บทความวิจัย

กระบวนการสำหรับการจำแนกความรู้สึกที่มีข้อมูลไม่สมดุล

A Method of Imbalanced Sentiment Classification

พีระวัฒน์ บุญบ้านงิ้ว[1] และจันทิมา พลพินิจ[2]

Pheerawat Bunbanngio[1] and Jantima Polpinij[2]

บทคัดย่อ

การจำแนกความรู้สึก (Sentiment Classification) คือการจำแนกเอกสารตามขั้นความรู้สึกซึ่งโดยทั่วไปอาจจะจำแนกเป็นความรู้สึกที่เป็นบวก (Positive) ความรู้สึกที่เป็นลบ (Negative) และความรู้สึกที่เป็นกลาง (Neutral) โดยการจำแนกความรู้สึกนั้น ได้รับการศึกษามาอย่างต่อเนื่อง เพราะการประยุกต์ใช้ในหลายลักษณะ แต่โดยทั่วไปมักจะนิยมใช้ในการจำแนกความรู้สึกที่มีการแสดงไว้ในรูปแบบข้อความ (Text) เช่น ประยุกต์ใช้ในการจัดอันดับความรู้สึกจากข้อความแสดงความคิดเห็นของผู้คนที่มีต่อสินค้าและบริการ การประยุกต์ใช้เพื่อวิเคราะห์ความรู้สึกของผู้เรียน การประยุกต์ใช้เพื่อวิเคราะห์ความรู้สึกของผู้คนในเรื่องการเมือง เป็นต้น ซึ่งปัญหาความไม่สมดุลของข้อมูลในคลาสนั้น เกิดจากกลุ่มตัวอย่างที่ใช้ในการเรียนรู้มีข้อมูลไม่สมดุลกัน โดยกลุ่มที่มีข้อมูลมากกว่าจะเรียกว่า “ข้อมูลกลุ่มหลัก (Majority Class)” ขณะที่กลุ่มตัวอย่างที่มีข้อมูลจำนวนน้อยกว่าจะเรียกว่า “ข้อมูลกลุ่มรอง (Minority Class)” เมื่อนำเอาชุดข้อมูลในลักษณะนี้ไปเรียนรู้เพื่อสร้างตัวจำแนกความรู้สึก (Sentiment Classifier) ข้อมูลใหม่ๆ ที่อ่านเข้ามาเพื่อวิเคราะห์เพื่อจำแนกกลุ่มด้วยตัวจำแนกความรู้สึกดังกล่าวก็มีแนวโน้มที่จะทำนายกลุ่มของข้อมูลนั้นไปยังทิศทางของข้อมูลกลุ่มหลักที่ใช้ในการเรียนรู้ตัวจำแนกความรู้สึกดังนั้น ในโครงงานปริญญานิพนธ์ฉบับนี้ จึงได้นำเสนอการศึกษาการแก้ปัญหาความไม่สมดุลของข้อมูลในการจำแนกความรู้สึกด้วยเทคนิคการให้น้ำหนักคำ 5 เทคนิค คือ TF-IDF, Delta TF-IDF, TF-IDF-ICF, TF-RF และ TF-IGM ร่วมกับแมชชีนเลิร์นนิง 3 ตัว คือ Naïve Bayes, K-Nearest Neighbor และสุดท้าย Convolution Neural Network

คำสำคัญ: การจำแนกเอกสาร, การให้น้ำหนักคำ, ข้อมูลไม่สมดุล, ซัพพอร์ตเวคเตอร์แมชชีน

บทนำ

การจำแนกความรู้สึก (Sentiment Classification) [1] คือการจำแนกเอกสารตามขั้นความรู้สึกซึ่งโดยทั่วไปอาจจะจำแนกเป็นความรู้สึกที่เป็นบวก (Positive) ความรู้สึกที่เป็นลบ (Negative) และความรู้สึกที่เป็นกลาง (Neutral) โดยการจำแนกความรู้สึกนั้น ได้รับการศึกษามาอย่างต่อเนื่อง เพราะการประยุกต์ใช้ในหลายลักษณะ แต่โดยทั่วไปมักจะนิยมใช้ในการจำแนกความรู้สึกที่มีการแสดงไว้ในรูปแบบข้อความ (Text) [1] เช่น ประยุกต์ใช้ในการจัดอันดับความรู้สึกจากข้อความแสดงความคิดเห็นของผู้คนที่มีต่อสินค้าและบริการ การประยุกต์ใช้เพื่อวิเคราะห์ความรู้สึกของผู้เรียน การประยุกต์ใช้เพื่อวิเคราะห์ความรู้สึกของผู้คนในเรื่องการเมือง เป็นต้น

อย่างไรก็ตาม แม้ว่าการจำแนกความรู้สึกจะได้รับการศึกษาและความสนใจมาอย่างต่อเนื่อง แต่ยังมีปัญหาที่พบในการจำแนกความรู้สึกหลายประเด็น ประเด็นที่น่าสนใจและยังคงได้รับการศึกษาเพื่อการแก้ปัญหาอยู่คือ ปัญหาความไม่สมดุลของข้อมูลในการจำแนกความรู้สึก (Imbalanced Sentiment Classification) โดยทั่วไปที่พบมากคือปัญหาความไม่สมดุลของข้อมูลในคลาส (Class Imbalance Data) [2–5]

ซึ่งปัญหาความไม่สมดุลของข้อมูลในคลาสนั้น เกิดจากกลุ่มตัวอย่างที่ใช้ในการเรียนรู้มีข้อมูลไม่สมดุลกัน โดยกลุ่มที่มีข้อมูลมากกว่าจะเรียกว่า “ข้อมูลกลุ่มหลัก (Majority Class)” ขณะที่กลุ่มตัวอย่างที่มีข้อมูลจำนวนน้อยกว่าจะเรียกว่า “ข้อมูลกลุ่มรอง (Minority Class)” เมื่อนำเอาชุดข้อมูลในลักษณะนี้ไปเรียนรู้เพื่อสร้างตัวจำแนกความรู้สึก (Sentiment Classifier) ข้อมูลใหม่ๆ ที่อ่านเข้ามาเพื่อวิเคราะห์เพื่อจำแนกกลุ่มด้วยตัวจำแนกความรู้สึกดังกล่าว ก็มีแนวโน้มที่จะทำนายกลุ่มของข้อมูลนั้นไปยังทิศทางของข้อมูลกลุ่มหลักที่ใช้ในการเรียนรู้ตัวจำแนกความรู้สึก

เทคนิคหลายๆ เทคนิคได้ถูกนำเสนอเพื่อใช้ในการควบคุมปัญหาความไม่สมดุลของข้อมูลในการจำแนกความรู้สึก เช่น Resampling Methods [4] สำหรับวิธีการนี้จะเป็นการประยุกต์เอาวิธีสุ่มตัวอย่างซึ่งเป็นวิธีการทางสถิติ เพื่อสร้างข้อมูลสำหรับการสอน โดยมีจุดประสงค์เพื่อให้จำนวนสมาชิกในข้อมูลทั้งสองกลุ่มมีความสมดุลกัน ซึ่งประกอบด้วย 2 วิธีการใหญ่ๆ คือ Oversampling [6] และ Undersampling [6] โดยวิธีการทำแบบ Oversampling จะทำการสุ่มข้อมูลในกลุ่มรองเพื่อสร้างข้อมูลใหม่ของกลุ่มรองให้มีจำนวนเพิ่มมากขึ้นให้ใกล้เคียงหรือเท่ากับจำนวนข้อมูลในกลุ่มหลัก และในทางตรงข้ามวิธีการ Undersampling จะทำการสุ่มเลือกข้อมูลสำหรับการสอนจากข้อมูลในกลุ่มหลัก ให้ได้จำนวนที่ใกล้เคียงกับจำนวนข้อมูลในกลุ่มรอง โดยทั่วไปมักประยุกต์วิธีการแบบ Undersampling แต่ก็จะเกิดปัญหาข้อมูลไม่เพียงพอต่อการเรียนรู้

อย่างไรก็ตาม เมื่อไม่นานมานี้ หลายงานวิจัยที่นำเสนอเทคนิคการให้น้ำหนักคำ (Term Weighting) เข้ามาช่วยในการแก้ปัญหาความไม่สมดุลของข้อมูลในการจำแนกความรู้สึก [8], [9] และพบว่าเทคนิคการให้น้ำหนักคำแบบมีผู้สอน (Supervised Term Weighting: STW) มีแนวโน้มที่จะทำให้เกิดประสิทธิภาพในการจำแนกความรู้สึกที่ดีขึ้น

