

การประยุกต์ใช้การเรียนรู้ของเครื่องในการพยากรณ์ผลการแข่งขัน League of Legends จากข้อมูลช่วง Drafting Phase

จินณวัตร ญาณโกมุท^{1*} และ เสกสันต์ แสงสวัสดิ์¹

¹สาขาคอมพิวเตอร์เกมและอีสปอร์ต วิทยาลัยนวัตกรรมการผลิตและเทคโนโลยี มหาวิทยาลัยรังสิต

*Jinnawat.y65@rsu.ac.th

บทคัดย่อ

งานวิจัยนี้ศึกษาประสิทธิภาพของแนวทางการใช้แชมเปียนเอ็มเบดดิ้ง (Champion Embedding) ผ่านโครงสร้างอัตโนมัติเพื่อพยากรณ์ผลการแข่งขันเกม League of Legends ในช่วง Draft Phase โดยใช้ข้อมูลฤดูกาลแข่งขันปี 2020 จำนวน 5,612 แมตช์ เพื่อลดมิติข้อมูลจากคุณลักษณะดั้งเดิม 28 ด้าน จากการทดลองเปรียบเทียบมิติตั้งแต่ 2 มิติ ถึง 16 มิติ ผู้วิจัยเลือกใช้ 6D Embedding เป็นขนาดที่เหมาะสมที่สุด (Sweet Spot) เนื่องจากสามารถรักษาสมาคมระหว่างการเรียนรู้และความสามารถในการทั่วไป (Generalization) ได้ดีที่สุด โดยช่วยลดระดับความเสี่ยงของภาวะการเรียนรู้ไม่เพียงพอ (Underfitting) ในมิติต่ำ และภาวะการเรียนรู้จดจำข้อมูลมากเกินไป (Overfitting) ในมิติสูงได้อย่างมีประสิทธิภาพ ผลการพยากรณ์ผ่านการทดลองซ้ำ 10 รอบ (10 Random Seeds) เพื่อความน่าเชื่อถือทางสถิติ พบว่าการใช้ 6D Embedding ร่วมกับโมเดล Soft Voting ให้ค่า AUC เฉลี่ย 0.5421 ± 0.0161 และค่าความถูกต้องเฉลี่ย 0.5329 ± 0.0123 ซึ่งสูงกว่าเกณฑ์มาตรฐาน (Baselines) อย่างชัดเจน แม้ประสิทธิภาพเฉลี่ยจะใกล้เคียงกับการใช้ข้อมูลดิบ (Raw Data) แต่จากการทดสอบนัยสำคัญด้วย Paired t-test พบค่า p-value เท่ากับ 0.509 ซึ่งยืนยันว่าประสิทธิภาพของทั้งสองรูปแบบไม่มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.05 ข้อค้นพบนี้พิสูจน์ให้เห็นว่าแนวทางที่นำเสนอสามารถลดมิติข้อมูลลงได้มหาศาลถึงร้อยละ 81.25 (จาก 320 มิติ เหลือ 60 มิติ) ซึ่งช่วยลดสัญญาณรบกวนและความซับซ้อนของข้อมูลได้โดยยังรักษาความแม่นยำไว้ได้ในระดับเดียวกับการใช้ข้อมูลทั้งหมด จึงถือเป็นแนวทางที่มีแนวโน้มช่วยเพิ่มประสิทธิภาพและเสถียรภาพในการวิเคราะห์ข้อมูลเชิงกลยุทธ์เพื่อพัฒนาระบบสนับสนุนการตัดสินใจในอุตสาหกรรมอีสปอร์ตต่อไป

คำสำคัญ: อีสปอร์ต การพยากรณ์ผลการแข่งขัน แชมเปียนเอ็มเบดดิ้ง การลดมิติข้อมูล การเรียนรู้ของเครื่อง

Applying Machine Learning to Predict League of Legends Match Outcomes Using Draft-Phase Data

Jinnawat Yankomut^{1*} and Seksan Sangsawad¹

¹Department of Computer Game and Esports, College of Digital Innovation Technology, Rangsit University

*Jinnawat.y65@rsu.ac.th

Abstract

This study investigates the effectiveness of Autoencoder-based Champion Embedding for predicting League of Legends match outcomes during the Draft Phase. Utilizing a dataset of 5,612 professional matches from the 2020 season, the research aims to address the complexity and noise of high-dimensional champion features that often degrade model generalization. The methodology involved compressing 28 initial champion attributes into low-dimensional vectors ranging from 2D to 16D, with strict data leakage prevention protocols where feature standardizers were fitted exclusively on the training set. Based on 10 repeated experimental trials (10 Random Seeds) to ensure statistical reliability, 6D Embedding was identified as the "sweet spot," providing the optimal balance between reconstruction accuracy and predictive generalization while minimizing underfitting and overfitting risks. Results demonstrate that 6D Embedding paired with a Soft Voting ensemble model achieved a mean Area Under the ROC Curve (AUC) of 0.5421 ± 0.0161 and a mean Accuracy of 0.5329 ± 0.0123 , consistently outperforming established baselines. While performance was numerically close to the high-dimensional Raw Data (AUC 0.5523), a paired t-test yielded a p-value of 0.509, confirming no statistically significant difference at the 0.05 level. This confirms the primary advantage of the proposed framework: reducing feature dimensionality by 81.25% (from 320 to 60 dimensions) while maintaining predictive integrity and reducing data noise. These findings suggest that Champion Embedding is a promising approach for enhancing strategic analytics and decision-support systems in the professional esports industry.

