิจัยนี้ผู้วิจัยเลือกใช้หน่วยวัดที่เรียกว่า Density Independent Pixel (dip or dp) [10] ซึ่งมีคุณลักษณะ คือ มีความคงที่ของขนาด ไม่เปลี่ยนแปลงไปตามความหนาแน่นของพิกเซลบนหน้าจอ ทำให้ขนาดของปุ่มที่ได้มีความคงที่เสมอ โดยในงานวิจัยนี้ข้อมูลตัวนี้จะถูกอ้างถึงในหน่วย “จุด (Point; pt)” เพื่อให้ง่ายต่อการทำความเข้าใจและการนำไปใช้แสดงผลบนคอมพิวเตอร์ ทั้งนี้แป้นพิมพ์เสมือนที่ใช้ในงานวิจัยนี้มีขนาดความกว้างของปุ่มอยู่ที่ 58 หรือ 59 หรือ 61 จุด ขึ้นกับจำนวนปุ่มทั้งหมดที่มีในแถวนั้นๆ อนึ่ง จำนวนปุ่มในแต่ละแถวของแป้นพิมพ์เสมือนที่ไม่เท่ากันนี้เป็นการออกแบบปกติของแป้นพิมพ์เสมือนภาษาไทยแบบ 4 แถว บนสมาร์ตโฟน

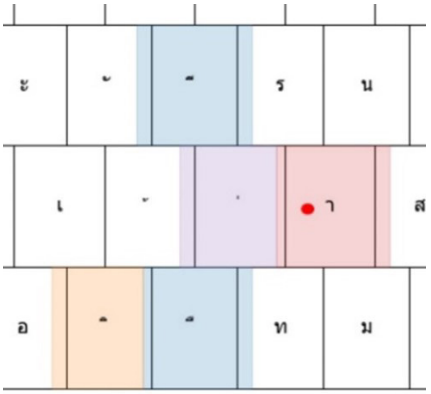
ในส่วนของการพิจารณาปรับขนาดของปุ่มนั้น มีประเด็นที่ต้องตัดสินใจ คือ 1) ทิศทางการขยายขนาดปุ่ม 2) ขนาดที่สามารถขยายได้ หน่วยเป็นจุดของหน้าจอ (Points; pt) และ 3) จำนวนปุ่มที่ต้องการขยาย

2.4.1 ทิศทางการขยายขนาดปุ่ม

ทิศทางการขยายขนาดปุ่มนั้นสามารถทำได้ทั้งแบบเฉพาะเจาะจงในแนวนอนหรือแนวตั้งเพียงอย่างเดียว หรือแบบขยายทั้งสองแนวพร้อมกันก็ได้ ในงานวิจัยนี้ผู้วิจัยเลือกปรับเฉพาะความกว้างของปุ่มในแนวนอนเท่านั้น เนื่องจากการปรับขนาดความสูงในแนวตั้งด้วยจะทำให้รูปทรงของปุ่มเปลี่ยนแปลงมากเกินไป และอาจไม่เป็นรูปสี่เหลี่ยมดังตัวอย่างในรูปที่ 8 (กรอบสีต่างๆ แสดงขนาดพื้นที่ที่ถูกขยายไปของแต่ละปุ่ม โดยแต่ละสีไม่มีความหมายพิเศษนอกจาก



รูปที่ 8 ปุ่มที่พยายามปรับทั้งความกว้างและความสูงพร้อมกัน จะได้ขนาดพื้นที่ออกมาไม่เป็นสี่เหลี่ยมมุมฉาก ทำให้ยากต่อการใช้งาน ทั้งนี้จุดสีแดงคือจุดสัมผัสของผู้ทดลอง