ดังนั้นในโครงงานปริญญานิพนธ์ฉบับนี้ จึงได้นำเสนอการศึกษาการแก้ปัญหาความไม่สมดุลของข้อมูลในการจำแนกความรู้สึกด้วยเทคนิคการให้น้ำหนักคำแบบมีผู้สอนอย่างน้อย 3 เทคนิค พร้อมทั้งทำการเปรียบเทียบการเทคนิคการให้น้ำหนักคำแบบไม่มีผู้สอน (Unsupervised Term Weighting) ที่นิยมใช้ในการจำแนกเอกสารความรู้สึกนั่นคือ *tf-idf* (Term Frequency-Inverse Document Fre-quency) (Salton, Wong, & Yang, 1975) ภายใต้ตัวจำแนกความรู้สึกอย่างน้อย 3 ตัว

ทบทวนวรรณกรรม

1. การประมวลผลภาษาธรรมชาติ (Natural Language Processing: NLP)

การประมวลผลภาษาธรรมชาติ [4, 5] คือ สาขาย่อยของปัญญาประดิษฐ์และภาษาศาสตร์ที่ศึกษาปัญหาในการประมวลผลและใช้งานภาษาธรรมชาติ รวมทั้งการทำความเข้าใจภาษาธรรมชาติ

ทั้งนี้เพื่อให้คอมพิวเตอร์สามารถเข้าใจภาษามนุษย์ได้ โดยแบ่งเป็นภาษาพูดและภาษาเขียน ซึ่งในที่นี้จะกล่าวถึงภาษาเขียนเท่านั้น

ระดับของการประมวลผลภาษาธรรมชาติ มีทั้งหมด 5 ระดับ คือ

1) Morphological Analysis เป็นการวิเคราะห์หน่วยคำว่าสามารถแยกย่อยได้เป็นอะไรบ้าง และคำๆ นั้นมีหน้าที่อะไร เช่น “friendly” แยกได้เป็น “friend” และ “ly” เป็นต้น

2) Syntactic Analysis เป็นการวิเคราะห์ทางไวยากรณ์ เพื่อให้รู้ว่าประโยคหนึ่งๆ มีโครงสร้างเชิงวากยสัมพันธ์อย่างไร

3) Semantic Analysis จะเป็นการวิเคราะห์ความหมายของประโยคนั้นๆ

4) Discourse Integration เป็นการพิจารณาความหมายของประโยค โดยพิจารณาจากประโยคข้างเคียง เนื่องจากบางคำจะเข้าใจความหมายได้ต้องดูความหมายของประโยคก่อนหน้า

5) Pragmatic Analysis คือการแปลความหมายของประโยค เพื่อดูความตั้งใจในการสื่อสารของผู้สื่อสารว่าจุดประสงค์กล่าวถึงอะไร

2. การวิเคราะห์ความรู้สึก (Sentiment Analysis)

การวิเคราะห์ความรู้สึก [6-8] คืองานวิจัยที่อยู่ในกลุ่มของการประมวลผลภาษาธรรมชาติ (Natural Language Processing: NLP) ที่มีกระบวนการมุ่งเน้นการวิเคราะห์และตรวจสอบความรู้สึก (Opinion) ของผู้คนจากข้อความ (Text) ที่คนเหล่านั้นเขียนหรือโพสต์เอาไว้ เพื่อบ่งบอกความรู้สึกของตนเองที่มีต่อบางสิ่งบางอย่างที่ตนเองสนใจ เช่นความรู้สึกดี (Positive หรือ Good) หรือความรู้สึกที่ไม่ดีหรือไม่ชอบ (Negative หรือ Bad) เช่น เมื่อลูกค้าซื้อคอมพิวเตอร์ไป 1 เครื่อง ลูกค้าอาจจะให้คะแนนเฉลี่ยเกี่ยวกับคอมพิวเตอร์รุ่นนั้นๆ ไว้ที่ 3 จากคะแนนเต็ม 5 แต่หัวข้อต่างๆ ที่สอบถามไปยังลูกค้าอาจยังไม่ครอบคลุมในทุกๆ กรณี ที่เป็นความต้องการหรือความคาดหวังต่อสินค้าและบริการของลูกค้า ทำให้ลูกค้าอาจจะไปเขียนแสดงความรู้สึกเกี่ยวกับคอมพิวเตอร์รุ่นนั้นๆ ไว้ใน Blog, Twitter หรือ Facebook ของตนเอง [10, 37]

การวิเคราะห์ความรู้สึกสามารถแบ่งได้ 3 ระดับ ดังนี้ [5]

1) การวิเคราะห์ความรู้สึกระดับเอกสาร (Document Level Analysis) เป็นการวิเคราะห์ข้อความแสดงความคิดเห็นในแบบหยาบ เนื่องจากเป็นการนำข้อความแสดงความคิดเห็นทั้งหมดจากเอกสารมาสรุป แยกขั้วความคิดเห็นเป็นขั้วบวก ขั้วลบ หรือเป็นกลาง

2) การวิเคราะห์ความรู้สึกระดับประโยค (Sentence Level Analysis) เป็นการวิเคราะห์ข้อความแสดงความคิดเห็น โดยแยกข้อความที่เป็นข้อความแสดงความคิดเห็น ออกมาจากข้อความที่เป็นข้อเท็จจริงในระดับที่เป็นประโยค แล้วนำมาแยกขั้วความคิดเห็นเป็นขั้วบวก ขั้วลบ หรือเป็นกลาง

3) การวิเคราะห์ความรู้สึกระดับคุณลักษณะ (Feature Level Analysis) เป็นการวิเคราะห์ข้อความแสดงความคิดเห็น โดยแยกคุณลักษณะที่สนใจหรือหัวข้อที่ถูกแสดงความคิดเห็นออกมาก่อน แล้วจึงนำมาแบ่งขั้วความคิดเห็นเป็นขั้วบวก ขั้วลบ หรือเป็นกลาง และนำมาจัดกลุ่มเข้ากับคำที่มีความหมายเหมือนกันในแต่ละคุณลักษณะ ซึ่งระบบจะวิเคราะห์ข้อความแสดงความคิดเห็นในระดับคุณลักษณะ แล้วนำผลลัพธ์ที่

ได้มาแสดงให้อยู่ในรูปแบบที่ผู้ใช้งานสามรถเข้าใจได้ง่ายขึ้น

3. การจำแนกหมวดหมู่เอกสาร (Text Classification)

การจำแนกหมวดหมู่เอกสาร [13, 25] เป็นการนำวิธีการเรียนรู้ด้วยคอมพิวเตอร์ (Machine Learning) ประยุกต์รวมกับการประมวลผลภาษาธรรมชาติ การจัดแบ่งกลุ่มเอกสารแบบอัตโนมัติเป็นการแบ่งกลุ่มตามเนื้อหาของเอกสารโดยที่มีการกำหนดกลุ่มหรือหมวดหมู่ของเอกสารไว้ก่อนหน้า เป็นลักษณะการวิเคราะห์เอกสารที่เข้ามากับเอกสารในแต่ละหมวดหมู่ เพื่อดูว่าเอกสารนั้นๆ ให้มีลักษณะคล้ายกับหมวดหมู่ใดมากที่สุด

โดยสามารถให้นิยามการจำแนกหมวดหมู่เอกสาร ดังนี้ กำหนดให้คู่ลำดับ $(d_j, c_i) \in D \times C$ โดยที่ D เป็นโดเมนของเอกสาร ขณะที่ C เป็นกลุ่มเอกสารที่เป็นไปได้ $\{ c_1, c_2, \ldots, c|c| \}$ และกำหนดให้ T เป็นคู่ลำดับ (d_j, c_i) ที่จะบ่งบอกว่าเอกสาร d_j อยู่ภายใต้กลุ่มหรือหมวดหมู่ c_i โดยให้ F เป็นฟังก์ชันที่กำหนดให้กับคู่ลำดับ (d_j, c_i) เพื่อบอกว่าเอกสาร d_j ควรอยู่ภายใต้กลุ่มหรือหมวดหมู่ c_i หรือไม่ ดังนั้นการประมาณค่าของฟังก์ชันเป้าหมายสามารถแสดงได้คือ $F: D \times C \longrightarrow \{T, F\}$ ซึ่งเป็นฟังก์ชันเป้าหมายที่จะแทนตัวจัดกลุ่มเอกสาร หรือ *Classifier*

4. การให้น้ำหนักคำ (Term Weighting)

การให้น้ำหนักคำ [17] ถือว่าเป็นส่วนหนึ่งของการจัดการเอกสาร โดยรูปแบบการให้น้ำหนักสามารถแบ่งออกเป็นสองประเภทหลักตามการใช้งานข้อมูลชั้นเรียนในเอกสารการฝึกอบรม ดังนี้