Keywords: Esports, Match Outcome Prediction, Champion Embedding, Dimensionality Reduction, Machine Learning

1. บทนำ

ในปัจจุบันกีฬาอิเล็กทรอนิกส์ หรือ อีสปอร์ต (Esports) ได้รับการยอมรับอย่างกว้างขวางในฐานะอุตสาหกรรมดิจิทัลที่มีการเติบโตอย่างรวดเร็วและมีมูลค่าทางเศรษฐกิจสูงในระดับโลก โดยมีการคาดการณ์ว่าตลาดอีสปอร์ตจะมีมูลค่าสูงถึง 16.47 พันล้านดอลลาร์สหรัฐภายในปีพ.ศ. 2576 [1] การเติบโตดังกล่าวส่งผลให้เกิดศาสตร์ด้านการวิเคราะห์ข้อมูลอีสปอร์ต (Esports Analytics) เพื่อสนับสนุนการตัดสินใจเชิงกลยุทธ์และเพิ่มประสิทธิภาพในการแข่งขัน [2] ด้วยลักษณะการแข่งขันที่

ต้องอาศัยการวางแผนเชิงกลยุทธ์ ทักษะเฉพาะบุคคล และการประสานงานภายในทีมอย่างเป็นระบบ อีสปอร์ตจึงกลายเป็นสาขาที่ได้รับความสนใจทั้งในเชิงวิชาการและอุตสาหกรรม [3]

เกมประเภท Multiplayer Online Battle Arena (MOBA) ถือเป็นหนึ่งในแนวเกมที่ได้รับความนิยมสูงสุดในการแข่งขันระดับอาชีพ โดยผู้เล่นทั้งสองทีมต้องแข่งขันกันเพื่อทำลายฐานหลักของฝ่ายตรงข้าม [4] หนึ่งในเกมที่ได้รับความนิยมมากที่สุดคือ League of Legends (LoL) ซึ่งมีฐานผู้เล่นและผู้ชมจำนวนมากทั่วโลก [5] มีรายงานว่าการแข่งขัน League of Legends World Championship ปี พ.ศ. 2562 มียอดผู้ชมสูงถึง 100 ล้านคน ซึ่งใกล้เคียงกับจำนวนผู้ชมการแข่งขัน Super Bowl ในช่วงเวลาเดียวกัน [4]

ปัจจัยสำคัญที่ส่งผลต่อโอกาสในการชนะของเกมประเภท MOBA คือกระบวนการ Draft Phase หรือขั้นตอนการเลือกและห้ามเลือกแชมเปียนก่อนเริ่มการแข่งขัน เนื่องจากแชมเปียนแต่ละตัวมีคุณลักษณะพื้นฐาน บทบาท และความสามารถเฉพาะที่แตกต่างกัน [6] ปัจจุบันเกม League of Legends มีแชมเปียนมากกว่า 150 ตัว และแต่ละตัวมีคุณลักษณะมากกว่า 28 ด้าน [4] ดังนั้น การจัดองค์ประกอบของทีมที่มีความสอดคล้องกัน (Synergy) และสามารถตอบโต้ฝ่ายตรงข้ามได้อย่างเหมาะสม (Counter Pick) จึงเป็นปัจจัยสำคัญที่ส่งผลต่อแนวโน้มของผลการแข่งขันตั้งแต่ก่อนเริ่มเกม [7]

ด้วยความซับซ้อนของข้อมูลแชมเปียนและข้อมูลการแข่งขันที่มีปริมาณสูง เทคโนโลยี Machine Learning จึงถูกนำมาใช้ในการวิเคราะห์ข้อมูลและทำนายผลการแข่งขันในรูปแบบต่าง ๆ [8] อย่างไรก็ตาม การใช้ข้อมูลดิบ (Raw Data) โดยตรงมักมีข้อจำกัดจากสัญญาณรบกวน (Noise) และความซับซ้อนของข้อมูล ซึ่งอาจส่งผลให้โมเดลเกิดภาวะ Overfitting และลดความสามารถในการทั่วไป (Generalization) ของโมเดลเมื่อใช้งานกับข้อมูลที่ไม่เคยพบมาก่อน [5]

จากข้อจำกัดดังกล่าว ผู้วิจัยจึงนำเสนอแนวทางการศึกษาประสิทธิภาพของแนวคิดแชมเปียนเอ็มเบดดิ้ง (Champion Embedding) ด้วยเทคนิค Deep Learning ผ่านโครงสร้าง Autoencoder เพื่อสกัดคุณลักษณะสำคัญของแชมเปียนและลดมิติของข้อมูลแชมเปียนจำนวน 28 คุณลักษณะ โดยใช้ข้อมูลการแข่งขันจริงจำนวน 5,612 แมตช์จากฤดูกาลแข่งขันปี พ.ศ. 2563 [3] กระบวนการดังกล่าวมุ่งเน้นการสร้างเวกเตอร์คุณลักษณะที่สามารถสะท้อนความสัมพันธ์ขององค์ประกอบภายในทีมได้อย่างมีประสิทธิภาพ ก่อนนำไปใช้เป็นข้อมูลนำเข้าสำหรับโมเดล Machine Learning ในการพยากรณ์ผลการแข่งขัน งานวิจัยนี้คาดว่าจะช่วยประเมินแนวทางที่มีแนวโน้มช่วยเพิ่มประสิทธิภาพในการพยากรณ์ผลการแข่งขันจากข้อมูล Draft Phase และสามารถต่อยอดสู่การพัฒนาระบบวิเคราะห์เชิงกลยุทธ์ในวงการอีสปอร์ตได้ในอนาคต

2. การทบทวนวรรณกรรมและทฤษฎีที่เกี่ยวข้อง

ในการศึกษาประสิทธิภาพของแนวคิดแชมเปียนเอ็มเบดดิ้งเพื่อพยากรณ์ผลการแข่งขันเกม League of Legends (LoL) ผู้วิจัยได้ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องเพื่อใช้เป็นแนวทางในการออกแบบกระบวนการวิจัยและการประเมินผลโมเดล โดยมีรายละเอียดดังต่อไปนี้

