

การทำนายความสัมพันธ์ความเค้นและความเครียดของดินเหนียวผสมซีเมนต์ ด้วยโครงข่ายประสาทแบบเกิดซ้ำชนิด LSTM Prediction of The Stress-Strain Curve of Cement Admixed Clay by using LSTM Recurrent Neural Network

ไปรยา รัตน์กิจกุล^{1,*} และ สมโพธิ อยู่วิ¹

¹ สาขาวิชาวิศวกรรมโยธา คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี จ.กรุงเทพมหานคร

*Corresponding author; E-mail address: praiya.722@mail.kmutt.ac.th

บทคัดย่อ

งานวิจัยนี้ได้พัฒนาโครงข่ายประสาทแบบเกิดซ้ำชนิด LSTM ทำนายความเค้นและความเครียดของการทดสอบดินเหนียวผสมซีเมนต์ด้วยวิธีแรงอัดสามแกน ตัวแปรที่ใช้ในการศึกษา ได้แก่ อัตราส่วนผสมของซีเมนต์ น้ำและดิน ค่าความเค้นประสิทธิผลเฉลี่ย ค่าความเค้นเบี่ยงเบน และค่าความเครียดในแนวตั้ง การศึกษานี้จะนำเสนอสถาปัตยกรรมที่เหมาะสมของโครงข่ายประสาทเทียมแบบ LSTM และเปรียบเทียบกับโครงข่ายประสาทเทียมแบบอื่นๆ จากผลการศึกษาพบว่าโครงข่ายประสาทแบบเกิดซ้ำชนิด LSTM เป็นแบบจำลองที่ดีที่สุดเมื่อเทียบกับแบบจำลองแบบ GRU และ SimpleRNN แบบจำลอง LSTM ที่มีการคิดขั้นของเวลาเท่ากับสองสามารถทำนายค่าความเค้นและความเครียดของดินซีเมนต์ได้มีความผิดพลาดสัมบูรณ์ที่ 4 เปอร์เซ็นต์

คำสำคัญ: ความเค้นและความเครียด, โครงข่ายประสาทแบบเกิดซ้ำ, ดินเหนียวผสมซีเมนต์

This study used the recurrent type of neural network LSTM to predict the stress strain of cement mixed clay in the triaxial test. The features for simulation were mixing ratio water and cement, mean stress, deviator stress and vertical strain. The best architecture for the neural network was proposed in this study with the lowest error. The LSTM was the best model with the lowest error among other types of recurrent neural network, GRU and SimpleRNN. LSTM with 2 time steps was the best architecture to predict the stress strain characteristic of clay mixed. The prediction model can simulate the stress-strain relationship with an average absolute error of 4 %.

Keywords: Clay with Soil cement, LSTM Recurrent Neural Network, Stress and strain

1. คำนำ

การใช้ดินซีเมนต์เป็นวิธีหนึ่งที่ยอมรับใช้ในการปรับปรุงดินที่มีกำลังและสติฟเนสที่ต่ำ ดินที่ผสมซีเมนต์จะเป็นดินที่มีกำลังสูงและสติฟเนสที่สูงกว่าดินธรรมชาติอันเนื่องมาจากแรงยึดเกาะของซีเมนต์ที่เกิดขึ้นในดิน ดินซีเมนต์จึงถูกนำไปใช้ในงานวิศวกรรมโยธาอาทิเช่น รองรับคันทางถนน กำแพงกันดิน การป้องกันน้ำซึมผ่านเข้าสู่อุโมงค์ เป็นต้น การที่วิศวกรต้องการที่จะออกแบบการปรับปรุงคุณภาพดินด้วยดินซีเมนต์จะต้องทำนายการเสยุรูปของดินซีเมนต์ที่จะเกิดขึ้น การจะทำนายพฤติกรรมดินซีเมนต์ซึ่งเป็นการผสมระหว่างปูนซีเมนต์กับดินหรือวัสดุที่อยู่ในพื้นที่ ในปี ค.ศ 1994 – 1997 การวิจัยส่วนมากจะนำเสนอเกี่ยวกับพฤติกรรมของการใช้ปูนซีเมนต์ในการปรับปรุงดิน [1-7]

การทำนายการเสยุรูปโดยทั่วไปจะใช้วิธีการใช้แบบจำลองคอนสแตนต์ที่ฟ [8-10] เช่น การใช้แบบจำลองแบบ Linear Elastic หรือแบบจำลองที่ซับซ้อน เช่น แบบจำลองแบบ Hypoplastic การที่จะได้สมการในแบบจำลองจำเป็นที่จะต้องตั้งสมมติฐานของสมการ และสอบเทียบกับผลการทดลองว่ามีความถูกต้องในระดับใด แต่ปัญหาที่ตามมา คือไม่สามารถระบุแน่ชัดว่าสมการเริ่มต้นที่เลือกใช้มีความเหมาะสมกับดินที่ต้องการจำลองหรือไม่ ทำให้ไม่มีความยืดหยุ่นในการทำนายพฤติกรรมของดินดังกล่าว การทำงานของนักวิจัยจะเป็นในลักษณะการทดสอบและดูความคลาดเคลื่อน (Trial and error) การใช้ปัญญาประดิษฐ์ในการทำนายพฤติกรรมของดินเป็นทางเลือกที่น่าสนใจเพราะสามารถมีสมการที่ซับซ้อนได้ถึงหลายแสนตัวแปรในแบบจำลองการเรียนรู้แบบลึก (Deep Learning) การหาสมการที่เหมาะสมในการทำนายกับข้อมูลสามารถให้คอมพิวเตอร์ได้เรียนรู้และหาแบบจำลองที่มีความเหมาะสมในการจำลองพฤติกรรมด้านการเสยุรูปของวัสดุ การให้คอมพิวเตอร์ได้เรียนรู้ได้ด้วยตัวเองจะนำไปสู่การได้แบบจำลองที่มีความเหมาะสมที่สุด (Optimization Model)

ในปัจจุบันมีการใช้ปัญญาประดิษฐ์อย่างแพร่หลายในทุกวงการในการสร้างแบบจำลองเพื่อทำนายเหตุการณ์ในอนาคต เช่น ตลาดหุ้น กระแสเงินสดในธนาคาร การเสยุรูปของเสาเข็ม เป็นต้น ปัญญาประดิษฐ์เป็นการ