รูปแบบแรกคือ Unsupervised Term Weighting (UTW) [18] คือรูปแบบการให้น้ำหนักคำที่ซึ่งไม่ใช้ข้อมูลชั้นเรียนเพื่อสร้างน้ำหนัก รูปแบบที่ได้รับความนิยมมากที่สุดคือ TF-IDF (Term Frequency - Inverse Document, Frequency) ซึ่งถูกใช้อย่างมีประสิทธิภาพในการศึกษาการดึงข้อมูล แต่อย่างไรก็ตามมันไม่เหมาะสำหรับงานการจัดหมวดหมู่ข้อความ เนื่องจากการให้น้ำหนักคำแบบ Unsupervised Term Weighting เป็นการให้น้ำหนักคำกับเอกสารทั้งหมดโดยไม่แบ่งหมวดหมู่เอกสาร โดยหากใช้รูปแบบนี้จะทำให้ประสิทธิภาพในการจำแนกหมวดหมู่ข้อความลดลง

ส่วนรูปแบบที่สองเป็นรูปแบบที่นักวิจัยใช้ในผลงานนี้ คือ Supervised Term Weighting (STW) [11] ซึ่งได้รับการเสนอครั้งแรกโดย Debolc และ Sebastiani [11] โครงร่าง Supervised Term Weighting ใช้ชุดข้อมูลการฝึกอบรมของข้อมูลระดับชั้นเรียนเพื่อคำนวณน้ำหนักของคำศัพท์ โดยการให้น้ำหนักในแบบนี้จะใช้ประโยชน์จากข้อมูลระดับที่รู้จักในคลังข้อมูลการฝึกอบรม โดยจะทำให้การให้น้ำหนักมีประสิทธิภาพที่ดียิ่งขึ้น สำหรับการจำแนกหมวดหมู่ความรู้สึกของข้อความ การวิเคราะห์ความรู้สึก การจำแนกความไม่สมดุลของชุดเอกสาร และอื่นๆ

5. งานวิจัยที่เกี่ยวข้อง

ในการจำแนกความรู้สึกของเอกสารข้อความก็พบปัญหาของขจ้อมูลที่ไม่สมดุล ซึ่ง Li และคณะ [2] ได้ศึกษาเกี่ยวกับข้อมูลที่ไม่สมดุลหลายรูปแบบ เช่น จำนวนเอกสารที่ไม่สมดุล ขนาดของคลาสที่ไม่สมดุล รวมถึงความไม่สมดุลในคลาสย่อย จากการศึกษาที่ต่อเนื่องพบว่า ประเด็นที่หนึ่งจำนวนเอกสารข้อความในสองคลาสจะเท่ากัน ความแตกต่างของจำนวนคำในเอกสารกลายเป็นปัจจัยสำคัญที่มีผลต่อความถูกต้องของการจำแนกเอกสาร ประเด็นที่สอง เพื่อปรับปรุงความถูกต้องของการจำแนกเอกสารด้วยการเพิ่มจำนวนของกลุ่มข้อมูบที่มีจำนวนน้อย และประเด็นที่สาม ในกรณีของข้อมูลที่ไม่สมดุล คำเดียวกันที่ปรากฏในสองคลาสมักจะเป็นสารสนเทศสำคัญของคลาส นั่นคือ คลาสทับซ้อนกันจะไม่ส่งผลกระทบต่อความถูกต้องของการจัดประเภท

Flavio Carvalho และ Gustavo Pai Guedes ได้นำเสนอการให้น้ำหนักคำแบบ Supervised Term Weighting ที่เหมาะสมต่อการจำแนกความรู้สึกที่ไม่สมดุล โดยได้นำเสนอการให้น้ำหนักคำที่ได้รับการควบคุมดูแลเจ็ดชุดและแผนการกำหนดน้ำหนัก ซึ่งวิธีนี้เป็นวิธีที่มีประสิทธิภาพมากกว่าการให้น้ำหนักคำในแบบ Unsupervised Term Weighting เนื่องจากการให้น้ำหนักคำในรูปแบบนี้เป็นใช้ประโยชน์จากข้อมูลที่อยู่ในคลังข้อมูลการฝึกอบรม

ในปี ค.ศ. 2011 Shoushan Li และคณะได้ทำงานวิจัย Imbalance Sentiment Classification [23] เพราะเล็งเห็นปัญหาในการจำแนกความรู้สึกที่ไม่สมดุลของข้อมูล เนื่องจากวิธีก่อนหน้านี้มีปัญหาในการทำงานค่อนข้างมาก จึงได้นำเสนอ วิธีการจำแนกความรู้สึกที่ไม่สมดุล โดยเสนอโครงร่างการจัดกลุ่มแบบ under-sampling ด้วยการแบ่งเป็นกลุ่มเพื่อเอาชนะปัญหาการกระจายระดับความไม่สมดุลในการจำแนกความรู้สึกไม่สมดุล ภายใต้กรอบงานนี้ กลุ่มตัวอย่างในกลุ่มเสียงส่วนใหญ่จะถูกจัดกลุ่มเป็นกลุ่มแรก จากนั้นเลือกกลุ่มตัวอย่างจำนวนที่เหมาะสมจากแต่ละกลุ่มจากตัวอย่างการฝึกอบรมของข้อมูลส่วนใหญ่

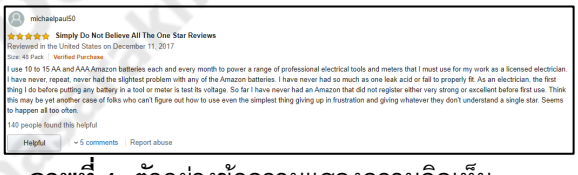
ในงานวิจัยของ Ah-Pine และ Pavel Soriano Morales [6] ศึกษาแก้ปัญหาความไม่สมดุลของข้อมูลในการวิเคราะห์ความรู้สึก (Sentiment Classification) ที่ใช้ข้อมูลจาก twitter ที่พบว่าการกระจายกลุ่มของข้อมูลมีความเอนเอียงไปกลุ่มใดกลุ่มหนึ่ง นั่นคือจำนวนข้อมูลในแต่ละกลุ่มขาดความสมดุล ดังนั้นนักวิจัยจึงนำเสนอการทำเทคนิคการสุ่มตัวอย่างแบบสังเคราะห์ (Synthetic Oversampling Techniques) สำหรับการจำแนกกลุ่มข้อความ Twitter

กระบวนการวิจัย

ในส่วนนี้จะอธิบายขั้นตอนการดำเนินงานที่นำเสนอในงานวิจัยฉบับนี้ โดยรายละเอียดสามารถแสดงได้ดังนี้

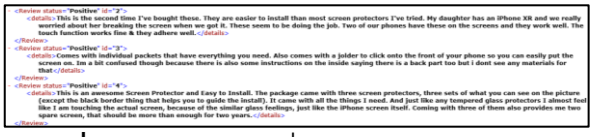
1. การรวบรวมข้อมูล

ในงานวิจัยนี้ ได้ใช้ชุดข้อความแสดงความคิดเห็นที่เกี่ยวกับอุปกรณ์อิเล็กทรอนิกส์ ซึ่งรวบรวมมาจากเว็บไซต์ Amazon โดยจะมีการแบ่งเอกสารออกเป็น 2 ชุด คือ ชุดข้อมูลสอน (Training set) และ ชุดข้อมูลทดสอบ (Test set) ซึ่งเอกสารจะอยู่ในรูปแบบ XML ข้อมูลที่ใช้ทั้งหมด 50,000 ความคิดเห็นและมีคำระหว่าง 30 ถึง 300 คำต่อหนึ่งเอกสารข้อความแสดงความคิดเห็น



ภาพที่ 1 ตัวอย่างข้อความแสดงความคิดเห็น

จากภาพที่ 1 เป็นตัวอย่างเอกสารข้อความแสดงความคิดเห็นจากเว็บ Amazon ที่ใช้ในงานวิจัยนี้โดยจะทำการดาวน์โหลดออกมาในรูปแบบ XML ซึ่งจะประกอบไปด้วย รหัส (ID), สถานะ (Status) และเนื้อหาของเอกสาร (details) ดังภาพที่ 2



ภาพที่ 2 ตัวอย่างเอกสารที่จัดเก็บรูปแบบ XML

2. การสร้างโมเดลเพื่อจำแนกความรู้สึกของบทวิจารณ์ (Classifier Modeling)

ส่วนที่ 1: การเตรียมข้อมูล (Text Preprocessing)

ขั้นตอนนี้เป็นการเตรียมข้อมูลเพื่อให้เหมาะสมต่อการนำไปสร้างโมเดลการจำแนกระดับคะแนนบทวิจารณ์ภาพยนตร์ โดยจะมีขั้นตอนดังนี้