2.1 การทำนายผลการแข่งขันในเกมประเภท MOBA

การทำนายผลการแข่งขันเกมประเภท Multiplayer Online Battle Arena (MOBA) เป็นหนึ่งในหัวข้อสำคัญของการวิเคราะห์ข้อมูลอีสปอร์ต เนื่องจากเกมประเภทดังกล่าวมีความซับซ้อนทั้งด้านกลยุทธ์ องค์ประกอบของทีม และปัจจัยระหว่างการแข่งขัน [8] ปัจจัยสำคัญที่ส่งผลต่อแนวโน้มของผลการแข่งขันคือกระบวนการ Draft Phase หรือขั้นตอนการเลือกและห้ามเลือกแชมเปียนก่อนเริ่มเกม ซึ่งมีผลต่อการจัดองค์ประกอบของทีมและความสามารถในการตอบโต้ฝ่ายตรงข้าม [4] งานวิจัยที่ผ่านมาได้มีการพัฒนาโมเดลทำนายผลการแข่งขันทั้งในรูปแบบก่อนเริ่มการแข่งขัน (Pre-match Prediction) และระหว่างการแข่งขัน (Real-time Prediction) โดย Do et al. [7] รายงานว่าการใช้ข้อมูลประสบการณ์ของผู้เล่นร่วมกับข้อมูลแชมเปียนสามารถทำนายผลการแข่งขันได้แม่นยำถึงร้อยละ 75.1 ขณะที่งานวิจัยด้าน Real-time Prediction พบว่าความแม่นยำของโมเดลมีแนวโน้มเพิ่มขึ้นตามระยะเวลาของการแข่งขันและปริมาณข้อมูลภายในเกมที่เพิ่มขึ้น [5,8]

2.2 การสร้าง Champion Embedding

Champion Embedding เป็นเทคนิคการสร้างเวกเตอร์คุณลักษณะมิติต่ำเพื่อแทนข้อมูลที่มีความซับซ้อนสูง โดยมีวัตถุประสงค์เพื่อรักษาคุณลักษณะสำคัญของข้อมูลไว้ภายใต้จำนวนมิติที่ลดลง [2] เทคนิคดังกล่าวได้รับความนิยมอย่างมากเนื่องจากสามารถช่วยลดความซับซ้อนของข้อมูลและลดผลกระทบจากสัญญาณรบกวน (Noise) ในบริบทของเกม League of Legends มีการประยุกต์ใช้ Champion Embedding ในหลายลักษณะ เช่น การแนะนำแชมเปียนที่เหมาะสมกับองค์ประกอบของทีม [6] หรือการประเมินรูปแบบการเล่นของผู้เล่นจากลักษณะการกระทำภายในเกม [9] นอกจากนี้ Sapienza et al. [3] ยังแสดงให้เห็นว่าการใช้ Deep Neural Network เพื่อเรียนรู้ความสัมพันธ์ระหว่างผู้เล่นและองค์ประกอบภายในทีมสามารถช่วยเพิ่มประสิทธิภาพในการวิเคราะห์เชิงกลยุทธ์ได้อย่างมีนัยสำคัญ

2.3 โครงสร้าง Autoencoder

Autoencoder เป็นสถาปัตยกรรม Deep Learning แบบไม่มีผู้สอน (Unsupervised Learning) ที่ถูกออกแบบมาเพื่อการสกัดคุณลักษณะสำคัญของข้อมูล (Feature Extraction) และการลดมิติข้อมูล โดยประกอบด้วยส่วน Encoder ที่บีบอัดข้อมูลให้อยู่ในรูปแบบเวกเตอร์มิติต่ำ และ Decoder ที่ทำหน้าที่สร้างข้อมูลกลับคืน [3] ในงานวิจัยด้านเกม MOBA มีการนำ Autoencoder มาใช้เพื่อสกัดคุณลักษณะสำคัญจากค่าสถานะของแชมเปียน [3,9] เทคนิคนี้ช่วยให้โมเดลสามารถเรียนรู้รูปแบบเชิงลึกและรักษาสมาดุลระหว่างการเรียนรู้ข้อมูลกับความสามารถในการทั่วไป (Generalization) ซึ่งเป็นประเด็นสำคัญในการป้องกันภาวะ Overfitting เมื่อทำงานกับข้อมูลที่มีความแปรปรวนสูง

2.4 โมเดล Machine Learning และการรายงานผลทางสถิติ

งานวิจัยด้านการทำนายผลการแข่งขันอีสปอร์ตนิยมใช้โมเดล Machine Learning หลายประเภท เช่น Logistic Regression, Random Forest และ XGBoost [6,8] นอกจากนี้เทคนิค Ensemble Learning เช่น Voting Classifier ยังเป็นที่ยอมรับในการเพิ่มความเสถียรของโมเดล [6,7] อย่างไรก็ตาม เพื่อให้ผลการเปรียบเทียบมีความน่าเชื่อถือทางวิชาการ งานวิจัยในปัจจุบันให้ความสำคัญกับการประเมินผลจากการทดลองซ้ำหลายรอบด้วยค่าเริ่มต้นที่แตกต่างกัน (Random Seeds) เพื่อรายงานค่าเฉลี่ย (Mean) และส่วนเบี่ยงเบนมาตรฐาน (Std) รวมถึงการทดสอบนัยสำคัญทางสถิติ เพื่อยืนยันว่าประสิทธิภาพที่เพิ่มขึ้นนั้นไม่ได้เกิดจากความบังเอิญของชุดข้อมูล [2]

3. วิธีดำเนินการวิจัย

งานวิจัยนี้มุ่งเน้นศึกษาประสิทธิภาพของแนวคิดแชมเปียนเอ็มเบดดิ้งเพื่อสกัดคุณลักษณะสำคัญและลดมิติข้อมูลแชมเปียน ก่อนนำไปพยากรณ์ผลการแข่งขันเกม League of Legends (LoL) โดยมีรายละเอียดขั้นตอนการดำเนินงานที่ปรับปรุงตามมาตรฐานทางสถิติดังนี้

3.1 ชุดข้อมูลและการวิเคราะห์เบื้องต้น (Dataset Analysis)