สอนให้คอมพิวเตอร์เรียนรู้ เลียนแบบการทำงานสมองของมนุษย์ ซึ่งจะเรียกว่า Artificial Neural Network (ANN) โดย Neural network นั้นสามารถจำแนกได้อีกหลายประเภท เช่น Feedforward Networks, Recurrent Neural Networks (RNNs) และ Long Short-term Memory networks (LSTM) เป็นต้น ในงานเทคนิคธรณีมีการใช้ปัญญาประดิษฐ์ ได้แก่ genetic programming, polynomial regression of evolution, ANN and LSTM. ปัญญาประดิษฐ์จะถูกใช้มากในการพัฒนาโมเดลจำลอง ในปี 1995 และ 1999 มีการใช้ ANN ในการทำนายพฤติกรรมของดิน [11-12] นอกจากนี้ในปี 2020 และ 2021 ได้มีการใช้ LSTM ในการทำนายพฤติกรรมของดิน [13-14] จากงานวิจัยในอดีตสามารถใช้ ANN ในการทำนายพฤติกรรมการเสียรูปของทรายได้เป็นอย่างดี ทั้งแบบการให้น้ำหนักครั้งเดียวและแบบวัฏจักร

โดยในบทความนี้มีจุดประสงค์ที่จะพัฒนาโมเดล LSTM สำหรับทำนายความเค้นและความเครียดของดินซีเมนต์ การพัฒนาแบบจำลองจะสร้างบนพื้นฐานของโปรแกรม TensorFlow [15] โดยมี Keras [16-18] เป็น Backend การศึกษาครั้งนี้ใช้ผลการทดสอบดินซีเมนต์ด้วยวิธีแรงอัดสามแกน (triaxial test) ทั้งหมด 8 ตัวอย่าง โดยตัวแปรที่ใช้ในการศึกษา ได้แก่ อัตราส่วนผสมของซีเมนต์น้ำและดิน (mixing ratio water and cement) ค่าความเค้นประสิทธิผลเฉลี่ย (mean effective stress) ค่าความเค้นเบี่ยงเบน (deviatoric stress) และค่าความเครียดในแนวตั้ง (vertical strain) สิ่งที่แตกต่างกันจากงานวิจัยอื่นในอดีตมีดังต่อไปนี้

- งานวิจัยนี้เป็นงานวิจัยแรกที่ใช้ ANN ในการทำนายการเสียรูปของดินซีเมนต์ ซึ่งมีความท้าทายในการใช้พารามิเตอร์เกี่ยวกับแรงยึดเหนี่ยว (Bonding) ในการสร้างแบบจำลอง แบบจำลองที่พัฒนาขึ้นจะต้องทำนายพฤติกรรมของดินเหนียวผสมซีเมนต์ในส่วนผสมที่ต่างกัน
- งานวิจัยในอดีตมีความพยายามใช้แบบจำลอง LSTM ในการทำนายพฤติกรรมของดิน แต่ไม่ได้ศึกษาถึงการนำผลของการนำค่าความเค้นและความเครียดในอดีตมาใช้ในการสร้างแบบจำลองความเค้นและความเครียดที่เกิดขึ้นในอดีตมีผลอย่างมากต่อความเค้นและความเครียดของดิน งานวิจัยนี้จะศึกษาถึงผลกระทบของการใช้ค่าความเค้นและความเครียดของขั้นก่อนหน้า (Previous step) ในการสร้างแบบจำลอง
- งานวิจัยนี้มีการเปรียบเทียบค่าความคลาดเคลื่อนในการจำลองระหว่างแบบจำลอง ANN แบบต่างๆ เช่น SimpleRNN LSTM Gated recurrent unit (GRU) และหาสถาปัตยกรรมที่เหมาะสมที่จะให้ค่าความคลาดเคลื่อนที่ต่ำที่สุด

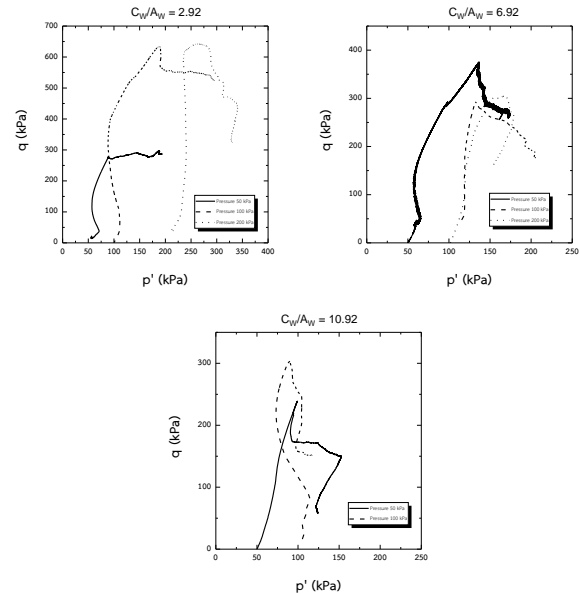
2. การเตรียมข้อมูล

2.1 ที่มาของข้อมูลที่ใช้ในการทดสอบ

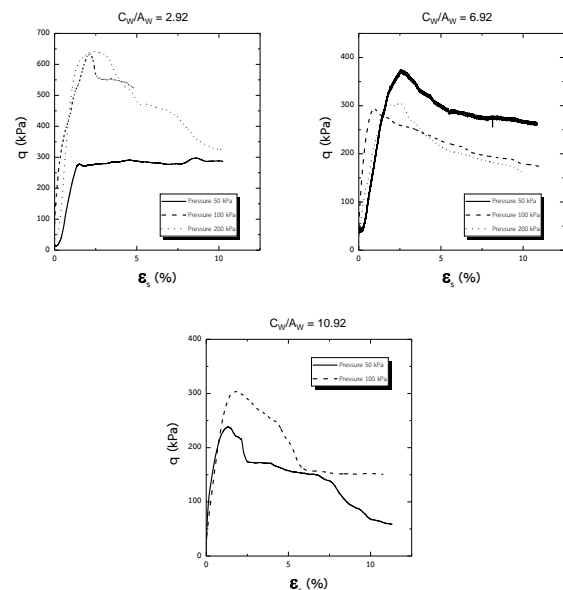
ข้อมูลที่นำมาใช้ในการศึกษาเป็นข้อมูลดินซีเมนต์ จากโครงการก่อสร้างกำแพงกันดินริมแม่น้ำป่าสัก โดยการปรับปรุงดินโดยใช้ปูนซีเมนต์