ขั้นที่ 1: การตัดคำ (Tokenization) [4]การตัดคำ คือกระบวนการที่แยกข้อความออกเป็น “คำ” เนื่องจากคำเป็นหน่วยที่เล็กที่สุดในภาษาที่สื่อความหมายได้ สำหรับภาษาอังกฤษจะใช้ช่องว่าง (space) ที่คั่นระหว่างคำในการตัดคำ และจะใช้จุด “.” เพื่อบอกการจบประโยค

ขั้นที่ 2: การตัดคำหยุด (Stop-word Removal) [4] การตัดคำหยุด คือกระบวนการตัดคำหรือสัญลักษณ์ที่พบบ่อยมากในเอกสาร แต่คำหรือสัญลักษณ์เหล่านั้นไม่ได้ส่งผลต่อการจัดกลุ่มเอกสาร ดังนั้นเมื่อทำการตัดออกแล้วไม่ทำให้ใจความในเอกสารนั้นๆ เปลี่ยนไป การตัดคำหยุดมีความจำเป็นอย่างมากในการจัดกลุ่มเอกสารแบบอัตโนมัติ เพราะจะช่วยลดระยะเวลาในการประมวลผลลงได้เป็นอย่างมาก เนื่องจากระบบฯ จะไม่เสียเวลาในการประมวลผลคำเหล่านี้ ตัวอย่างเช่น a, an, the หรือกลุ่มคำจำพวก Article

ขั้นที่ 3: การคัดเลือกคำด้วยพจนานุกรม (English-Dictionary) เนื่องจากข้อมูลที่ใช้ในการสร้างโมเดลการจำแนกระดับคะแนนบทวิจารณ์ภาพยนตร์ เป็นข้อความที่ผู้คนเข้ามาเขียนแสดงความรู้สึกต่อภาพยนตร์เรื่องนั้นๆ ทำให้เกิดการใช้ภาษาที่ผิดเพี้ยนไปจากปกติ ในงานวิจัยฉบับนี้จึงได้มีการนำพจนานุกรมมาใช้ในการคัดเลือกคำ

ขั้นที่ 4: การเลือกคุณลักษณะ (Feature Selection) ภายหลังจากขั้นตอนข้างต้นแล้ว คลังคำที่ได้จะถูกนำเข้าสู่กระบวนการคัดเลือกคุณลักษณะด้วย Information Gain [18, 19, 35] สำหรับวิธีการคัดเลือกคุณลักษณะจะเป็นวิธีเบื้องต้นในการลดขนาดเอกสาร เนื่องจากจำนวนคุณลักษณะมีผลต่อประสิทธิภาพของการจำแนกหมวดหมู่เอกสาร เพราะอัลกอริทึมที่ใช้ในการเรียนรู้เพื่อสร้างตัวจำแนกหมวดหมู่เอกสารโดยทั่วไปไม่สามารถรองรับการทำงานกับจำนวนคุณลักษณะของเอกสารที่สูงมากได้ดี

ภาพที่ 3 ตัวอย่างการใช้ภาษาที่ผิดปกติ

ดังนั้น การลดขนาดของเอกสารจึงเป็นขั้นตอนหนึ่งที่จะต้องกระทำก่อน และในงานวิจัยฉบับนี้จะใช้ค่าเกนสารสนเทศ (IG: Information Gain) เป็นตัวลดคุณ ลักษณะของเอกสาร ซึ่งค่า *IG* จะคำนวณจากจำนวนบิตที่ได้รับสำหรับการทำนายกลุ่ม โดยการดูจากการมีอยู่หรือไม่มีอยู่ของคำในเอกสาร ให้ C_i , ..., C_K แทนเซตที่เป็นไปได้ของกลุ่ม ค่า *IG* ของคำ *w* นิยามโดย

$$\begin{aligned} IG(w) = & -\sum P(c_j)\,\log P(c_j) \\ & +P(w)\sum P(c_j|w)\,\log P(c_j|w) \\ & +P(w)\sum P(c_j|w)\,\log P(c_j|w) \end{aligned} \quad (1)$$

โดยค่า $P(C_j)$ คือความน่าจะเป็นของกลุ่มแต่ละกลุ่มที่พบ (class) ค่า $P(w)$ คือความน่าจะเป็นของคำแต่ละคำ (word) ที่พบ และค่า $P(C_j|w)$ คือความน่าจะเป็นของกลุ่มที่ได้จากคำ

เมื่อทำการคำนวณค่า *IG* ของแต่ละคุณลักษณะที่ได้ จากนั้นจะทำการตัดคุณลักษณะที่มีค่า *IG* เท่ากับ 0 ทิ้งทั้งหมด เพราะแสดงว่าคำๆ นั้นไม่มีความสำคัญต่อการจัดกลุ่มเอกสาร อีกทั้งยังช่วยลดระยะเวลาที่ระบบใช้ในการประมวลผล

ขั้นที่ 5: การสร้างตัวแทนเอกสารและการให้น้ำหนักคำ (Document Representation and Term Weighting)

ในขั้นตอนนี้จะเป็นการนำเสนอเอกสารในรูปแบบ Vector Space Model [17] เพื่อแสดงให้เห็นถึงความ สัมพันธ์ระหว่างเอกสารและคำที่

ปรากฏในเอกสาร พร้อมทั้งการให้น้ำหนักของคำเพื่อแสดงว่าคำๆ นั้นมีความสำคัญกับเอกสารมากน้อยเพียงใด ซึ่งถ้าหากค่าน้ำหนักของคำใดมีค่ามาก ก็แสดงว่ามีความสำคัญและสามารถบ่งชี้ถึงเอกสารสูง โดยการให้น้ำหนักคำจะมีอยู่ 5 รูปแบบ คือ

รูปแบบที่ 1: การให้น้ำหนักคำแบบ *tf-idf* [18]

เมื่อ *tf* เป็น local weight ที่เป็นความถี่ของ

$$ICF(t_i) = (1 + log(\frac{M}{CF(t_i)})) \quad (2.2)$$

คำหนึ่งๆ ที่พบในแต่ละเอกสาร และ *idf* ก็คือ global weight ที่เป็นการหาส่วนกลับของความถี่ของคำในเอกสาร หรือที่เรียกว่าระบบน้ำหนักความถี่เอกสารผกผัน

$$\boldsymbol{idf} = \mathbf{log}(\boldsymbol{N}/\boldsymbol{df}) \quad (1)$$

โดยที่ *N* คือจำนวนเอกสารทั้งหมดในคลัง และ *df* คือจำนวนเอกสารที่มีคำนั้นๆ ปรากฏอยู่

$$\boldsymbol{tf} - \boldsymbol{idf} = \boldsymbol{tf} \times \boldsymbol{idf} \quad (2)$$

รูปแบบที่ 2 :การให้น้ำหนักระยะยาวตาม *Delta TF-IDF*

Delta TF-IDF ถูกเสนอโดย Martineau และ *Finin* [19] มันคำนวณความแตกต่างของคะแนน *TF-IDF* ในคลาสที่เป็นบวกและลบเพื่อปรับปรุงความแม่นยำ ในฐานะที่เป็น STW จะพิจารณาการกระจายของคุณสมบัติระหว่างสองคลาสก่อนการจำแนกประเภทการรับรู้และการเพิ่มความสูงของผลคำที่แตกต่าง *Delta TF-IDF* ช่วยเพิ่มความสำคัญของคำที่กระจายอย่างไม่สม่ำเสมอระหว่างคลาสบวกและคลาสลบ โดยที่นี้ N_p และ N_n คือจำนวนของเอกสารในคลาสบวกและลบตามลำดับ ส่วน *A* และ *C* แสดงความถี่เอกสารของคำว่า t_i ในคลาสบวกและลบตามลำดับ ดังสมการที่ 3

$$w_{\&TF.IDF}(t_i) = TF(t_i, d_j) \times log_2(\frac{N_p \times C + 1.5}{A \times N_n + 1.5}) \quad (3)$$