ข้อมูลที่ใช้ในการวิจัยถูกรวบรวมผ่านระบบ Official API ของ Riot Games [8,9] โดยแสดงรายละเอียดตามตารางที่ 1 ประกอบด้วยข้อมูลการแข่งขันระดับอาชีพจริงจากฤดูกาลปี พ.ศ. 2563 (Season 2020) จำนวน 5,612 แมตช์ ซึ่งระบุรายชื่อแชมเปียนของผู้เล่นทั้ง 10 ตำแหน่งและผลลัพธ์ของการแข่งขัน ร่วมกับข้อมูลคุณลักษณะพื้นฐานของแชมเปียนจำนวน 150 ตัว ใน 28 ด้าน เช่น พลังชีวิต พลังโจมตี และบทบาทของแชมเปียน [6] จากการวิเคราะห์สัดส่วนคลาสพบว่ามีแมตช์ที่ทีมชนะร้อยละ 52.82 (2,964 แมตช์) และทีมแพ้ร้อยละ 47.18 (2,648 แมตช์) ซึ่งแสดงถึงความสมดุลของชุดข้อมูลโดยมีค่าเกณฑ์มาตรฐานจากคลาสส่วนใหญ่ (Majority Class Baseline) อยู่ที่ 0.5282 ในกระบวนการเตรียมข้อมูล ผู้วิจัยได้แบ่งชุดข้อมูลออกเป็นชุดฝึกสอน (Train) ร้อยละ 80 และชุดทดสอบ (Test) ร้อยละ 20 โดยได้จัดสรรร้อยละ 25 ของชุดฝึกสอนเป็นชุดตรวจสอบ (Validation Set) ทั้งนี้ เพื่อป้องกันภาวะการรั่วไหลของข้อมูล (Data Leakage) ผู้วิจัยได้ดำเนินการคำนวณค่าเพื่อ

ปรับมาตรฐานข้อมูล (StandardScaler) จากเฉพาะข้อมูลในชุดฝึกสอนเท่านั้น แล้วจึงนำไปประยุกต์ใช้กับชุดข้อมูลตรวจสอบและชุดทดสอบในภายหลัง

ตารางที่ 1 รายละเอียดของชุดข้อมูลที่ใช้ในงานวิจัย (Dataset Description)

รายการรายละเอียด	ค่าสถิติ
ฤดูกาลแข่งขัน	2020
จำนวนแมตช์การแข่งขันทั้งหมด	5,612
จำนวนแชมป์ในชุดข้อมูล	150
จำนวนคุณลักษณะตั้งต้นต่อแชมป์	28
จำนวนแมตช์ที่ผลการแข่งขันเป็น "ชนะ" (Class 1: Win)	2,964 (52.82%)
จำนวนแมตช์ที่ผลการแข่งขันเป็น "แพ้" (Class 0: Loss)	2,648 (47.18%)
ค่าเกณฑ์มาตรฐานจากคลาสส่วนใหญ่	0.5282

3.2 การแบ่งข้อมูลและการป้องกันการรั่วไหลของข้อมูล (Data Splitting & Leakage Prevention)

ผู้วิจัยดำเนินการแบ่งข้อมูลการแข่งขันออกเป็นชุดฝึกสอน (Train) ร้อยละ 80 และชุดทดสอบ (Test) ร้อยละ 20 โดยใช้เทคนิค Stratified Sampling เพื่อรักษาสัดส่วนของคลาสให้คงเดิม และทำการทดลองซ้ำจำนวน 10 รอบด้วยค่าเริ่มต้นที่แตกต่างกัน (1 Random Seed ต่อ 1 รอบการทดลอง) เพื่อความน่าเชื่อถือทางสถิติ ในส่วนของการเตรียมข้อมูล ผู้วิจัยทำการแปลงข้อมูลคุณลักษณะให้อยู่ในรูปแบบเชิงตัวเลขด้วยเทคนิค Binary Encoding [8] จากนั้นจึงปรับมาตรฐานข้อมูลเชิงตัวเลขทั้งหมดด้วยวิธี StandardScaler เพื่อให้ข้อมูลมีค่าเฉลี่ยเท่ากับ 0 และส่วนเบี่ยงเบนมาตรฐานเท่ากับ 1 ซึ่งช่วยลดความแตกต่างของช่วงค่าข้อมูลและเพิ่มประสิทธิภาพในการเรียนรู้ของโมเดล [7,9] ทั้งนี้ ผู้วิจัยได้ป้องกันการรั่วไหลของข้อมูล (Data Leakage) ตามคำแนะนำของผู้ทรงคุณวุฒิ โดยการคำนวณค่าเฉลี่ย (μ) และส่วนเบี่ยงเบนมาตรฐาน (σ) เพื่อปรับมาตรฐานข้อมูลจากเฉพาะข้อมูลในชุดฝึกสอนเท่านั้น แล้วจึงนำพารามิเตอร์ดังกล่าวไปประยุกต์ใช้กับชุดข้อมูลตรวจสอบและชุดทดสอบในภายหลังดังสมการที่ (1)

$$z = \frac{x - \mu}{\sigma} \quad (1)$$

โดยที่ z คือค่าข้อมูลหลังผ่านการปรับมาตรฐาน, x คือค่าข้อมูลดิบ, μ คือค่าเฉลี่ยของข้อมูล และ σ คือส่วนเบี่ยงเบนมาตรฐานของข้อมูล