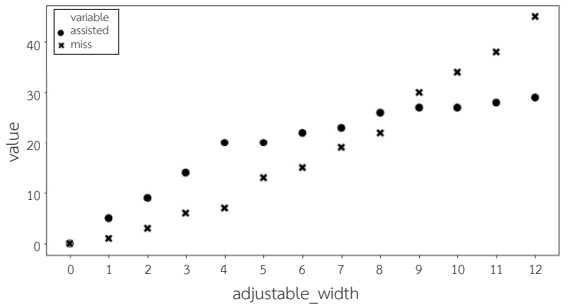
รูปที่ 9 ปุ่มที่ผ่านการปรับเฉพาะความกว้าง เมื่อจุดสีแดงคือจุดสัมผัสของผู้ทดลอง

ทำให้เห็นขอบเขตของปุ่มชัดเจนขึ้น และทำให้แยกออกจากกันได้เมื่อขอบเขตอยู่ติดกัน)

อีกทางเลือกหนึ่ง คือ การขยายขนาดปุ่มโดยคงสัดส่วนความเป็นสี่เหลี่ยมไว้ [11] ซึ่งจะมีผลทำให้ปุ่มเสียความเป็นแถวไป โดยจากการศึกษาของ Himberg และคณะ [12] พบว่า “การเปลี่ยนแปลงของส่วนต่อประสานกับผู้ใช้ที่มากเกินไปจะสร้างความสับสนและความรำคาญให้กับผู้ใช้” และ “ความขัดแย้งระหว่างโมเดลทางความคิด (Mental Model) ของผู้ใช้กับรูปลักษณ์ของแป้นพิมพ์ (Visual Appearance) จะสร้างความรู้สึกไม่สบายใจให้กับผู้ใช้” นอกจากนี้ Himberg และคณะ [12] ยังได้สรุปแนวทางการเปลี่ยนภาพลักษณ์ ซึ่งสามารถนำมาเป็นกรอบแนวคิดของงานวิจัยนี้ได้ กล่าวคือ “การเปลี่ยนแปลงควรที่จะกระทำในรูปแบบที่ไม่ทำให้ผู้ใช้ไขว้เขว เช่น เส้นกรอบของแต่ละปุ่มสามารถทำให้เป็นแบบโปร่งใส ปุ่มควรจะเล็กกว่าส่วนพื้นที่ตรวจสัมผัส (Detection Area) ปุ่มควรจะเป็นรูปทรงมาตรฐานอย่างสี่เหลี่ยม หรือวงกลม” นอกจากนี้ จากการศึกษาของ Gunawardana และคณะ [13] พบว่า ปุ่มที่มองเห็นกับพื้นที่สัมผัสจริงที่มีความสัมพันธ์กันจะให้ผลลัพธ์ที่ดีกว่า เช่น ตัวอย่างในรูปที่ 9 ซึ่งแสดงการขยายปุ่มออกไปทางด้านข้างทับอยู่บนขนาดปุ่มเดิม โดยเมื่อเปรียบเทียบกับรูปที่ 8 ซึ่งเป็นการขยายทั้งความกว้าง และความสูงด้วย จะเห็นว่ารูปที่ 8 พบปัญหามีพื้นที่ติดกันมากกว่า 1 ปุ่ม และปุ่มที่ได้ไม่เป็นรูปทรงสี่เหลี่ยมมุมฉาก