ที่ทดสอบโดย triaxial test จำนวนข้อมูลทั้งหมด 8 ตัวอย่าง โดยอัตราส่วนน้ำต่อซีเมนต์ ได้แก่ 2.92 6.92 และ 10.92 โดยความเค้นที่ใช้ในการทดสอบได้แก่ 50 100 และ 200 [19]

2.2 ข้อมูลที่ใช้ในการทดสอบ



รูปที่ 1 ความสัมพันธ์ระหว่าง p' - q ของข้อมูล



รูปที่ 2 ความสัมพันธ์ระหว่าง ϵ_v (%) - q ของข้อมูล

การศึกษานี้จะใช้ผลจากการทดสอบการอัดแบบสามแกน (Triaxial Test) แบบ consolidated undrained test (CIU) ของดินเหนียวป่าสักผสมปูนซีเมนต์ จะมีความแตกต่างในส่วนผสมอยู่สามแบบโดยจะมีค่าอัตราส่วนระหว่างอัตราส่วนผสมของซีเมนต์น้ำและดิน (C_w/A_w) สามค่า

ดังแสดงในรูปที่ 1 กราฟผลทดสอบโดย triaxial test ของดินซีเมนต์ดิน แสดงความสัมพันธ์ระหว่าง mean effective stress และ deviatoric stress และรูปที่ 2 กราฟผลทดสอบโดย triaxial test ของดินซีเมนต์ดิน แสดงความสัมพันธ์ระหว่าง vertical strain และ deviatoric stress จากผลการทดสอบจะเห็นว่าสตีเฟนของดินเหนียวผสมซีเมนต์จะเพิ่มขึ้นกับเมื่อค่า C_w/A_w มีค่าที่ลดลง และจะขึ้นอยู่กับค่าความเค้นเริ่มต้น ดังนั้นตัวแปรที่มีผลต่อความสัมพันธ์ของความเค้นและความเครียดของดินเหนียวผสมซีเมนต์แสดงดังตารางที่ 1

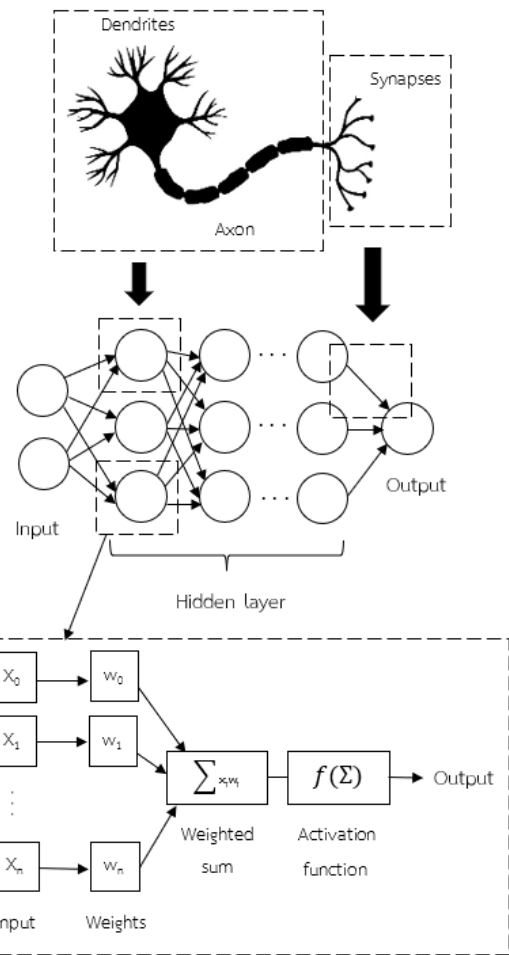
ตารางที่ 1 ตัวแปรที่มีผลต่อความสัมพันธ์ของความเค้นและความเครียดของดินเหนียวผสมซีเมนต์

C_w/A_w	Confining pressure (kPa)	Total unit weight (kN/m ³)	Total Water content (%)
2.92	50	13.214	112.600
	100	13.753	114.160
	200	14.912	109.230
6.92	50	14.296	112.130
	100	13.850	112.130
	200	14.219	114.130
10.92	50	13.069	116.393
	100	14.244	114.321

3. LSTM Recurrent Neural Network

3.1 Neural Network

Neural Networks คือ โครงข่ายที่เป็นการจำลองสมอง ด้วยโมเดลทางคณิตศาสตร์หรือโมเดลทางคอมพิวเตอร์ สำหรับประมวลผลสารสนเทศด้วยการคำนวณแบบคอนเนคชันนิสต์ (connectionist) โดยปกติสมองของมนุษย์จะมีหน่วยประมวลผลขนาดเล็กอยู่จำนวนมาก หรือเรียกว่าเซลล์ประสาท (neurons) และเชื่อมโยงกันด้วยโครงข่ายประสาท หรือจุดประสานประสาท (synapses) รวมกันเป็นโครงข่ายประสาทช่วยให้มนุษย์สามารถเรียนรู้และคิดวิเคราะห์ได้ จากรูปที่ 3 เปรียบเทียบการทำงานของสมองและ Neural Network โดยในแต่ละ neuron จะเปรียบเทียบกับเซลล์ประสาท และ เส้นเชื่อม หรือ Connected จะเปรียบเทียบกับ จุดประสานประสาท และ Hidden layer แต่ละชั้นจะเปรียบเทียบกับ การส่งผ่านเซลล์ประสาทหนึ่งไปยังอีกเซลล์ประสาทหนึ่งโมเดลทางคอมพิวเตอร์สามารถแบ่งการจำลองสมองมนุษย์ โดยสามารถแบ่งออกเป็น 2 ประเภทใหญ่ๆ ได้แก่ 1. โครงข่ายประสาทแบบตื้น (Shallow neural network) : มี Hidden layer แค่ 1 layer 2. โครงข่ายประสาทแบบลึก (Deep neural network) : มี Hidden layer มากกว่า 1 layer