3.3 การสร้าง Champion Embedding ด้วย Autoencoder

งานวิจัยนี้ใช้สถาปัตยกรรมออโตเอนโค้ดเดอร์ (Autoencoder) ซึ่งเป็นโครงข่ายประสาทเทียมแบบไม่มีผู้สอน (Unsupervised Learning) สำหรับการลดมิติข้อมูลและการสกัดคุณลักษณะสำคัญ [3,9] โดยสกัดจากคุณลักษณะตั้งต้นของแชมป์จำนวน 28 คุณลักษณะ ผ่านโครงสร้างส่วนเอนโค้ดเดอร์ (Encoder) ที่ประกอบด้วยชั้นซ่อน (Hidden Layers) ขนาด 16 และ 8 โหนด ตามลำดับ และใช้ฟังก์ชันกระตุ้นแบบ Tanh ในการเรียนรู้ข้อมูล [9] สำหรับพารามิเตอร์ในการฝึกฝน ผู้วิจัยเลือกใช้ตัวหาค่าเหมาะสมที่สุด (Optimizer) แบบ Adam ด้วยอัตราการเรียนรู้ (Learning Rate) 0.001 ฟังก์ชันสูญเสียแบบ Mean Squared Error (MSE) กำหนดขนาด Batch size เท่ากับ 8 และใช้เทคนิคการหยุดฝึกล่วงหน้า (Early Stopping) ที่ค่า Patience 20 รอบ เพื่อป้องกันการเรียนรู้ที่นานเกินไป (Overfitting) ทั้งนี้ เพื่อความน่าเชื่อถือทางสถิติ ผู้วิจัยได้ทำการทดลองซ้ำจำนวน 10 รอบด้วยค่าเริ่มต้นที่แตกต่างกัน (10 Random Seeds) เพื่อเปรียบเทียบขนาดแชมป์เอ็มเบดดิ้งหลายขนาด ได้แก่ 2D, 4D, 6D, 8D, 10D, 12D และ 16D สำหรับใช้ค้นหาขนาดเวกเตอร์ที่เหมาะสมที่สุด โดย

พิจารณาจากความสมดุลระหว่างค่าความสูญเสียในชุดฝึกสอน (Train Loss) ค่าความสูญเสียในชุดตรวจสอบ (Validation Loss) และความสามารถในการทั่วไป (Generalization) ของโมเดล [2,8]

3.4 การพัฒนาและเปรียบเทียบโมเดลพยากรณ์

เวกเตอร์คุณลักษณะ (Feature Vector) สำหรับการพยากรณ์ผลการแข่งขันถูกสร้างจากการนำ Champion Embedding ของแชมป์ทั้ง 10 ตำแหน่งภายในแต่ละแมตช์มาต่อกัน (Concatenate) เพื่อใช้เป็นข้อมูลนำเข้าสำหรับ โมเดลการเรียนรู้ของเครื่อง โดยในกรณีของการเปรียบเทียบกับข้อมูลดิบจะใช้การต่อกันของคุณลักษณะตั้งต้นแทน ผู้วิจัยได้ ศึกษาประสิทธิภาพของโมเดลจำนวน 4 รูปแบบ ได้แก่ Logistic Regression, Random Forest, XGBoost และ Voting Classifier (เทคนิค Soft Voting) [7,8] นอกจากนี้ เพื่อให้เป็นไปตามข้อเสนอแนะของผู้ทรงคุณวุฒิ ผู้วิจัยได้เพิ่มเกณฑ์ เปรียบเทียบมาตรฐาน (Baselines) จำนวน 2 รูปแบบ ได้แก่ Majority Class Baseline และ Random Stratified Baseline เพื่อใช้ประเมินว่าแนวทางที่นำเสนอมีประสิทธิภาพเหนือกว่าการพยากรณ์แบบพื้นฐานเพียงใด

ในการทดลอง ผู้วิจัยแบ่งชุดข้อมูลออกเป็นชุดฝึกสอนร้อยละ 80 และชุดทดสอบร้อยละ 20 สำหรับประเมิน ประสิทธิภาพของโมเดล [4,7] โดยใช้เทคนิค Stratified Sampling เพื่อรักษาความสมดุลของคลาส และทำการทดลองซ้ำ จำนวน 10 รอบด้วยค่าเริ่มต้นที่แตกต่างกัน (10 Random Seeds) เพื่อรายงานผลประสิทธิภาพในรูปแบบค่าเฉลี่ยและส่วน เบี่ยงเบนมาตรฐาน (Mean $\pm$ Std) ซึ่งกระบวนการนี้จะช่วยลดความไม่แน่นอนของผลลัพธ์และเพิ่มความน่าเชื่อถือในการ ประเมินความสามารถในการทั่วไป (Generalization) ของโมเดลพยากรณ์เมื่อนำไปใช้กับข้อมูลในช่วง Draft Phase ที่ไม่เคย พบมาก่อน

3.5 การประเมินประสิทธิภาพและการทดสอบนัยสำคัญ

การประเมินประสิทธิภาพของโมเดลใช้ตัวชี้วัดหลัก ได้แก่ ค่าความถูกต้อง (Accuracy) และพื้นที่ใต้เส้นโค้ง ROC (Area Under the ROC Curve: AUC) เพื่อวัดความสามารถในการจำแนกผลการแข่งขัน [5,8] ดังสมการที่ (2) และ (3)

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

$$AUC = \int_0^1 TPR(FPR)dFPR \quad (3)$$

โดยตัวแปรในสมการมีรายละเอียดดังนี้:

TP (True Positive) คือ จำนวนกรณีที่โมเดลทำนายว่าชนะ และผลการแข่งขันจริงเป็นชนะ

TN (True Negative) คือ จำนวนกรณีที่โมเดลทำนายว่าแพ้ และผลการแข่งขันจริงเป็นแพ้

FP (False Positive) คือ จำนวนกรณีที่โมเดลทำนายว่าชนะ แต่ผลการแข่งขันจริงเป็นแพ้

FN (False Negative) คือ จำนวนกรณีที่โมเดลทำนายว่าแพ้ แต่ผลการแข่งขันจริงเป็นชนะ

TPR (True Positive Rate) คือ อัตราความถูกต้องในการทำนายผลชนะ

FPR (False Positive Rate) คือ อัตราความผิดพลาดในการทำนายผลชนะจากกรณีที่ผลการแข่งขันจริงเป็นแพ้