2.4.2 ขนาดที่สามารถขยายได้

ในงานวิจัยนี้ขนาดที่ขยายได้ใช้หน่วยเป็นจุด (Points; pt) ซึ่งผู้วิจัยยังไม่พบการศึกษาเรื่องนี้ในแป้นพิมพ์เสมือนภาษาไทย ผู้วิจัยจึงทำการทดสอบด้วยขนาดการขยายที่แตกต่างกันตั้งแต่ 0–12 จุด และเนื่องจากผลลัพธ์ที่ได้จากแบบจำลองภาษา LSTM (หัวข้อที่ 2.2) มีลักษณะเป็นค่าความน่าจะเป็น ผู้วิจัยจึงใช้วิธีกำหนดค่าขนาดที่จะขยายเป็น “ค่าสูงสุดที่ขยายได้” ด้วยหลักการคิดดังนี้ 1) หากปุ่มที่ต้องการขยายไม่มีปุ่มอื่นที่ต้องขยายเช่นกันอยู่ที่ด้านซ้าย หรือขวา ให้ใช้ขนาดสูงสุดที่เป็นไปได้ในด้านนั้น หรือ 2) ถ้ามีปุ่มอื่นที่ต้องการขยายอยู่ติดกันในด้านใด ให้นำความน่าจะเป็นของทั้งสองปุ่มมาเปรียบเทียบว่าปุ่มใดมีความน่าจะเป็นสูงกว่าก็จะได้พื้นที่ของฝั่งที่มีความน่าจะเป็นน้อยกว่าตามสูตรวิธีการคิดตามสมการที่ (3) ยกตัวอย่างเช่น ถ้ากำหนดให้มีค่าสูงสุดที่ขยายได้เป็น 4 และมีสองปุ่มที่ความน่าจะเป็นอยู่ที่ 0.6 และ 0.2 แล้ว ปุ่มฝั่งที่มีความน่าจะเป็นสูงกว่าจะขยายเข้ามากินพื้นที่ของปุ่มฝั่งที่มีความน่าจะเป็นน้อยกว่าอยู่ (4*0.5) + ((4*0.5) * (0.6/(0.6+0.2))) = 3.5



รูปที่ 10 จำนวนครั้งที่ช่วยเหลือ (O) กับคลาดเคลื่อน (X) เทียบกับขนาดการขยายสูงสุดตั้งแต่ 0 ถึง 12 จุด

$$adj = (adj_{base} * 0.5) + \left(adj_{base} * 0.5 * \frac{prob_A}{prob_A + prob_A} \right) \quad (3)$$

2.4.3 จำนวนปุ่มที่จะขยาย

ในงานวิจัยนี้ผู้วิจัยทำการทดลองขยายขนาดปุ่มจำนวนตั้งแต่ 1 ปุ่มถึง 12 ปุ่ม โดยใช้ตัวชี้วัดเป้าหมาย Objective และ Net Gain ในการพิจารณาว่าจำนวนปุ่มที่ขยายจำนวนเท่าใดจึงจะได้ผลลัพธ์ที่ดีที่สุด

3. ผลการทดลอง

ผลการทดลองเปรียบเทียบข้อมูลการพิมพ์ของผู้เข้าร่วมการทดลอง 7 คน (ผู้เข้าร่วม 1 คน ทำการทดลองพิมพ์เพียงครั้งเดียว) กับการจำลองการขยายขนาดปุ่มด้วยค่าการขยายที่แตกต่างกันตั้งแต่ 0 ถึง 12 จุด สำหรับตัวชี้วัดสองตัว ได้แก่ ตัวชี้วัด Objective [คำนวณตามสมการที่ (1)] และ Net Gain [คำนวณตามสมการที่ (2)] สามารถสรุปได้ดังตารางที่ 2 และรูปที่ 10 ซึ่งจะเห็นว่าความกว้างในการขยายสูงสุดที่ 4 จุด ให้ผลลัพธ์ดีที่สุด เนื่องจากมีระยะห่างระหว่าง “ช่วยเหลือ” (20) กับ”คลาดเคลื่อน” (7) สูงสุดที่ 13 ครั้ง จากการพิมพ์ผิดรวม 257 ครั้ง คิดเป็นอัตราลดของการพิมพ์ผิดอยู่ที่ 13/257 หรือ 5.05%

ในส่วนของจำนวนปุ่มที่จะขยายนั้น ในทางปฏิบัติไม่จำเป็นต้องจำกัดการคาดการณ์อยู่เฉพาะ 6 ปุ่มแรก ที่มีความน่าจะเป็นสูงสุดแต่อย่างใด จากการทดลองปรับขยายขนาดปุ่มจำนวนตั้งแต่ 1 ถึง 12 ปุ่ม (ที่ขนาดการขยายสูงสุด 4 จุด อันเป็นค่าการขยายที่ดีที่สุดจากตารางที่ 2) ผลการทดลองดังแสดงในตารางที่ 3 พบว่า ตัวชี้วัดเป้าหมาย Objective