รูปที่ 3 เปรียบเทียบการทำงานของสมองและ Neural Network

3.2 Deep learning

Deep learning หรือการเรียนรู้เชิงลึก [20] เป็นการเพิ่มความซับซ้อนโครงข่ายประสาทเทียมและใช้การคำนวณแบบขนานเพื่อเพิ่มประสิทธิภาพของการทำนายให้มีความแม่นยำขึ้น

3.2.1 Recurrent Neural Network (RNNs)

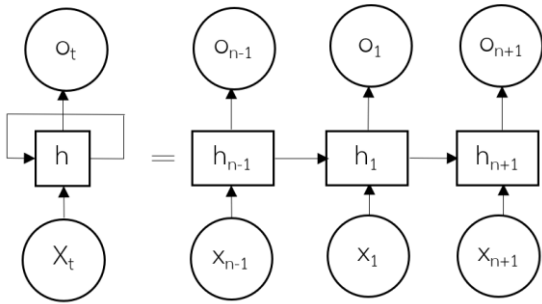
Recurrent Neural Network (RNNs) หรือโครงข่ายประสาทเทียมแบบเกิดซ้ำ เป็นโครงข่ายประสาทเทียมที่ใช้ข้อมูลในอดีตและปัจจุบันในการนำเข้าสู่โมเดล เพื่อหาผลลัพธ์ที่จะเกิดขึ้นในอนาคต กล่าวคือในการทำนายจะใช้ข้อมูลจาก Hidden state ก่อนหน้าหรือในลำดับ t-1 และข้อมูลในลำดับ t เพื่อทำนายผลลัพธ์ ณ ระดับ t ดังสมการที่ 1

$$h_t = f_n(w_t \cdot x_t + w_r \cdot h_{t-1}) \tag{1}$$

โดยที่

- h_t = ข้อมูลจาก Hidden state ณ ลำดับ t
- f_n = Activation function ของ hidden layer
- w_t = Weight matrix ของ hidden layer ณ ลำดับ t
- x_t = Input data ณ ลำดับ t

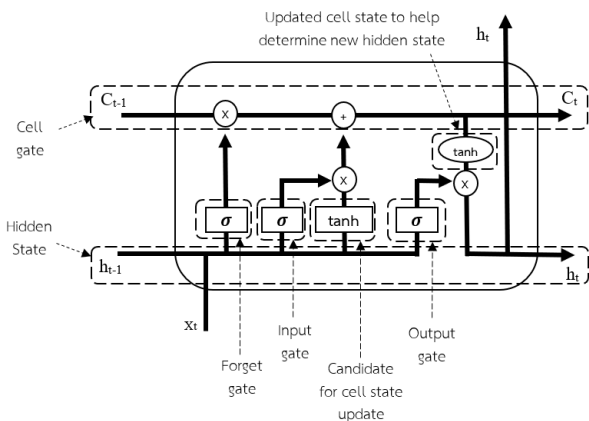
w_r = Weight matrix ของ hidden layer ณ ลำดับ t-1
 h_{t-1} = Input data ณ ลำดับ t-1 จาก hidden state



รูปที่ 4 ขั้นตอนการทำงานของโครงข่ายประสาทเทียมแบบเกิดซ้ำ

3.2.2 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) [21] หรือโครงข่ายประสาทเทียมแบบวนกลับชนิดพิเศษ เป็นโครงข่ายประสาทเทียมอีกรูปแบบหนึ่ง ซึ่งถูกพัฒนามาจาก RNNs โดยจะสามารถใช้กับข้อมูลที่มีจำนวนชั้นหรือลำดับที่ต่อเนื่องกันมากได้ นอกจากนี้ LSTM ยังสามารถเลือกที่จะจดจำข้อมูลหรือการทำข้อมูลในอดีตหรือ Hidden state ก่อนหน้าได้ โดยกระบวนการทำงานของ LSTM มี 5 ส่วน ได้แก่ การลืม การเขียน การอัปเดต การอ่านค่าและการส่งออก ตามลำดับ [22]



รูปที่ 5 การทำงานของ Long Short-Term Memory (LSTM) [23]

การลืมหรือ Forget Gate คือ ส่วนที่กำหนดว่าข้อมูลใดบน cell state ควรส่งต่อและข้อมูลใดตัดออกไป ทำงานโดยการรับข้อมูลจาก Node ก่อนหน้า การเขียนหรือ Input Gate คือ ส่วนที่ทำหน้าที่รับข้อมูลเพื่อบันทึกใน Node หรือเรียกว่าการ “Write” ข้อมูล Input Gate จะเป็นตัวตัดสินใจว่าจะอัปเดตข้อมูลที่เข้ามาใหม่หรือไม่ การอัปเดตหรือ Update Cell State คือ การอัปเดตของข้อมูลที่ผ่าน Input Gate และ Forget Gate จะมี Node หนึ่งที่ถูกส่งต่อทันทีและอีก Node ถูกผลักเข้าสู่การแปลงค่าโดยฟังก์ชัน Tan Hyperbolic และนำค่ามารวมกันที่จะซบปลาย หลังจาก

Update Cell State ข้อมูลที่ได้จะถูกนำไปคูณกับข้อมูลจาก Input Gate และนำไปรวมกับข้อมูลจาก Forget Gate ค่าที่ได้ออกมาจะเป็นค่าสถานะใหม่ของ Node นั้น ที่ผ่านการอัปเดตแล้ว การอ่านค่า หรือ Read คือการอนุญาตให้ข้อมูลผ่านเข้ามาใน Output โดยใช้ฟังก์ชัน Sigmoid เป็นตัวกรอง การส่งออกหรือ Output Gate เมื่อข้อมูลเข้ามาในขั้นนี้แล้วข้อมูลส่วนหนึ่งจะผ่านการอัปเดตสถานะแล้วจาก Update Cell State จะถูกส่งต่อทันทีโดยไม่ผ่าน Output Gate และข้อมูลอีกส่วนหนึ่งถูกส่งต่อไปยัง Output Gate เพื่อแปลงค่าโดยฟังก์ชัน Tan Hyperbolic ก่อนจะถูกส่งต่อไปยัง Node ถัดไป และยังคงส่งไปยังผลลัพธ์ของ Node เหล่านั้นด้วยการใช้แบบจำลองแบบ LSTM มีการใช้อย่างแพร่หลายและทั้งด้านวิศวกรรมโยธา ด้านการเงิน การระบาดของโรค [24-26]