นอกจากนี้ เพื่อให้เป็นไปตามข้อเสนอแนะของผู้ทรงคุณวุฒิ ผู้วิจัยได้เพิ่มการประเมินด้วยค่าความแม่นยำ (Precision) ค่าความระลึก (Recall) และค่า F1-score เพื่อให้เห็นประสิทธิภาพของโมเดลในทุกมิติ โดยรายงานผลการทดลองทั้งหมดใน รูปแบบค่าเฉลี่ย $\pm$ ส่วนเบี่ยงเบนมาตรฐาน (Mean $\pm$ Std) จากการทดลองซ้ำจำนวน 10 รอบด้วยค่าเริ่มต้นที่แตกต่างกัน ใน ขั้นตอนสุดท้าย ผู้วิจัยได้ดำเนินการทดสอบนัยสำคัญทางสถิติเพื่อเปรียบเทียบความแตกต่างของประสิทธิภาพระหว่างรูปแบบ ข้อมูล 6D Embedding และ Raw Data โดยใช้การทดสอบทีแบบรายคู่ (Paired t-test) และการทดสอบวิลคอกชัน

(Wilcoxon signed-rank test) เพื่อยืนยันว่าผลลัพธ์ที่ได้มีความน่าเชื่อถือทางวิชาการและไม่ได้เกิดขึ้นจากความบังเอิญของชุดข้อมูล

4. ผลการทดลองและการวิเคราะห์ผล

การทดลองในงานวิจัยนี้ได้ปรับปรุงกระบวนการให้เป็นไปตามมาตรฐานความน่าเชื่อถือทางสถิติ โดยทำการทดลองซ้ำจำนวน 10 รอบด้วยค่าเริ่มต้นที่แตกต่างกัน (Random Seeds) เพื่อรายงานผลเป็นค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐาน โดยแบ่งการวิเคราะห์ออกเป็น 3 ส่วนหลัก ดังนี้

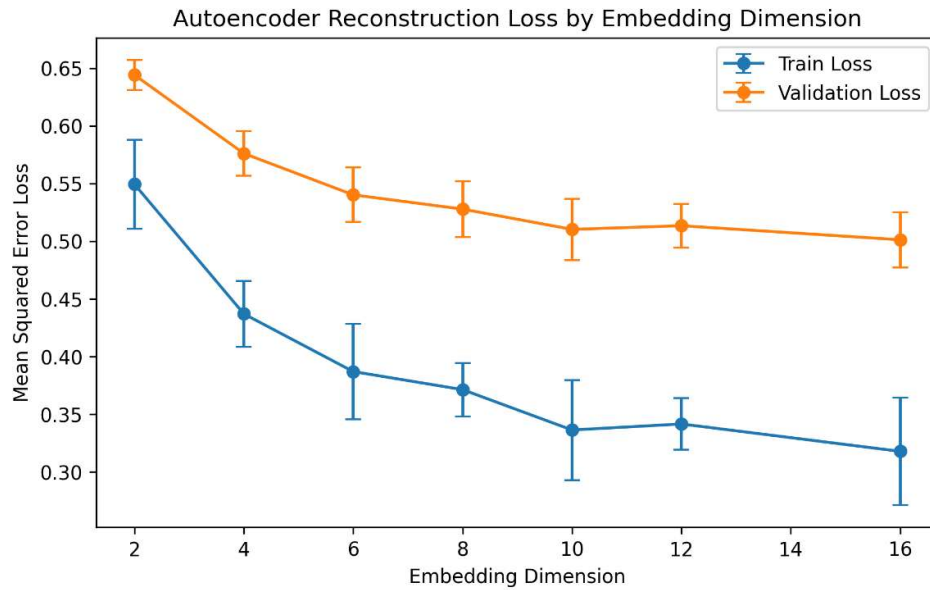
4.1 การวิเคราะห์และเลือกขนาด Champion Embedding (Embedding Dimension Selection)

ผู้วิจัยได้ดำเนินการศึกษาประสิทธิภาพของออโตเอนโค้ดเดอร์ในการสกัดคุณลักษณะและลดมิติข้อมูลแชมเปียนจาก 28 คุณลักษณะดั้งเดิม โดยทำการทดลองเปรียบเทียบขนาดของเอ็มเบดดิ้ง (Embedding Dimension) ตั้งแต่ 2 มิติ ถึง 16 มิติ เพื่อให้ผลการทดลองมีความน่าเชื่อถือทางสถิติ ผู้วิจัยได้ทำการทดลองซ้ำจำนวน 10 รอบด้วยค่าเริ่มต้นที่แตกต่างกัน (Random Seeds) และรายงานผลในรูปแบบค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐาน ($\text{Mean} \pm \text{Std}$) โดยพิจารณาจากค่าความสูญเสียในชุดฝึกสอน (Train Loss) ค่าความสูญเสียในชุดตรวจสอบ (Validation Loss) และส่วนต่างความคลาดเคลื่อน (Generalization Gap) ผลการประเมินประสิทธิภาพแสดงดังตารางที่ 2

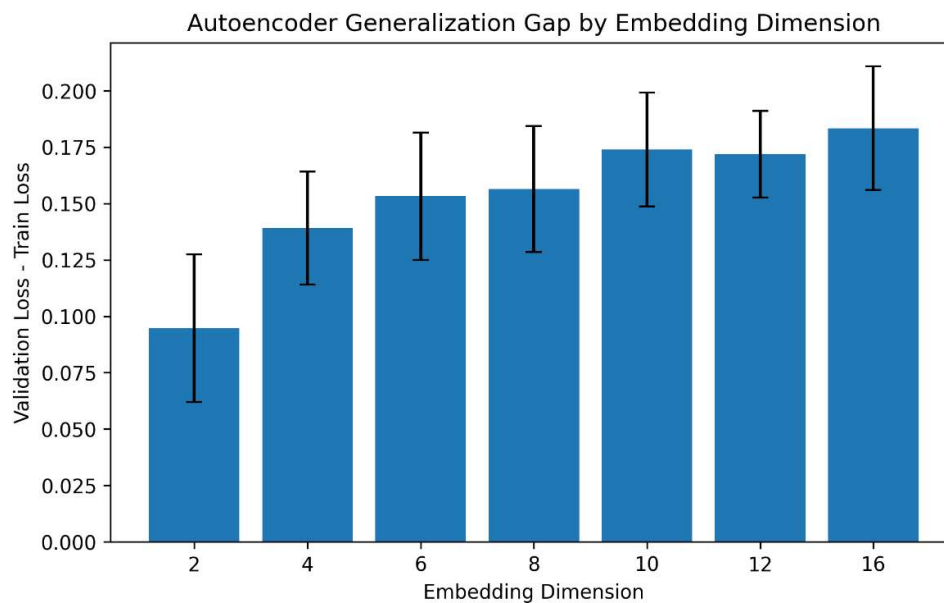
ตารางที่ 2 ผลการประเมินประสิทธิภาพของออโตเอนโค้ดเดอร์ในมิติต่าง ๆ