ตารางที่ 2 ผลการทดลองค่าตัวชี้วัดต่างๆ ตามขนาดการขยายปุ่มสูงสุด (คอลัมน์ Width) ที่ต่างกันตั้งแต่ 0 ถึง 12 จุด

Width (pt)	Correct	Incorrect	Assisted	Miss	Objective	Net Gain
0	6,878	52	0	0	6,826	0
1	6,877	48	5	1	6,834	4
2	6,875	46	9	3	6,838	6
3	6,872	44	14	6	6,842	8
4	6,871	39	20	7	6,852	13
5	6,865	45	20	13	6,840	7
6	6,863	45	22	15	6,840	7
7	6,859	48	23	19	6,834	4
8	6,856	48	26	22	6,834	4
9	6,848	55	27	30	6,820	-3
10	6,844	59	27	34	6,812	-7
11	6,840	62	28	38	6,806	-10
12	6,833	78	29	45	6,784	-16
Median	6,863	48	22	15	6,834	4
Mean	6,860	51	19	18	6,828	1
SD	14.37	9.79	9.06	14.37	17.57	7.88

และ Net Gain นั้น สูงสุดที่การขยายปุ่มจำนวนเพียง 2 ปุ่ม โดยมีค่า Objective และ Net Gain สูงสุดที่ 6,854 และ 14 ตามลำดับ แต่เมื่อเทียบค่าสูงสุดนี้กับค่าที่ได้จากการขยายปุ่มจำนวน 6 ปุ่ม ซึ่งมี Objective และ Net Gain คือ 6,852 และ 13 ตามลำดับ จะเห็นว่าความแตกต่างนั้น น้อยมากจนไม่มีนัยสำคัญทางสถิติ

ทั้งนี้ งานวิจัยชิ้นนี้ถูกพัฒนาขึ้นบนเครื่องคอมพิวเตอร์ของผู้วิจัยซึ่งมีรายละเอียด คือ Microsoft Surface Book 2, CPU Intel Core i7-8650U, RAM 16GB และ GPU NVIDIA GeForce GTX 1060 (6GB) โดยผลลัพธ์สำหรับแบบจำลองภาษา LSTM จะถูกเก็บเป็นไฟล์ Pickle และไฟล์แบบจำลอง H5 สำหรับนำไปใช้งานต่อ สรุปจากผลการทดลองทั้งหมดซึ่งผู้วิจัยทดลองขนาดการขยายปุ่มที่ต่างกันตั้งแต่ขนาด 0 ถึง 12 จุด และทดลองใช้แบบจำลองทางภาษา LSTM ในการคาดเดาปุ่มถัดไปจำนวนตั้งแต่ 1 ถึง 12 ปุ่ม โดยใช้ตัวชี้วัด คือ ค่า Net Gain นำมาสู่ข้อสรุปของงานวิจัยชิ้นนี้ว่า “การใช้แบบจำลองทางภาษา LSTM สามารถลดความผิดพลาดจากการพิมพ์บนแป้นพิมพ์เสมือนภาษาไทยบนสมาร์ตโฟนได้ โดยในกรณีศึกษาที่ใช้ความกว้างในการขยายปุ่มสูงสุดที่ 4 จุด และคาดเดาปุ่มที่จะขยายจำนวน 6 ปุ่ม นั้น พบว่าสามารถช่วยลดความผิดพลาดได้ 13 ครั้ง จากการพิมพ์ผิดทั้งหมด 257 ครั้ง คิดเป็น 5.05%”

4. อภิปรายผลและสรุป

ประเด็นอื่นๆ ที่น่าสนใจซึ่งผู้วิจัยค้นพบจากการทำงานวิจัยชิ้นนี้ ซึ่งน่าจะมีประโยชน์ต่อการพัฒนาต่อยอดในอนาคต ถูกอภิปรายไว้ในหัวข้อย่อย 4.1 ถึง 4.3 ต่อไปนี้