3.3 การประเมินประสิทธิภาพของแบบจำลอง (Performance Measurement)

Mean Absolute Percentage Error (MAPE) คือ การหาค่าเฉลี่ยของร้อยละความแตกต่างสัมบูรณ์ระหว่างค่าพยากรณ์และค่าจริงมีหน่วยเปอร์เซ็นต์ หากผลลัพธ์ที่ได้มีค่าน้อยแสดงว่าแบบจำลองนั้นสามารถทำนายค่าได้ใกล้เคียงกับค่าจริง แสดงดังสมการที่ 2

$$MAPE = \left(\frac{1}{n} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \right) \times 100 \quad (2)$$

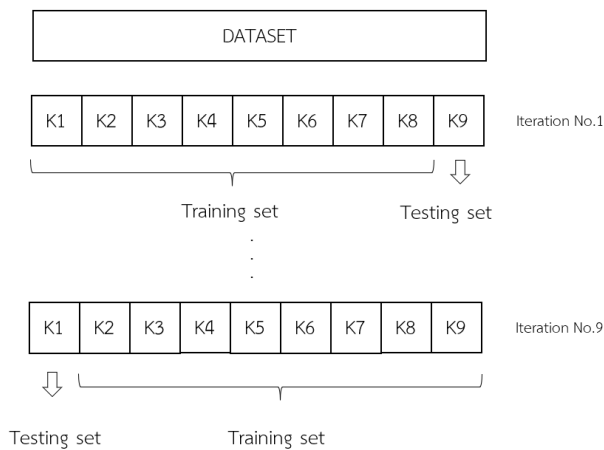
โดยที่

- y_i = ค่าจริงจากการทดสอบ
- \hat{y}_i = ค่าจากการทำนาย
- n, N = จำนวนข้อมูลทั้งหมด

3.4 การทดสอบความถูกต้องของโครงสร้าง LSTM

หาค่าความสามารถของโครงข่ายประสาทเทียมแบบเกิดซ้ำชนิด LSTM โดยการ Cross-Validation ซึ่งคือ วิธีการเชิงสถิติ เป็นการประเมินความสามารถของโมเดลโดยการใช้ Dataset กับโมเดลที่ถูกสร้างขึ้น โดยข้อมูลจะถูกแบ่งเป็น Training set และ Testing set และวัดค่าความถูกต้องด้วยดัชนีชี้วัดค่าต่างๆ เช่น MAPE MAE เป็นต้น

K-Fold Cross-Validation [27] คือ การแบ่ง dataset ออกเป็น K ส่วน โดยจะใช้ K-x ส่วน ในการ Training set และ x ส่วน ในการ Testing set เช่น K = 10 Dataset จะถูกแบ่งออกเป็น 10 ส่วนเท่าๆกัน ได้แก่ K1 - K10 ดังรูปที่ 6 ในการฝึกฝนโมเดลครั้งที่ 1 (Iteration No.1) Dataset ชุด K1-K9 จะนำมาเป็น Training set ที่ใช้ในการฝึกฝนโมเดล และ Dataset ชุด K10 จะนำมาใช้เป็น Testing set ในการฝึกฝนโมเดลในรอบถัดไป จะใช้ข้อมูล 9 ส่วนในการใช้เป็น Training set และ 1 ส่วน ในการเป็น Testing set กระบวนการนี้ทำซ้ำจนข้อมูลทุกส่วนจะถูกใช้เป็น Testing set



รูปที่ 6 แสดงขั้นตอนการทำงานของ K-Fold Cross-Validation

4. ขั้นตอนการพัฒนาแบบจำลอง

การสร้าง Machine Learning หรือ การสร้างโมเดลการเรียนรู้มีขั้นตอนหลัก 3 ลำดับ ดังนี้

4.1 Preparing data

Preparing data and cleaning data คือ การจัดเตรียมข้อมูลและทำความสะอาดข้อมูลก่อนใช้ในการนำไปเลือกตัวแปรและสร้างโมเดล กล่าวคือ การลบข้อมูลที่เป็น NaN หรือข้อมูลที่ขาดหายไป และการจัดการชุดข้อมูลให้เหมาะสมกับการสร้างโมเดล การ Preparing data มีด้วยกันหลายวิธี เช่น Scaling data, Feature selection

4.1.1 Scaling data

Scaling data คือ การปรับข้อมูลให้อยู่ในช่วงค่าที่เหมาะสม โดยการปรับค่าความแปรปรวนและค่าเฉลี่ยของข้อมูล ให้มีค่าเท่ากันก่อนในการวิจัยนี้ใช้ MinMaxScaler เพื่อปรับข้อมูล โดยปรับให้ข้อมูลมีค่าระหว่างค่าต่ำสุดและสูงสุด โดยค่าที่ได้จะอยู่ในช่วง -1 ถึง 1 ดังสมการที่ 3

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (3)$$

โดยที่

- x' = ค่าที่ได้จากการปรับข้อมูล
- x = Input data
- x_{min} = Input data ค่าน้อยที่สุด
- x_{max} = Input data ค่ามากที่สุด

4.1.2 Feature selection

Feature selection เป็นกระบวนการลดขนาดของข้อมูลที่ใช้ในการสร้างโมเดล โดยการเลือกตัวแปรที่มีคุณลักษณะและเพิ่มประสิทธิภาพในการประมวลผล ในบทความนี้ เลือกใช้วิธี Recursive Feature Elimination คือ การเพิ่มหรือลดตัวแปรในงานทดสอบ และทดสอบโมเดล

จนค่าความแม่นยำไม่เพิ่มหรือลดแล้ว จึงเลือกตัวแปรที่เหมาะสมในการสร้างโมเดล feature ที่ใช้ในแบบจำลอง ได้แก่ mean effective stress, deviatoric stress, vertical strain, pore water pressure, C_w/A_w , Confining pressure, ค่าน้ำหนักรวมต่อหน่วยปริมาตรของมวลดิน (Total unit weight), and Total water content.