รูปแบบที่ 3: การให้น้ำหนักระยะยาวตาม *TF-IDF-ICF*

TF-IDF-ICF เป็นรูปแบบการควบคุมน้ำหนักตามแบบ *TF-IDF* แบบดั้งเดิม อย่างไรก็ตามมันเพิ่มปัจจัยความถี่ผกผันในคลาส (Inverse Class Frequency : *ICF*) [8] เพื่อให้ค่าน้ำหนักที่สูงขึ้นไปยังคำที่หายากที่เกิดขึ้นน้อยกว่าในเอกสาร (IDF) และ Class (ICF) และใน (2.2) *M* คือจำนวนคลาสในคอลเลกชันและ *CF(t_i)* สอดคล้องกับความถี่ของคลาสที่คำ t_i ปรากฏในคอลเลกชัน *TF-IDF-ICF* แสดงใน (4)

$$w_{TF.ICF}(t_i) = TF(t_i, d_j) \times IDF(t_i) \times ICF(t_i) \quad (4)$$

รูปแบบที่ 4: ระยะน้ำหนักตาม *TF-RF*

*TF-RF (*Term Frequency - Relevance Frequency) [18] ได้รับการเสนอ เช่นเดียวกับ *Delta TF-IDF, TF-RF* คำนึงถึงการกระจายคำศัพท์ในชั้นเรียนทั้งบวกและลบ อย่างไรก็ตามมีการพิจารณาเฉพาะเอกสารที่มีคำดังกล่าวนั่นคือ ความเกี่ยวข้องของความถี่ *(*RF) ของข้อกำหนด *TF-RF* ถูกระบุใน *(*2.3) โดยที่ตัวหารน้อยที่สุดคือ 1 เพื่อหลีกเลี่ยงการหารด้วยศูนย์

$$w_{TF.RF}(t_i) = TF(t_i, d_j) \times log_2(2 + \frac{A}{max(1, C)}) \quad (5)$$

รูปแบบที่ 5: ระยะน้ำหนักตาม *TF-IGM*

ระยะความถี่-ช่วงเวลาแรงโน้มถ่วงผกผัน (Term Frequency - Inverse Gravity Moment : *TF-IGM*) [20] ถูกนำเสนอให้วัดความไม่สม่ำเสมอหรือความเข้มข้นของการแจกแจงคำศัพท์ระหว่างคลาสซึ่งสะท้อนให้เห็นถึงอำนาจการจำแนกชั้นข้อตกลง

สมการ *IGM* มาตรฐานกำหนดอันดับ (r) ตามความเข้มข้นของการแจกแจงระหว่างคลาสของคำซึ่งคล้ายกับแนวคิดของ “แรงโน้มถ่วง

โมเมนต์ (Gravity Moment : *GM*)" จากฟิสิกส์ *IGM* ถูกระบุใน (6) โดยที่ f_{ir} $(r = 1, 2, \ldots, M)$ ระบุจำนวนเอกสารที่มีคำว่า t_i ในคลาส $r-th$ ซึ่งส่วนโค้งเรียงตามลำดับจากมากไปน้อย ดังนั้น f_{il} จึงแสดงความถี่ของ t_i ในคลาสที่ปรากฏบ่อยที่สุด

$$IGM(t_i) = \left(\frac{f_{i1}}{\sum_{r=1}^{M} f_{ir} \times r}\right) \quad (6)$$

โดยน้ำหนักเทอม *TF-IGM* นั้นกำหนดตาม $IGM(t_i)$ ดังที่แสดงใน (7) ค่า λ คือ สัมประสิทธิ์แบบปรับได้ที่ใช้เพื่อรักษาสมดุลสัมพัทธ์ระหว่างปัจจัยทั่วโลก และท้องถิ่นในน้ำหนักของคำ สัมประสิทธิ์ λ มีค่าเริ่มต้นที่ 7.0 และสามารถตั้งเป็นค่าระหว่าง 5.0 ถึง 9.0 [20] สมการ 8 นำเสนอ SQRT_TF-IGM ซึ่งคำนวณสแควร์รูทของ *TF* ซึ่งเป็นเทคนิคในการรับน้ำหนักในระยะที่สมเหตุสมผลมากขึ้นโดยลดผลกระทบของ *TF* สูง [9]

$$w_{TF.IGM}(t_i) = TF(t_i, d_j) \times (1 \times \lambda \times IGM(t_i)) \quad (7)$$

$$w_{SQRT\_TF-IGM}(t_i) = \sqrt{TF(t_i, d_j)} \times (1 \times \lambda \times IGM(t_i)) \quad (8)$$

ส่วนที่ 2: การสร้างโมเดลเพื่อการจำแนกระดับคะแนนของบทวิจารณ์ภาพยนตร์

ขั้นตอนนี้เป็นขั้นตอนของการสร้างโมเดลเพื่อการจำแนกระดับคะแนนของบทวิจารณ์ด้วยอัลกอริทึมแบบมีผู้สอน (Supervised Learning) ดังนี้

1. อัลกอริทึมนาอีฟเบย์ (Naïve Bayes)

นาอีฟเบย์ (Naïve Bayes) [10, 13] เป็นการเรียนรู้อย่างง่าย เป็นวิธีการจำแนกประเภทของข้อมูลที่มีประสิทธิภาพวิธีหนึ่ง และเหมาะกับการนำมาใช้กับกรณีที่มีเซตตัวอย่างเป็นจำนวนมาก และแต่ละคุณสมบัติ (Attribute) ของตัวอย่างเป็นอิสระต่อกัน โดยนำการจำแนกประเภทนาอีฟเบย์มาประยุกต์ใช้ในการจำแนกประเภทของเอกสาร (Document Classification) พบว่ายังสามารถใช้งานได้ดีไม่ต่างจากการจำแนกวิธีการอื่นๆ และวิธีการไม่มีความซับซ้อน

การกำหนดความน่าจะเป็นของข้อมูลเป็นกลุ่ม V_j สำหรับข้อมูลที่มีคุณสมบัติ *n* ตัว ใช้สัญลักษณ์ว่า $P(a_1, a_2, \ldots a_n)$ คือ

$$\boldsymbol{P(v_j \mid a_1, a_2, \ldots, a_n) = \prod_{i=1}^{n} P(a_i \mid v_j)} \quad (9)$$

โดยที่ $\prod$ หมายถึงผลคูณของค่า $P(a_i \mid v_j)$ เมื่อ *i* และ *j* มีค่าเท่ากับ *1, 2, 3, …, n*

วิธีการเรียนรู้เบย์อย่างง่ายไปใช้มีวิธีดังต่อไปนี้คือ

(1) หาค่าความน่าจะเป็นของคำที่พบในแต่ละกลุ่มโดยนำค่า $P(a_1,a_2,\ldots,a_n \mid v_j)$ จากสมการมาคูณกับค่าความน่าจะเป็นของกลุ่มนั้นๆ คือ $P(v_j)$ ได้เท่ากับ V_{NB}

(2) นำค่าที่ได้มาเปรียบเทียบกัน กลุ่มที่มีความน่าจะเป็นสูงสุดคือกลุ่มที่ข้อมูลนั้นอยู่ และจะถูกจัดเข้าไป เขียนเป็นสมการได้คือ

$$\boldsymbol{v_{NB} = argmax\, P(v_j) \times \prod_{i=1}^{n} P(a_i \mid v_j)} \quad (10)$$

ในงานวิจัยฉบับนี้ จะสร้างโมเดลการจำแนกระดับคะแนนบทวิจารณ์แบบมัลติโนเมียลนาอีฟเบย์ (Multinomial Naïve Bayes) ซึ่งเป็นการจำแนกระดับคะแนนบทวิจารณ์เป็น 5 กลุ่ม คือ *Very bad, Bad, Neutral, Good* และ *Very Good* โดยมีขั้นตอนดังนี้

ขั้นที่ 1: การหาความน่าจะเป็นของแต่ละกลุ่ม

$$\boldsymbol{P(\ \ _j) = \frac{count(v_j)}{\sum_{i=1}^{j} (v_i)}} \quad (11)$$

ขั้นที่ 2: การหาความน่าจะเป็นของคำในแต่ละกลุ่ม

$$\boldsymbol{P(a_i \mid v_j) = \frac{count(a_i, v_j)}{count(v_i)}} \quad (12)$$

แต่ในบางครั้งการหาความน่าจะเป็นโดยใช้ Naïve Bayes อาจจะมีกรณีที่ค่าความถี่ของคำที่เกิดขึ้นเป็น 0 หรือก็คือคำที่อยู่ในถุงคำไม่ปรากฏอยู่ในเอกสาร ทำให้ค่าความน่าจะเป็นของคำนั้นเป็น 0 ตามไปด้วย ซึ่งไม่เป็นที่ยอมรับในทางสถิติที่โอกาสในการพยากรณ์จะมีค่าเป็นศูนย์ และเพื่อหลีกเลี่ยงกรณีนี้การสร้างโมเดลการจำแนกเอกสารด้วยนาอีฟเบย์มักจะมีการทำ Laplace Smoothing [32] ซึ่งเป็นลักษณะการทำ Normalization โดยจะมีการเพิ่มค่าความถี่ข้อมูลเข้าไปอีกครั้งละ 1 และบวกเพิ่มค่าความถี่รวมด้วยค่าคงที่ k จากคำทั้งหมด n คำ และกลุ่มทั้งหมด m กลุ่ม ซึ่งวิธีการนี้เป็นที่นิยมในการสร้างโมเดลเพื่อการจำแนกเอกสารด้วยนาอีฟเบย์ ดังนั้นจึงได้สมการนาอีฟเบย์ที่ปรับแล้ว ดังนี้

$$P(a_i \mid v_j) = \frac{1 + count(a_i, v_j)}{k + count(v_i)} \qquad (13)$$

2. อัลกอริทึมเพื่อนบ้านใกล้ที่สุด (K-Nearest Neighbor)