4.1 โมเดลทางภาษา พารามิเตอร์ และคลังศัพท์

ในงานวิจัยนี้ผู้วิจัยทำการทดลองโดยอิงกับแบบจำลองภาษา LSTM เป็นหลักโดยได้ทดลองปรับค่าพารามิเตอร์ต่างๆ ที่เกี่ยวข้อง อาทิ ปรับจำนวนตัวอักษรที่เป็นอินพุตของแบบจำลองตั้งแต่ 4, 5, 8, 12 และ 24 (ผลพบว่า ได้ความแม่นยำใกล้เคียงกันอยู่ที่ 40–50%) ปรับจำนวนโหนดและจำนวนชั้นของแบบจำลอง LSTM (ไม่พบผลลัพธ์ต่างกันชัดเจน) เปลี่ยนคลังศัพท์สำหรับฝึกสอนแบบจำลองภาษา (พบว่า ถ้าใช้ Thai

ตารางที่ 3 ผลการทดลองค่าตัวชี้วัดต่างๆ ตามขนาดการขยายปุ่มสูงสุดที่ 4 จุด แต่ใช้การคาดเดาจำนวนปุ่มถัดไป (คอลัมน์ Guessed) ต่างกัน ตั้งแต่คาดเดาปุ่มถัดไป 1 ปุ่ม จนถึงคาดเดาปุ่มถัดไป 12 ปุ่ม

Width (pt)	Guessed	Correct	Incorrect	Assisted	Miss	Objective	Net Gain
4	1	6,876	40	14	2	6,850	12
4	2	6,874	38	18	4	6,854	14
4	3	6,872	39	19	6	6,852	13
4	4	6,870	41	19	8	6,848	11
4	5	6,871	39	20	7	6,852	13
4	6	6,871	39	20	7	6,852	13
4	7	6,871	39	20	7	6,852	13
4	8	6,871	41	18	7	6,848	11
4	9	6,871	41	18	7	6,848	11
4	10	6,872	41	17	6	6,848	11
4	11	6,873	40	17	5	6,850	12
4	12	6,872	40	18	6	6,850	12
Median	6.5	6,871.5	40.0	18.0	6.5	6,850.0	12.0
Mean	6.50	6,872.00	39.83	18.17	6.00	6,850.33	12.17
SD	3.45	1.58	0.99	1.62	1.58	1.97	0.99

National Corpus จะได้ผลลัพธ์ที่แย่กว่า BEST I Corpus มาก คาดว่าเป็นเพราะจำนวนข้อมูลในการเรียนรู้ที่น้อยเกินไป)

ในส่วนของแบบจำลองทางภาษา แม้ว่างานวิจัยนี้จะเลือกใช้ LSTM เป็นหลักเนื่องจากความยืดหยุ่นของอินพุตและความแม่นยำขั้นต้นที่ดีกว่า แต่จากการทดสอบกับ Markov Chain ที่ถูกฝึกสอนด้วย BEST I Corpus เช่นเดียวกัน ผลลัพธ์ที่ได้ก็ด้อยกว่า LSTM เพียงเล็กน้อย อีกทั้งแบบจำลองทั้งสองก็ให้ผลลัพธ์ที่สอดคล้องกันคือขนาดการขยายปุ่มสูงสุดที่ 4 จุด จะได้ผลลัพธ์ดีที่สุด