4.2 Modeling

Modeling ขั้นตอนหลัก คือ การเลือกโมเดล และการปรับ Parameter ต่างๆ โดยโมเดลแต่ละชนิดจะเหมาะสมกับข้อมูลที่ต่างชนิดกัน เช่น ข้อมูลที่เป็นภาพ จะเหมาะกับการใช้ Computer vision เพื่อทำ image recognitions หรือข้อมูลที่เป็นลำดับจะเหมาะกับการใช้ SimpleRNN, LSTM หรือ GRU เป็นต้น

4.3 Evaluation

Evaluation หรือการประเมินประสิทธิภาพของโมเดล โดยในบทความนี้จะประเมินโดยการใช้วิธี K-Fold Cross-Validation และใช้ Mean Absolute Percentage Error (MAPE) ในการวัดค่าความแม่นยำ

5. กรณีศึกษา

5.1 การตั้งค่าโครงสร้าง LSTM

งานวิจัยนี้จะสร้างแบบจำลอง ANN โดยใช้ Keras เป็นเครื่องมือในการสร้างแบบจำลองโดยจะพัฒนาเขียนขึ้นโดยภาษา Python สาเหตุการเลือกพัฒนาระบบบน Python เพราะเป็นภาษาที่จัดการในด้าน Array ที่สะดวกโดยการใช้ NumPy หลังจากพัฒนาแล้วเสร็จ สามารถนำไปโปรแกรมที่พัฒนาแล้วไปใช้ต่อด้านอื่นได้ง่าย เช่น การนำไปใช้เชื่อมต่อกับโปรแกรม Finite Element เป็นต้น

Feature ที่ใช้ในสำหรับ Training มีทั้งหมด 11 ตัว แบ่งเป็นส่วน Input 8 ตัว ได้แก่

- mean effective stress ในลำดับที่ i (p^i)
- deviatoric stress ในลำดับที่ i (q^i)
- vertical strain ในลำดับที่ i (ϵ_s^i)
- pore water pressure (Δu)
- Total unit weight (γ)
- ปริมาณน้ำต่อนูนซีเมนต์ (C_w / A_w)
- Confining pressure (p_{int})
- Total water content (C_w)

Output 3 ตัว ได้แก่

- mean effective stress ในลำดับที่ i +time step ($p^{i+timestep}$)
- deviatoric stress ในลำดับที่ i + time step ($q^{i+timestep}$)
- vertical strain ในลำดับที่ i + time step ($\epsilon_s^{i+timestep}$)

โดยในการจัดการ array สำหรับการ training 1 ครั้ง สำหรับ 2 time steps แบ่งเป็น ค่า Input feature, array

$$X = [(p^{i-1}, q^{i-1}, \varepsilon_s^{i-1}, \Delta u, \gamma, C_w/A_w, p_{int}, C_w), (p^i, q^i, \varepsilon_s^i, \Delta u, \gamma, C_w/A_w, p_{int}, C_w)]$$

หมายถึง การจัดชุดข้อมูลสำหรับ Input ทั้งหมด 2 ชุด โดยกำหนดให้ i คือ step แรก ซึ่งคือ step ปัจจุบัน และ $i-1$ เป็น step ที่สองสำหรับย้อนกลับไป

Output feature, array

$$Y = (p^{i+timstep}, q^{i+timstep}, \varepsilon_s^{i+timstep})$$

ในที่นี้จะเป็นการทำนาย โดยให้ผลได้แก่ mean effective stress, deviatoric stress, vertical strain ไปข้างหน้า 2 time steps และในการทดสอบจะใส่เพียงค่า array X

การพัฒนาโครงสร้างเพื่อให้ได้โครงสร้างที่เหมาะสมที่สุดในการศึกษานี้ จะใช้การ Trial and error ในการหาโครงสร้างที่เหมาะสมที่สุดของ ANN ซึ่งงานวิจัยนี้จะใช้ Hidden layer ที่ 4 ชั้น รายละเอียดแสดงในตารางที่ 2 เริ่มจากจาก Input ข้อมูลเข้าสู่โครงสร้าง โดยใน Hidden layer ชั้นแรก และชั้นที่ 2 จะเป็น LSTM ทำหน้าที่เรียนรู้ข้อมูล โดย activation function ที่ใช้สำหรับสองชั้นนี้คือ ReLu และสองชั้นสุดท้ายจะเป็น Dense หรือ Fully connected layer ทำหน้าที่รวม Output ให้มีขนาดเท่ากับ จำนวน Node ที่เราต้องการในโครงสร้างนี้คือ 10 และ 5 ตามลำดับ และในชั้นตอน Output activation function คือ tanh โดยมีจำนวนผลที่ต้องการคือ 3 เท่ากับจำนวน Node

ตารางที่ 2 โครงสร้างที่ใช้ในการทำโมเดล LSTM

Parameters	
Hidden layer	4
Node	(50,20,10,5)
Flatten	1
Activation function	ReLu, tanh
Optimizer	Adam
loss	mean squared error
Matrics	accuracy

5.2 ผลการศึกษา

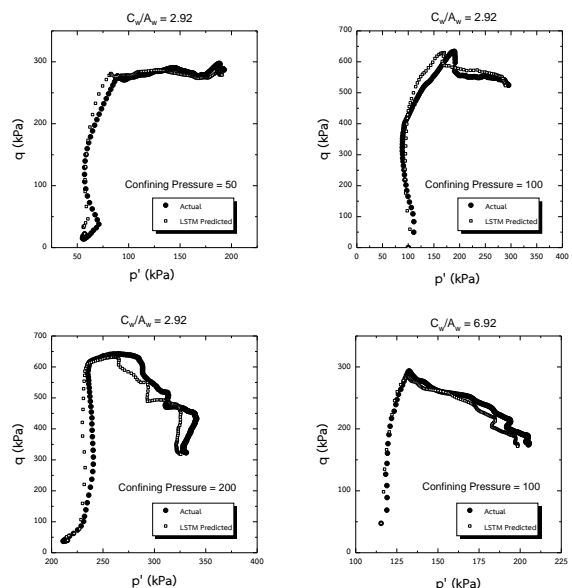
หลังจากสร้างแบบจำลองแล้วเสร็จจะต้องมีการทดสอบว่าโครงสร้างของแบบจำลองมีความเหมาะสมหรือไม่ โดยดูได้จากค่า MAPE จากการทำนาย จากการทดสอบโมเดลเพื่อหาค่าความแม่นยำของโมเดลด้วย K-Fold Cross-Validation เปรียบเทียบผลการทดสอบได้ดังตารางที่ 2 ในงานวิจัยนี้มีการเปรียบเทียบกับแบบจำลองอื่นๆ ของ RNNs ด้วยคือ Gated recurrent unit (GRU) [28-29] และ การใช้ SimpleRNN [30-31] จากผลการทดสอบโดยใช้ K-Fold Cross-Validation พบว่าได้ผลออกมาเป็นที่น่าพอใจคือได้ความคลาดเคลื่อนที่ต่ำมาก 5-9 % ขึ้นอยู่กับชนิดและ time step ของแบบจำลอง จะพบว่าแบบจำลอง LSTM ที่ time step เท่ากับ 2 จะให้ค่าความคลาดเคลื่อนที่ต่ำที่สุดคือ 4 % แบบจำลอง LSTM จะให้ผลที่ดีที่สุดเมื่อเทียบกับ GRU และ SimpleRNN ซึ่งเป็นไปตามความคาดหมายของผู้วิจัย แต่ข้อเสียของ LSTM คือใช้เวลาในการฝึกหัดแบบจำลองที่นานเนื่องจากสมการมีความซับซ้อน ต้องใช้คอมพิวเตอร์ที่