Embedding Dimension	Train Loss	Validation Loss	Generalization Gap
2D	0.5494 ± 0.0385	0.6442 ± 0.0130	0.0948 ± 0.0327
4D	0.4371 ± 0.0285	0.5761 ± 0.0195	0.1391 ± 0.0250
6D	0.3870 ± 0.0415	0.5402 ± 0.0236	0.1532 ± 0.0282
8D	0.3714 ± 0.0232	0.5279 ± 0.0242	0.1565 ± 0.0279
10D	0.3364 ± 0.0434	0.5103 ± 0.0266	0.1739 ± 0.0252
12D	0.3416 ± 0.0223	0.5135 ± 0.0191	0.1719 ± 0.0193
16D	0.3179 ± 0.0465	0.5013 ± 0.0240	0.1834 ± 0.0274

จากการวิเคราะห์ข้อมูลในตารางที่ 2 พบว่าเมื่อจำนวนมิติเพิ่มขึ้น ค่าความสูญเสียทั้งในชุดฝึกสอนและชุดตรวจสอบมีแนวโน้มลดลงอย่างต่อเนื่อง แสดงให้เห็นว่าโมเดลสามารถเรียนรู้รายละเอียดของคุณลักษณะแชมเปียนได้มากขึ้น ดังแสดงในภาพที่ 1 อย่างไรก็ตาม เมื่อพิจารณาเชิงลึกพบว่าในกรณีของ 2D Embedding มีค่าความสูญเสียในชุดตรวจสอบสูงที่สุด (0.6442 ± 0.0130) ซึ่งบ่งชี้ถึงภาวะการเรียนรู้ไม่เพียงพอ (Underfitting) เนื่องจากมิติข้อมูลที่น้อยเกินไปไม่สามารถรักษาคุณลักษณะสำคัญของแชมเปียนไว้ได้ ในทางตรงกันข้าม 16D Embedding แม้จะให้ค่าความสูญเสียในชุดฝึกสอนต่ำที่สุด แต่กลับมีค่าส่วนต่างความคลาดเคลื่อน (Generalization Gap) สูงที่สุดถึง 0.1834 ± 0.0274 ซึ่งสะท้อนถึงแนวโน้มการเกิดภาวะการเรียนรู้จดจำข้อมูลมากเกินไป (Overfitting) และความเสี่ยงในการนำไปใช้กับข้อมูลใหม่ ดังแสดงในภาพที่ 2



ภาพที่ 1 แนวโน้มค่า Train Loss และ Validation Loss ของ Champion Embedding ในแต่ละมิติ



ภาพที่ 2 ค่าส่วนต่างความคลาดเคลื่อน (Generalization Gap) ของ Autoencoder ในแต่ละขนาด Champion Embedding

ด้วยเหตุนี้ ผู้วิจัยจึงตัดสินใจเลือกใช้ 6D Embedding เป็นเวกเตอร์คุณลักษณะหลักสำหรับการพยากรณ์ผลการแข่งขัน เนื่องจากเป็นจุดที่รักษาสสมดุลระหว่างความสามารถในการเรียนรู้ข้อมูลและความสามารถในการทั่วไป (Generalization) ได้ดีที่สุด โดยมีค่าความคลาดเคลื่อนในระดับที่เหมาะสมและมีความเสถียรสูงจากการทดลองซ้ำหลายครั้ง

4.2 การเปรียบเทียบประสิทธิภาพการพยากรณ์ผลการแข่งขัน

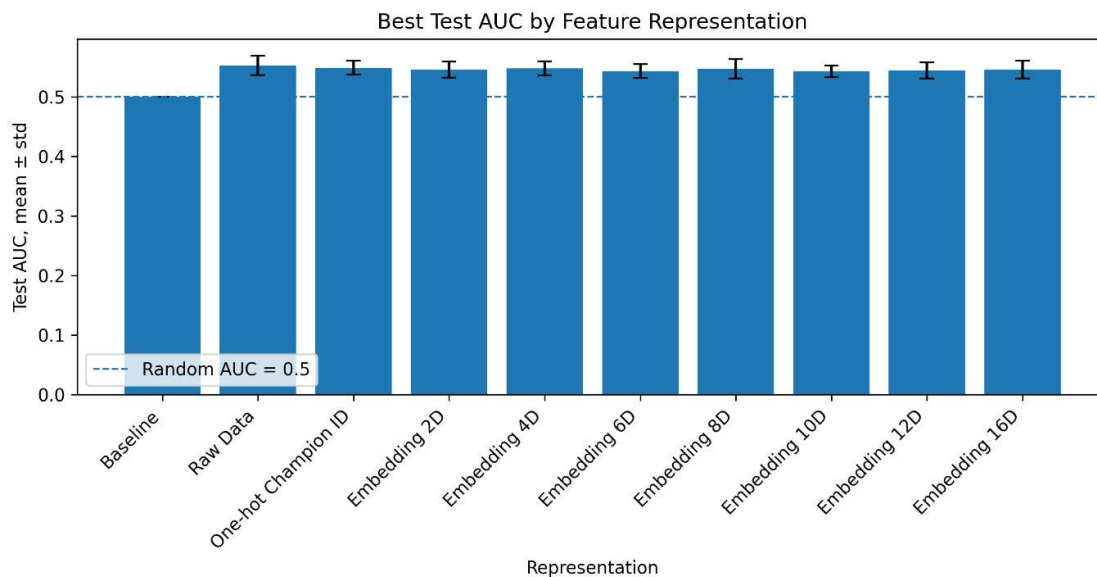
ผู้วิจัยนำคุณลักษณะแชมป์เปี้ยนที่สกัดได้จาก 6D Embedding มาทดสอบประสิทธิภาพในการพยากรณ์ผลการแข่งขัน เปรียบเทียบกับรูปแบบข้อมูลดิบ (Raw Data), ข้อมูล One-hot Champion ID และเกณฑ์มาตรฐาน (Baselines) ผ่านโมเดล