สำหรับแนวทางการวิจัยต่อยอดนั้น ผู้วิจัยมีความเห็นว่าควรทำการทดลองกับแบบจำลองทางภาษา หรือแบบจำลองการเรียนรู้เชิงลึกตัวอื่น เพื่อสำรวจผลลัพธ์ที่ได้ว่าแตกต่างจากผลในงานวิจัยนี้อย่างมีนัยสำคัญหรือไม่ อีกทั้งควรมุ่งไปในทางการหาคลังศัพท์ภาษาไทยอื่นๆ มาทดสอบเพิ่มเติมให้มากกว่านี้ ซึ่งน่าจะเห็นผลการเปลี่ยนแปลงได้ชัดเจนกว่าการปรับจูนพารามิเตอร์ในแบบจำลอง และอีกวิธีการหนึ่งที่น่าศึกษา คือ การฝึกแบบจำลองภาษาด้วยคำศัพท์ที่ผ่านการตัดคำมาเรียบร้อยแล้ว ซึ่งผู้วิจัยคาดการณ์ว่าน่าจะช่วยเพิ่มความแม่นยำของแบบจำลองภาษาได้ แต่ก็แลกมาด้วยความยุ่งยากในการใช้งานคือเราจำเป็นต้องตัดคำก่อนนำอินพุตที่ผู้ใช้พิมพ์ไปเข้าสู่แบบจำลองภาษา ซึ่งกระบวนการตัดคำภาษาไทยอยู่นอกเหนือจากขอบเขตของงานวิจัยนี้

4.2 ประโยคทดสอบและผู้ทดสอบ

ประโยคที่ผู้วิจัยนำมาให้ผู้เข้าร่วมการทดลองทดสอบพิมพ์ในงานวิจัยนี้ เป็นประโยคที่ใช้คำค่อนข้างเป็นทางการและใช้คำตามพจนานุกรม ซึ่งอาจยังไม่ครอบคลุมรูปแบบประโยคที่ผู้ใช้คนไทยมีการพิมพ์ใช้งานจริงในโปรแกรมต่างๆ ของสมาร์ตโฟน ดังนั้นในการทดลองต่อยอดจึงควรเพิ่มประโยคพิมพ์ทดสอบที่ไม่เป็นทางการซึ่งมีคำแสลงต่างๆ ร่วมด้วย ซึ่งผลลัพธ์ที่ได้น่าจะนำไปสู่การสร้างเป็นผลิตภัณฑ์สุดท้าย (Consumer Product) ได้ตรงตามเป้าประสงค์ของแป้นพิมพ์เสมือนบนสมาร์ตโฟนมากขึ้น

อีกจุดหนึ่งที่ต้องมีการศึกษาเพิ่มเติม คือ ปัจจัยเกี่ยวกับตัวผู้ใช้เองที่ทำให้เกิดการพิมพ์ผิดบนสมาร์ตโฟน โดยในขณะที่งานวิจัยชิ้นนี้มุ่งเป้าไปที่การพิมพ์ผิดที่เกิดจากขนาดปุ่มที่เล็กเกินไป ในความเป็นจริงยังมีปัจจัยอื่นๆ อาทิ ขนาดมือของผู้ทดสอบ ลักษณะการจับสมาร์ตโฟน ความใหญ่ของนิ้ว ความยาวของเล็บ หรือแม้แต่อัตราการพิมพ์ผิดอันเป็นลักษณะเฉพาะตัวของผู้ใช้แต่ละคน โดยสำหรับงานวิจัยชิ้นนี้ผู้วิจัยทำการควบคุมตัวแปรในส่วนของผู้ใช้ โดยกำหนดให้ผู้เข้าร่วมการทดลองจับสมาร์ตโฟนด้วยมือทั้งสองมือและพิมพ์ด้วยนิ้วโป้งทั้งสองข้าง ซึ่งผลการทดลองในผู้เข้าร่วมการทดลองทั้ง 7 ราย พบว่า เพศชายและเพศหญิงมีอัตราการพิมพ์ผิดอยู่ที่ 49/2500 ครั้ง (1.96%) และ 208/4378 ครั้ง (4.75%) ตามลำดับ

4.3 การขยายปุ่ม

ในงานวิจัยนี้ผู้วิจัยสรุปจากผลการทดลองในรูปที่ 10 ว่าขนาดการขยายปุ่มต้องเหมาะสมพอดี ไม่มากหรือน้อยเกินไป อย่างไรก็ตาม เรื่องของการขยายขนาดปุ่ม และทิศทางการขยายปุ่มนั้น ยังมีประเด็นให้ศึกษาต่อได้อีกมาก เช่น การขยับปุ่มที่อยู่ติดกันออกไปเพื่อให้มีพื้นที่สำหรับการขยายปุ่มเพิ่มมากขึ้น การแสดงให้ผู้ใช้เห็นภาพของปุ่มที่ถูกขยายขึ้นขณะทดสอบ การเปลี่ยนวิธีการแบ่งพื้นที่ของปุ่มไปจากสมการที่ (3)