ค่อนข้างมีความสามารถสูง ในงานวิจัยนี้ใช้ Cloud ของ google ในการฝึกแบบจำลองใช้เวลา 1 นาที และการทดสอบโดยใช้ K-Fold Cross-Validation ใช้เวลา 4 นาที เมื่อเปรียบเทียบกับค่า MAPE กับ Zhang, N. ที่มีการพัฒนาโมเดลโดยใช้ข้อมูลจากทดสอบดินเหนียว [14] จะเห็นได้ว่าโมเดลของงานวิจัยนี้สามารถทำงานได้มีค่าความผิดพลาดที่น้อยกว่า โดยโมเดลนี้สามารถนำไปใช้ในการทำนายผลของ mean effective stress, deviatoric stress and vertical strain ของดินซีเมนต์ได้

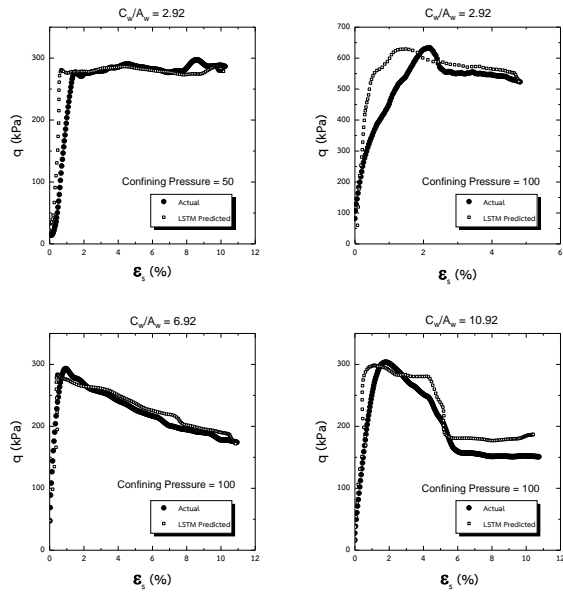
ตารางที่ 2 เปรียบเทียบผลการทดสอบ LSTM ANN และ GRU โดยใช้ K-Fold Cross-Validation

โครงสร้าง	1 time step	2 time steps	3 time steps
LSTM	5.10% (+/- 1.30%)	4.00% (+/- 1.16%)	4.53% (+/- 1.55%)
GRU	6.03% (+/- 1.15%)	4.43% (+/- 1.09%)	4.47% (+/- 0.52%)
SimpleRNN	6.72% (+/- 1.24%)	6.18% (+/- 2.30%)	9.68% (+/- 4.38%)

โดยรูปที่ 7 และรูปที่ 8 แสดงผลเปรียบเทียบระหว่างค่าจริงที่ได้จากการทดสอบหรือค่าที่ใช้ในการ Training โมเดล เปรียบเทียบกับค่าที่ได้จากการทำนายผลโดยโมเดล LSTM (2 time steps) จะได้ว่าสามารถจำลองพฤติกรรมของดินเหนียวผสมซีเมนต์ได้เป็นอย่างดีโดยสามารถจำลองได้ในส่วนผสมและความเค้นเริ่มต้นที่ต่างกัน ซึ่งแสดงถึงประสิทธิภาพของการใช้ ANN ในการทำนายความสัมพันธ์ของความเค้นและความเครียดของวัสดุทางด้านวิศวกรรมเทคนิคธรณีได้เป็นอย่างดี น่าจะเป็นจุดเริ่มต้นในการพัฒนาการใช้ ANN ในการทดแทนการใช้ Constitutive model แบบเดิมร่วมกับโปรแกรม Finite Element



รูปที่ 7 ความสัมพันธ์ระหว่าง p' - q ของค่าจริงและค่าที่ได้จากการทำนายโดย LSTM



รูปที่ 8 ความสัมพันธ์ระหว่าง ϵ_s (%) - q ของค่าจริงและค่าที่ได้จากการทำนายโดย LSTM

6. สรุปผลการศึกษา

จากผลการศึกษาสามารถสรุปผลได้ดังต่อไปนี้

- งานวิจัยนี้สามารถพัฒนาแบบจำลอง Recurrent Neural Network เพื่อใช้ทำนายลักษณะการเสียรูปของดินเหนียวผสมดินซีเมนต์ แบบจำลองที่สร้างขึ้นสามารถทำนายความสัมพันธ์ของความเค้นและความเครียดของดินเหนียวผสมซีเมนต์ได้เป็นอย่างดีในทุกๆ ส่วนผสม

- การเลือกโมเดลที่เหมาะสมกับข้อมูลมีผลกับการทำงานของโมเดล เนื่องจากข้อมูลมีลักษณะเป็นลำดับและมีข้อมูลจำนวนมาก การเลือกใช้โมเดล LSTM จึงมีความเหมาะสม นอกจากนี้ time step ที่เลือกใช้ในการ Training โมเดล ส่งผลต่อค่าความแม่นยำของโมเดล โดย time step เท่ากับ 2 จะมีความแม่นยำของโมเดลมากที่สุดเมื่อเทียบกับ time step เท่ากับ 1 และ 3

- จากการทดสอบแบบจำลองพบว่า แบบจำลองที่มีค่า MAPE น้อยที่สุดหรือใกล้เคียงค่าความจริงมากที่สุดคือโมเดล LSTM ที่มีการใช้ 2 time steps ในการทำนายโดยค่า MAPE เท่ากับ 4.00%

เอกสารอ้างอิง

[1] Uddin, K., (1994). *Strength and Deformation Behavior of Cement Treated Bangkok Clay*, Doctoral of Engineering thesis, Civil Engineering, Asian Institute of Technology, Bangkok, Thailand, pp. 63-65.
[2] Uddin, K., Balasubramaniam, A.S. and Bergado, D.T., (1997). "Engineering Behavior of Cement-Treated Bangkok Soft Clay", *Geotechnique*, Vol. 28, No. 1, pp. 89-119.