อัลกอริทึมเพื่อนบ้านใกล้ที่สุด (K-Nearest Neighbor) [27] เป็นอัลกอริทึมที่ใช้ในการจัดกลุ่มข้อมูลที่ไม่ซับซ้อนเข้าใจง่าย ซึ่งวิธีนี้จะสามารถสร้างโมเดลที่มีประสิทธิภาพได้แม้เงื่อนไขที่ใช้ในการตัดสินใจจะมีความซับซ้อนก็ตาม ซึ่งอัลกอริทึมเพื่อนบ้านใกล้ที่สุดจะเป็นการจำแนกประเภทข้อมูลโดยขึ้นกับข้อมูลที่มีคุณสมบัติใกล้เคียงที่สุด *K* ตัวจากชุดข้อมูลตัวอย่าง แล้วเลือกคลาสที่สมาชิกส่วนใหญ่ที่อยู่ในกลุ่ม *K* ดังกล่าว สังกัดอยู่มากที่สุดให้กับ สมาชิกใหม่ การจำแนกประเภทข้อมูลโดยใช้ข้อมูลข้างเคียง *K* ตัวจะประกอบด้วยแอททริบิวต์หลายตัวแปร X_i ซึ่งจะนำมาใช้ในการแบ่งกลุ่ม Y_i โดยระบุค่าตัวเลขจำนวนเต็มบวกให้กับ *K* ซึ่งค่านี้จะเป็นตัวบอกจำนวนของกรณี (Case) ที่จะต้องค้นหาในการทำนายกรณีใหม่ โดยในที่นี้จะกำหนด *1-KNN* หมายถึง อัลกอริทึมนี้จะค้นหา 1 กรณีที่มีลักษณะใกล้เคียงกับกรณีใหม่ (1 Nearest Cases) การนำระยะทางที่หาได้จากสมาชิกในข้อมูลตัวอย่างฝึกฝน มาเรียงลำดับจากน้อยไปหามาก แล้วเลือกสมาชิกที่มีระยะทาง (Distance) ใกล้เคียงที่สุดออกมา *K* ตัว โดยใช้การวัดระยะทางแบบ *Euclidean distance* [28] ซึ่งมีหลักการ คือการวัดระยะทางระหว่างสองวัตถุ ถ้าวัตถุห่างกันมากแสดงว่าวัตถุนั้นมีความคล้ายคลึงกันน้อย ถ้าระยะทางมีค่าน้อยก็แสดงว่ามีความคล้ายคลึงกันมาก โดยที่ค่า p_i แทน คุณสมบัติจากฐานข้อมูล q_i แทนคุณสมบัติที่ผู้ใช้ระบุ

$$E(p, q) = \sqrt{\sum_{i=0}^{n} (p_i - q_i)^2} \qquad (14)$$

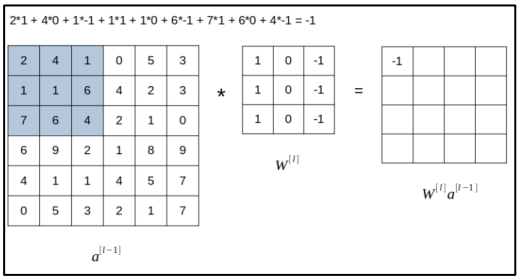
3. อัลกอริทึมโครงข่ายประสาทคอนโวลูชัน (Convolution Neural Network)

CNN ได้รับการการนำเสนอ เพื่อให้ได้ผลลัพธ์ที่น่าประทับใจในภารกิจที่สำคัญในทางปฏิบัติของการจัดหมวดหมู่ประโยค ซึ่ง CNN สามารถใช้ประโยชน์จากการแทนคำแบบกระจายโดยการแปลงโทเค็น (Tokens) ที่ประกอบด้วยแต่ละประโยคเป็นเวกเตอร์ก่อนแล้วสร้างเมทริกซ์เพื่อใช้เป็นอินพุต

Convolutional Neural Network หรือ CNN ซึ่งเป็นโครงสร้าง Neural network แบบพิเศษที่มีความสามารถในการจำแนกข้อมูลได้ดีกว่า Neural network ทั่วไปมาก โดย CNN คือการที่ใช้ Layer ชนิดพิเศษ ที่เรียกว่า Convolution layer ซึ่งทำหน้าที่สกัดเอาส่วนต่างๆ ของข้อมูลออกมา CNN จะใช้ Convolution layer มาประกอบกับ Layer ชนิดอื่น เช่น Pooling layer แล้วนำกลุ่ม Layer ดังกล่าวมาซ้อนต่อๆ กัน โดยอาจเปลี่ยน Hyperparameter บางอย่าง เช่นขนาดของ Filter layer (ซึ่งเป็นส่วนหนึ่งของ Convolution layer) และจำนวน Channel ของ layer วิธีการนำเอาส่วน

ต่างๆ มาประกอบกันนี้ เรียกว่าเป็นโครงสร้าง (Architecture) ของ CNN ซึ่งมีหลายแบบ เช่น LeNet, AlexNet, VGG, ResNet, Inception Network เป็นต้น ส่วนประกอบต่างๆ ของ CNN ซึ่งเป็นพื้นฐานที่เป็นส่วนสำคัญในการทำงานของ CNN ดังนี้

3) Convolution layer



ภาพที่ 3 ตัวอย่างการคำนวณ Convolution

จากภาพที่ 3 สมมุติเรามี Matrix ซ้ายมือขนาด 6x6 และมี Matrix ตรงกลาง ซึ่งเรียกว่า Filter หรือ Kernel ขนาด 3x3 เราจะนำเฉพาะ 3x3 ช่องแรกของ Matrix แรก มาคูณแบบ Element-wise กับ Filter matrix แล้วนำผลที่ได้แต่ละค่า (ซึ่งมีทั้งสิ้น 9 ค่า) มาบวกกัน แล้วนำไปสู่ในแถวแรกคอลัมน์แรกของ Matrix ที่สามซึ่งเป็นผลลัพธ์ โดยในภาพ ผลลัพธ์ที่ว่า เท่ากับ -1

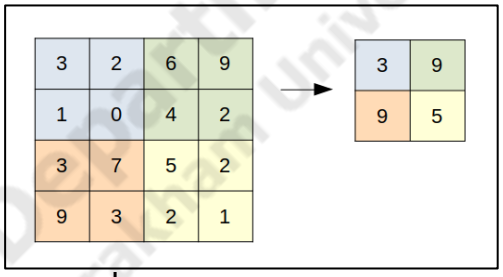
ถัดมา เราจะเลื่อนกรอบขนาด 3x3 ใน Matrix แรกไปทางขวา 1 ช่อง แล้วทำแบบเดิม ผลลัพธ์ที่ได้ นำไปใส่ในแถว 1 ช่อง 2 ของ Matrix ผลลัพธ์ ทำไปเรื่อยๆ จนสุดทาง แล้วเลื่อนกรอบ 3x3 ลงมาด้านล่าง 1 ช่อง (ชิดขอบด้านซ้ายมือ) แล้วทำแบบเดิม จนกระทั่งเติมค่าใน Matrix ผลลัพธ์จนเต็ม

กระบวนการนี้ เรียกว่า Convolution ซึ่งแสดงสัญลักษณ์ด้วย $*$ ส่วน Neural network ที่มี Layer ที่ใช้กระบวนการ Convolution นี้อย่างน้อย 1 Layer เราก็เรียกว่า Convolutional neural network

4) Pooling layer

หลังจากที่ข้อมูลผ่าน Convolution layer แล้ว บ่อยครั้งที่จะถูกส่งเข้า Layer อีกแบบหนึ่งที่เรียกว่า Pooling layer

หน้าที่ของ Pooling layer คือการสกัดเอาส่วนที่สำคัญที่สุดของข้อมูล และเพิ่มประสิทธิภาพการประมวลผลให้รวดเร็วยิ่งขึ้น กลไกของ Pooling layer นั้นเรียบง่ายมาก คือการสกัดเอาเฉพาะค่าสูงสุดของ Grid เก็บไว้ใน Output เช่นจากภาพที่ 4 แสดง Pooling layer ขนาด 2x2 โดยมีค่า Stride s=2:



ภาพที่ 4 ตัวอย่างการทำ Pooling layer

Pooling layer ที่สกัดเอาเฉพาะค่าสูงสุดของ Grid เก็บไว้ เรียกว่า Max pooling ซึ่งเป็นรูปแบบที่ใช้บ่อยที่สุด นอกจากนั้นยังมี Average pooling ซึ่งหาค่าเฉลี่ยของ Grid เก็บไว้ แต่ใช้น้อยกว่า Max pooling มาก หลังจากที่ทำ Pooling layer เสร็จ ก็จะได้ feature map หรือ feature vector ที่จะนำไปทำเป็น model สำหรับทดสอบกับชุดข้อมูลอื่นๆ

ส่วนที่ 3: การวัดประสิทธิภาพโมเดลเพื่อการจำแนกระดับคะแนนของบทวิจารณ์ภาพยนตร์

เป็นขั้นตอนการประเมินโมเดลเพื่อใช้ในการจัดกลุ่มเอกสารก่อนการนำไปใช้งานจริงที่โดยทั่วไป จะใช้เทคนิคมาตรฐาน [22] คือ

การค่าความระลึก (Recall) ซึ่งจะเป็นอัตราส่วนของเอกสารที่จัดกลุ่มได้จากเอกสารทั้งหมดที่มีอยู่ โดยจะนำค่าจากตาราง *Confusion-matrix* มาใช้ในการคำนวณหาค่าความระลึกได้ดังนี้

$$\boldsymbol{Recall} = \frac{\boldsymbol{tp}}{\boldsymbol{tp} + \boldsymbol{fn}} \qquad (15)$$

การวัดค่าความแม่นยำ (Precision) เป็นอัตราส่วนของเอกสารที่จัดกลุ่มได้และถูกต้อง ส่วนด้วยจำนวนของเอกสารที่จัดกลุ่มได้

$$\boldsymbol{Precision} = \frac{\boldsymbol{tp}}{\boldsymbol{tp} + \boldsymbol{fp}} \qquad (16)$$

การวัดค่า F-measure หรือ F1 เป็นการพิจารณาค่าความสัมพันธ์ระหว่างค่าความระลึก และค่าความแม่นยำ

$$\boldsymbol{F - measure} = \mathbf{2} \times \frac{\boldsymbol{Precision} \times \boldsymbol{Recall}}{\boldsymbol{Precision} + \boldsymbol{Recall}} \qquad (17)$$

โดยที่ค่า *F* จะมีค่าระหว่าง 0 ถึง 1 ซึ่งถ้าหากค่า *F-measure* มีค่าเข้าใกล้ 1 มากเท่าไหร่ ก็จะหมายถึงการจัดกลุ่มเอกสารนั้นมีประสิทธิภาพ และมีความถูกต้องมากขึ้นเท่านั้น

ผลการวิจัย

ผลการประเมินประสิทธิภาพโมเดลการจำแนกบทวิจารณ์สินค้าอิเล็กทรอนิกส์

ตารางที่ 1 ผลการประเมินที่ใช้ในการจำแนกบทวิจารณ์สินค้าอิเล็กทรอนิกส์ด้วยอัลกอริทึม KNN

การให้น้ำหนักคำ	สัดส่วนเอกสารที่ใช้ในการสร้างโมเดล (ร้อยละ)	ค่าความระลึก	ค่าความแม่นยำ	ค่าเฉลี่ย F-measure
TF-IDF	100:10	0.5162	0.5021	0.5074
	100:20	0.5447	0.5246	0.5342
	100:30	0.5941	0.5702	0.5821
	ค่าเฉลี่ย	**0.5462**	**0.5346**	**0.5363**
Delta TF-IDF	100:10	0.5362	0.5744	0.5546
	100:20	0.5414	0.5204	0.5366
	100:30	0.5922	0.5702	0.5812
	ค่าเฉลี่ย	**0.5566**	**0.5550**	**0.5574**
TF-ICF-IDF	100:10	0.5610	0.5532	0.5564
	100:20	0.6012	0.5830	0.5912
	100:30	0.6332	0.6242	0.6262
	ค่าเฉลี่ย	**0.5967**	**0.5834**	**0.5866**
TF-RF	100:10	0.6601	0.6410	0.6532
	100:20	0.6911	0.6862	0.6812
	100:30	0.7135	0.7046	0.7062
	ค่าเฉลี่ย	**0.6863**	**0.6734**	**0.6767**
TF- IGM	100:10	0.6956	0.6884	0.6894
	100:20	0.7103	0.7014	0.7063
	100:30	0.7345	0.7264	0.7201
	ค่าเฉลี่ย	**0.7134**	**0.7054**	**0.7052**

ตารางที่ 2 ผลการประเมินที่ใช้ในการจำแนกบทวิจารณ์สินค้าอิเล็กทรอนิกส์ด้วยอัลกอริทึมนาอีฟเบย์

การให้น้ำหนักคำ	สัดส่วนเอกสารที่ใช้ในการสร้างโมเดล (ร้อยละ)	ค่าความระลึก	ค่าความแม่นยำ	ค่าเฉลี่ย F-measure
TF-IDF	100:10	0.5546	0.5350	0.5421
	100:20	0.5747	0.5542	0.5632
	100:30	0.6304	0.6346	0.6323
	ค่าเฉลี่ย	**0.5767**	**0.5764**	**0.5737**
Delta TF-IDF	100:10	0.5431	0.5294	0.5374
	100:20	0.5766	0.5546	0.5675
	100:30	0.6445	0.6328	0.6368
	ค่าเฉลี่ย	**0.5769**	**0.5568**	**0.5734**
TF-ICF-IDF	100:10	0.6143	0.6233	0.6176
	100:20	0.6436	0.6312	0.6366
	100:30	0.6744	0.6561	0.6674
	ค่าเฉลี่ย	**0.6424**	**0.6337**	**0.6339**
TF-RF	100:10	0.6332	0.6242	0.6262
	100:20	0.6911	0.6862	0.6812
	100:30	0.7135	0.7046	0.7062
	ค่าเฉลี่ย	**0.6863**	**0.6734**	**0.6767**
TF- IGM	100:10	0.6977	0.6945	0.6912
	100:20	0.7216	0.7264	0.7235
	100:30	0.7448	0.7468	0.7482
	ค่าเฉลี่ย	**0.7287**	**0.7266**	**0.7232**

ตารางที่ 3 ผลการประเมินที่ใช้ในการจำแนกบทวิจารณ์สินค้าอิเล็กทรอนิกส์ด้วยอัลกอริทึม CNN

การให้น้ำหนักคำ	สัดส่วนเอกสารที่ใช้ในการสร้างโมเดล (ร้อยละ)	ค่าความระลึก	ค่าความแม่นยำ	ค่าเฉลี่ย F-measure
TF-IDF	100:10	0.5562	0.5644	0.5546
	100:20	0.5914	0.6304	0.6066
	100:30	0.6398	0.6202	0.6412
	ค่าเฉลี่ย	**0.5966**	**0.6150**	**0.6074**

ตารางที่ 3 ผลการประเมินที่ใช้ในการจำแนกบทวิจารณ์สินค้าอิเล็กทรอนิกส์ด้วยอัลกอริทึม CNN (ต่อ)

การให้น้ำหนักคำ	สัดส่วนเอกสารที่ใช้ในการสร้างโมเดล (ร้อยละ)	ค่าความระลึก	ค่าความแม่นยำ	ค่าเฉลี่ย F-measure
Delta TF-IDF	100:10	0.5610	0.5832	0.5764
	100:20	0.6112	0.6530	0.6412
	100:30	0.6632	0.6742	0.6662
	ค่าเฉลี่ย	**0.6267**	**0.6534**	**0.6266**
TF-ICF-IDF	100:10	0.6342	0.6384	0.6362
	100:20	0.6871	0.6872	0.6852
	100:30	0.7135	0.7066	0.7102
	ค่าเฉลี่ย	**0.6782**	**0.6774**	**0.6761**
TF-RF	100:10	0.6221	0.6512	0.6354
	100:20	0.7398	0.7130	0.7212
	100:30	0.8132	0.7954	0.8054
	ค่าเฉลี่ย	**0.7257**	**0.7198**	**0.7282**
TF- IGM	100:10	0.6552	0.6752	0.6652
	100:20	0.7598	0.7430	0.7512
	100:30	0.8142	0.8214	0.8112
	ค่าเฉลี่ย	**0.7430**	**0.7438**	**0.7441**