นอกจากนี้ในการวิจัยต่อยอดผู้วิจัยมีแผนจะทำการทดสอบเพิ่มเติม โดยพัฒนาแอปพลิเคชันแอนดรอยด์ต้นแบบอีกตัวที่สามารถทำการขยายขนาดปุ่มบนแป้นพิมพ์เสมือนได้แบบทันทีทันใดตามวิธีการที่ได้นำเสนอในผลงานวิจัยชิ้นนี้ ทั้งนี้ เพื่อให้สามารถทดสอบประสิทธิภาพและปัญหาที่พบในการใช้งานจริง (Usability Testing) โดยจะเป็นการทดลองในลักษณะ A/B Testing ที่เปรียบเทียบผลระหว่างผู้ใช้กลุ่มที่พิมพ์ด้วยแป้นพิมพ์เสมือนปกติของระบบปฏิบัติการแอนดรอยด์ และผู้ใช้กลุ่มที่พิมพ์ด้วยแป้นพิมพ์เสมือนที่มีการขยายขนาดปุ่มดังที่นำเสนอไปในงานวิจัยชิ้นนี้

เอกสารอ้างอิง

[1] Advanced Info Services Public Company Limited. (2020, July). *Operating and Financial Report*. [Online] (in Thai). Available: http://

investor-th.ais.co.th/operational_highlight.html

[2] J. H. Kim, L. Aulck, O. Thamsuwan, M. C. Bartha, and P. W. Johnson, "The effects of virtual keyboard key sizes on typing productivity and physical exposures," in *Human Factors and Ergonomics Socety Annual Meeting*, 2013.

[3] T. Baldwin and J. Y. Chai, "Towards online adaptation and personalization of key-target resizing for mobile devices," presented at the IUI, 2012.

[4] D. Weir, S. Rogers, R. Murray-Smith, and M. Löchtefeld, "A user-specific machine learning approach for improving touch accuracy on mobile devices," in *UIST*, Cambridge, Massachusetts, USA, 2012.

[5] National Statistical Office of Thailand, "Household information technology usage survey 2018 Q1," Ministry of Digital Economy and Society, Bangkok, Thailand, 2018 (in Thai).

[6] Faculty of Arts. (2021, March). *Thai National Corpus*. [Online] (in Thai). Available: http://www.arts.chula.ac.th/ling/tnc/

[7] Human Language Technology Laboratory. (2021, March). *BEST by Human Language Technology Laboratory, National Electronics and Computer Technology Center*. [Online]. Available: http://thailang.nectec.or.th/archive/indexdca0.html?q=node/21

[8] Mozilla. (2020, November). *Mozilla Common Voice GitHub Page*. [Online]. Available: https://github.com/common-voice/common-voice/tree/main/server/data/th

[9] H. Wassdahl and K. Cho, "Personalized stroke order dependent keyboard with adaptive key-target areas using user generated data," in *HCI Korea*, 2016.

[10] Google, Inc. (2021, April). *Pixel density*. [Online]. Available: https://material.io/design/layout/pixel-density.html#density-independence

[11] N. Piyapramote. (2021, March). *Keyboard ManMan Google Play Store Page*. [Online]. Available: https://play.google.com/store/apps/details?id=net.siamdev.nattster.manman&hl=th&gl=US

[12] J. Himberg, J. Häkkilä, and J. Mäntyjärvi, "On-line personalization of a touch screen based keyboard," presented at the IUI, Miami, Florida, USA, 2003.

[13] A. Gunawardana, T. Paek, and C. Meek, "Usability guided key-target resizing for soft keyboards," presented at the IUI, Hong Kong, China, 2010.