[3] Bergado, D.T., Lin, A.D.G. and Nakamura, M., (1999). "Evaluation of Silty Sand as a Material for Sand Compaction Piles and Applications", *Ground Improvement*, 3(2), pp. 7-19.
[4] Lorenzo, G.A. and Bergado, D.T., (2004). "Fundamental Parameters of Cement-Admixed Clay", *Geotechnical and Geoenvironmental Engineering*, 130(10), pp. 1042-1050.
[5] Horpibulsuk, S., Miura, N. and Bergado, D.T., (2004a). "Undrained Shear Behavior of Cement Admixed Clay at High Water Content", *Journal of the Geotechnical Engineering Division*, 130(10), pp. 1096-1105.
[6] Horpibulsuk, S., Miura, N. and Bergado, D.T., (2004b). "Undrained Shear Behavior of Cement Admixed Clay at High Water Content", *Journal of the Geotechnical Engineering Division*, 130(10), pp. 1096-1105.
[7] Horpibulsuk, S., Rachan, R., Suddeepong, A., and Chinkulkijniwat, A., (2011). "Strength Development in Cement Admixed Bangkok Clay: Laboratory and Field Investigations", *Japanese Geotechnical Society*, 51(2), pp. 239-251.
[8] Arroyo, M., Ciantia, M., Castellanza, R., Gens, A., & Nova, R. (2012). Simulation of cement-improved clay structures with a bonded elasto-plastic model, *A practical approach. Computers and Geotechnics*, 45, 140-150.
[9] Chattonjai, P. (2016). Hypoplastic model for simulation of compressibility characteristics of cement-admixed Bangkok soft clay at high water content. *AIP Conference Proceedings*, 1738.
[10] Youwai, S., Chattanjai, P., Jongpradist, P., & Kongkitkul, W. (2010). Hypoplastic Model for Simulation of Deformation Characteristics of Bangkok Soft Clay with Different Stress Paths. May, 160-165
[11] Ellis, G. W., Yao, C., Zhao, R., & Penumadu, D. (1995). Stress-strain modeling of sands using artificial neural networks. *Journal of Geotechnical Engineering*, 121(5), 429-435.
[12] Penumadu, D., & Zhao, R. (1999). Triaxial compression behavior of sand and gravel using artificial neural networks (ANN). *Computers and Geotechnics*, 24(3), 207-230.
[13] Zhang, P., Yin, Z. Y., Jin, Y. F., & Ye, G. L. (2020). An AI-based model for describing cyclic characteristics of granular materials. *In International Journal for Numerical and Analytical Methods in Geomechanics*, 44(9), pp. 1315-1335.

- [14] Zhang, N., Shen, S. L., Zhou, A., & Jin, Y. F. (2021). Application of LSTM approach for modelling stress-strain behaviour of soil. *Applied Soft Computing*, 100.
- [15] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., ... Zheng, X. (2016). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*.
- [16] Chollet, F. (2015) Keras Retrieved from <https://github.com/keras-team/keras>
- [17] Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2017-January*, 1800–1807.
- [18] Manaswi, N. K. (2018). Deep Learning with Applications Using Python. *Deep Learning with Applications Using Python*, 31–43.
- [19] Rodpol, P. (2019). *Deformation Characteristics Of Cement Treated Clay Under Compression Unloading Stress Condition*. M.Eng. Dissertation, King Mongkut's University of Technology Thonburi, Thailand.
- [20] Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, 31(3), 685–695.
- [21] Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735–1780.
- [22] Christopher Olah (27 Aug 2015) Understanding LSTM Networks. Retrieved from <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>
- [23] Saul Dobilas (7 Feb 2022) LSTM Recurrent Neural Networks — How to Teach a Network to Remember the Past. Retrieved from <https://towardsdatascience.com/lstm-recurrent-neural-networks-how-to-teach-a-network-to-remember-the-past-55e54c2ff22e>
- [24] Kim, H. Il, & Kim, B. H. (2020). Flood Hazard Rating Prediction for Urban Areas Using Random Forest and LSTM. *KSCE Journal of Civil Engineering*, 24(12), 3884–3896.
- [25] Murtaza, R., Harshal, P., & Shraddha, V. (2015). Predicting Stock Prices Using LSTM. *International Journal of Science and Research*, 6(4), 2319–7064.
- [26] Shahid, F., Zameer, A., & Muneeb, M. (2020). Predictions for COVID-19 with deep learning models of LSTM, GRU and Bi-LSTM. *Chaos, Solitons and Fractals*, 140, 110212.
- [27] Scikit-learn developers (2007-2022) Retrieved from https://scikit-learn.org/stable/modules/cross_validation.html
- [28] Cho, K., van Merriënboer, B., Bahdanau, D., & Bengio, Y. (2014). On the properties of neural machine translation: Encoder–decoder approaches. *Proceedings of SSST 2014 - 8th Workshop on Syntax, Semantics and Structure in Statistical Translation*, 103–111.
- [29] Simeon Kostadinov (16 Dec 2017) Understanding GRU Networks Retrieved from <https://towardsdatascience.com/understanding-gru-networks-2ef37df6c9be>
- [30] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning Internal Representations by Error Propagation. *Readings in Cognitive Science: A Perspective from Psychology and Artificial Intelligence*, V, 399–421.
- [31] Debasish Kalita (11 Mar 2022) A Brief Overview of Recurrent Neural Networks (RNN) Retrieved from <https://www.analyticsvidhya.com/blog/2022/03/a-brief-overview-of-recurrent-neural-networks-rnn/>