สำหรับรูปแบบการให้น้ำหนักคำแต่ละรูปแบบจะเห็นได้ชัดว่ารูปแบบการให้น้ำหนักคำ *TF-IGM* มีค่าเฉลี่ยสูงสุดในทุกอัลกอริทึม เนื่องจากรูปแบบการให้น้ำหนักคำแบบ *TF-IGM* นั้น ถูกนำเสนอให้วัดความไม่สม่ำเสมอหรือความเข้มข้นของการแจกแจงคำศัพท์ระหว่างคลาสซึ่งสะท้อนให้เห็นถึงอำนาจการจำแนกชั้นข้อตกลง จึงทำให้เห็นความชัดเจนของการแยกข้อมูลในแต่ละคลาสเป็นอย่างดี ซึ่งเมื่อนำรูปแบบการให้น้ำหนักคำไปใช้กับอัลกอริทึม *CNN* แล้วทำให้เห็นว่าหากเอกสารมีข้อมูลไม่สมดุลมากการให้น้ำหนักคำแบบ *TF-IGM* ที่ใช้กับอัลกอริทึม *CNN* สามารถแก้ปัญหาได้ดีที่สุดเมื่อเอกสารมีสัดส่วนที่ 100: 10 โดยมีค่าเฉลี่ยอยู่ที่ 0.6652 เมื่อเทียบกับรูปแบบอื่นๆ รองลงมาคือรูปแบบการให้น้ำหนักคำแบบ *TF-ICF-IDF* ที่มีค่าเฉลี่ยอยู่ที่ 0.6362 และรูปแบบที่มีค่าเฉลี่ยต่ำสุดคือ *TF-IDF* ที่มีค่าเฉลี่ยอยู่ที่ 0.5546

สำหรับรูปแบบการให้น้ำหนักที่มีค่าเฉลี่ยมากสุดที่ทดสอบกับชุดข้อมูลมีสัดส่วน 100:20 และ 100:30 นั้น คือรูปแบบ *TF-IGM* ที่ทดสอบกับอัลกอริทึม *CNN* เช่นเดียวกับสัดส่วน 100:10 โดยมีค่าเฉลี่ย *F-measure* อยู่ที่ 0.7512 และ 0.8112 ตามลำดับ ซึ่งสัดส่วน 100:30 เป็นค่าที่สูงที่สุดในการทดสอบรูปแบบการให้ทั้งหมด และเห็นได้ชัดว่าหากข้อมูลมีค่าความไม่สมดุลต่างกันนั้นก็จะให้การจำแนกข้อมูลมีประสิทธิภาพมาก

วิจารณ์และสรุปผล

เนื่องจากบ่อยครั้งที่ การจำแนกเอกสารที่ไม่สมดุลกันนั้นมีการเอนเอียงการให้คำแนนไปฝั่งที่มี

ข้อมูลมากกว่าเนื่องจากมีข้อมูลที่คอบครุมการทำนายที่ดีกว่า

ดังนั้นงานวิจัยฉบับนี้จึงได้นำเสนอวิธีการการจำแนกข้อมูลที่ไม่สมดุลด้วยการให้น้ำหนักคำเปรียบเทียบ 2 รูปแบบหลักคือ UTW และ STW โดย UTW ใช้รูปแบบการให้น้ำหนักคำที่ได้รับความนิยมมากที่สุดคือ TF-IDF และ STW ใช้ทั้งหมด 4 รูปแบบคือ Delta TF-IDF, TF-ICF-IDF, TF-RF และ TF-IGM โดยผลที่ได้คือการให้น้ำหนักคำแบบ STW มีประสิทธิภาพในการจำแนกข้อมูลที่ไม่สมดุลมากกว่ารูปแบบการให้น้ำหนักคำแบบ UTW ซึ่งได้แก่การให้น้ำหนักคำแบบ TF-IGM โดยใช้อัลกอริทึม CNN ในการสร้างโมเดล มีค่าเฉลี่ย F-measure สูงที่สุดอยู่ที่ 74.41%

เอกสารอ้างอิง

[1] B. Pang, L. Lee, and S. Vaithyanathan, “Thumbs up? Sentiment Classification using Machine Learning Techniques,” May 2002, Accessed: Aug. 06, 2020. [Online]. Available: http://arxiv.org/abs/cs/0205070.

[2] Y. Li, G. Sun, and Y. Zhu, “Data imbalance problem in text classification,” *Proc. - 3rd Int. Symp. Inf. Process. ISIP 2010*, pp. 301–305, 2010, doi: 10.1109/ISIP.2010.47.

[3] Y. Liu, H. T. Loh, and A. Sun, “Imbalanced text classification: A term weighting approach,” *Expert Syst. Appl.*, vol. 36, no. 1, pp. 690–701, 2009, doi: https://doi.org/10.1016/j.eswa.2007.10.042.

[4] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling Technique,” 2002.

[5] R. Longadge and S. Dongre, “Class Imbalance Problem in Data Mining Review,” May 2013, Accessed: Aug. 06, 2020. [Online]. Available: http://arxiv.org/abs/1305.1707.

[6] J. Ah-Pine and E. P. S. Morales, “A study of synthetic oversampling for twitter imbalanced sentiment analysis,” *CEUR Workshop Proc.*, vol. 1646, pp. 17–24, 2016.

[7] C. Zhang, J. Bi, and P. Soda, *Feature selection and resampling in class imbalance learning: Which comes first? An empirical study in the biological domain*. 2017.

[8] F. Ren and M. G. Sohrab, “Class-indexing-based term weighting for automatic text classification,” *Inf. Sci. (Ny).*, vol. 236, pp. 109–125, 2013, doi: https://doi.org/10.1016/j.ins.2013.02.029.

[9] Y. Gu and X. Gu, “A Supervised Term Weighting Scheme for Multi-class Text Categorization BT - Intelligent Computing Methodologies,” 2017, pp. 436–447.

[10] P. Juszczak and R. P. W. Duin, “Uncertainty sampling methods for one-class classifiers.”

[11] F. Debole and F. Sebastiani, “Supervised Term Weighting for Automated Text Categorization BT - Text Mining and its Applications,” 2004, pp. 81–97.

[12] A. C. E. S. Lima and L. N. de Castro, “Automatic sentiment analysis of Twitter messages,” in *2012 Fourth International Conference on Computational Aspects*

of Social Networks (CASoN), 2012, pp. 52–57, doi: 10.1109/CASoN.2012.6412377.

[13] M. Ibrahim and M. Carman, "Undersampling Techniques to Re-balance Training Data for Large Scale Learning-to-Rank BT - Information Retrieval Technology," 2014, pp. 444–457.

[14] V. Balakrishnan and L.-Y. Ethel, "Stemming and Lemmatization: A Comparison of Retrieval Performances," *Lect. Notes Softw. Eng.*, vol. 2, no. 3, pp. 262–267, 2014, doi: 10.7763/lnse.2014.v2.134.

[15] F. Sebastiani, "Machine Learning in Automated Text Categorization." [Online]. Available: www.ira.uka.de/bibliography/Ai/automated.text.

[16] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Inf. Process. Manag.*, vol. 24, no. 5, pp. 513–523, 1988, doi: https://doi.org/10.1016/0306-4573(88)90021-0.

[17] G. Domeniconi, G. Moro, R. Pasolini, and C. Sartori, "A Comparison of Term Weighting Schemes for Text Classification and Sentiment Analysis with a Supervised Variant of tf.idf BT - Data Management Technologies and Applications," 2016, pp. 39–58.

[18] M. Lan, C. L. Tan, J. Su, and Y. Lu, "Supervised and Traditional Term Weighting Methods for Automatic Text Categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 4, pp. 721–735, 2009, doi: 10.1109/TPAMI.2008.110.

[19] J. Martineau, T. Finin, C. Fink, C. Piatko, J. Mayfield, and Z. Syed, "Delta TFIDF: An Improved Feature Space for Sentiment Analysis," *Proc. Second Int. Conf. Weblogs Soc. Media (ICWSM*, vol. 29, no. May, pp. 490–497, 2008, [Online]. Available: http://ebiquity.umbc.edu/papers/select/person/Tim/Finin/.

[20] K. Chen, Z. Zhang, J. Long, and H. Zhang, "Turning from TF-IDF to TF-IGM for term weighting in text classification," *Expert Syst. Appl.*, vol. 66, pp. 245–260, 2016, doi: https://doi.org/10.1016/j.eswa.2016.09.009.

[21] T. Dogan and A. K. Uysal, "Improved inverse gravity moment term weighting for text classification," *Expert Syst. Appl.*, vol. 130, pp. 45–59, 2019, doi: https://doi.org/10.1016/j.eswa.2019.04.015.

[22] D. M. W, "EVALUATION: FROM PRECISION, RECALL AND F-MEASURE TO ROC, INFORMEDNESS, MARKEDNESS & CORRELATION," *J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, 2011, [Online]. Available: http://dspace.flinders.edu.au/dspace/http://www.bioinfo.in/contents.php?id=51.

[23] S. Li, G. Zhou, Z. Wang, S. Y. M. Lee, and R. Wang, "Imbalanced sentiment classification," *Int. Conf. Inf. Knowl. Manag. Proc.*, pp. 2469–2472, 2011, doi: 10.1145/2063576.2063994.