การเรียนรู้ของเครื่อง 4 รูปแบบ เพื่อให้เป็นไปตามมาตรฐานความน่าเชื่อถือทางสถิติ ผู้วิจัยได้ทำการทดลองซ้ำจำนวน 10 รอบด้วยค่าเริ่มต้นที่แตกต่างกัน (Random Seeds) และรายงานผลลัพธ์ในรูปแบบค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐาน (Mean $\pm$ Std) ของค่าพื้นที่ใต้เส้นโค้ง ROC (AUC) บนชุดทดสอบ (Test Set) โดยแสดงรายละเอียดดังตารางที่ 3

ตารางที่ 3 เปรียบเทียบค่า AUC เฉลี่ยของแต่ละรูปแบบข้อมูลและโมเดล (Mean $\pm$ Std) จาก 10 Seeds

รูปแบบข้อมูล	Logistic Regression	Random Forest	XGBoost	Soft Voting
Majority Baseline	-	-	-	0.5000 $\pm$ 0.0000
Random Baseline	-	-	-	0.4945 $\pm$ 0.0155
Raw Data	0.5434 $\pm$ 0.0150	0.5462 $\pm$ 0.0176	0.5480 $\pm$ 0.0110	0.5523 $\pm$ 0.0162
One-hot Champion ID	0.5358 $\pm$ 0.0128	0.5449 $\pm$ 0.0131	0.5486 $\pm$ 0.0118	0.5442 $\pm$ 0.0095
6D Embedding	0.5241 $\pm$ 0.0179	0.5391 $\pm$ 0.0171	0.5429 $\pm$ 0.0116	0.5421 $\pm$ 0.0161

จากตารางที่ 3 พบว่ารูปแบบข้อมูล 6D Embedding เมื่อนำไปใช้งานร่วมกับโมเดลแบบ Soft Voting (Ensemble) ให้ค่า AUC เฉลี่ยอยู่ที่ 0.5421 $\pm$ 0.0161 และค่าความถูกต้อง (Accuracy) เฉลี่ย 0.5329 $\pm$ 0.0123 ซึ่งมีประสิทธิภาพเหนือกว่าเกณฑ์มาตรฐานทั้ง Majority และ Random Baseline อย่างชัดเจน ดังที่แสดงในภาพที่ 3

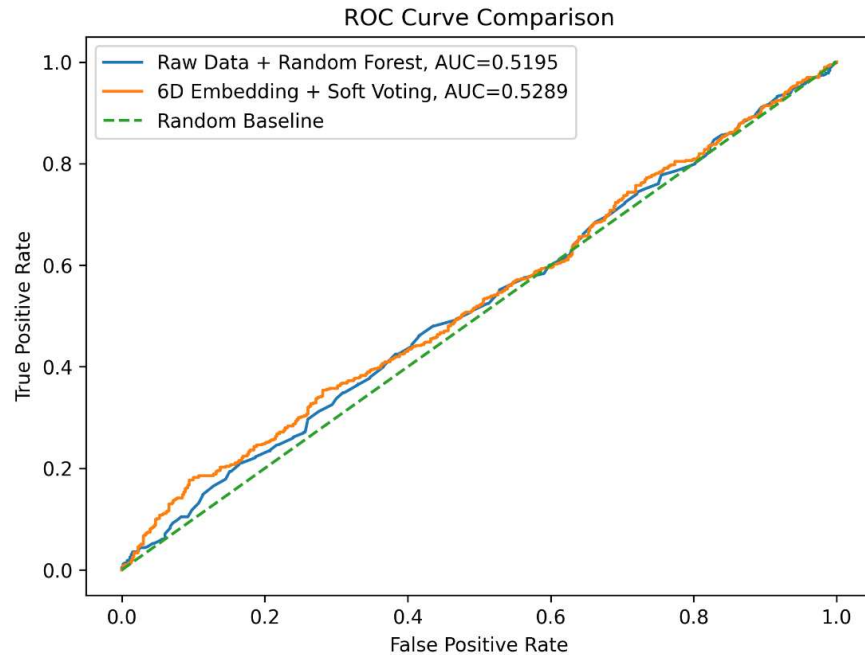


ภาพที่ 3 ค่า Test AUC เฉลี่ยของโมเดลที่ดีที่สุดในแต่ละรูปแบบข้อมูล

แม้ว่าประสิทธิภาพเฉลี่ยของ 6D Embedding จะอยู่ในระดับที่ใกล้เคียงกับข้อมูลดิบ (Raw Data) แต่แนวทางนี้มีข้อดีที่สำคัญคือสามารถลดมิติของข้อมูลลงได้มหาศาลจาก 320 มิติ เหลือเพียง 60 มิติ (ลดลงประมาณร้อยละ 81.25) การลดมิติดังกล่าวมีส่วนช่วยลดความซับซ้อนและสัญญาณรบกวน (Noise) ที่แฝงอยู่ในคุณลักษณะแบบเป็นในช่วง Draft Phase ส่งผลให้โมเดลสามารถเรียนรู้ความสัมพันธ์ขององค์ประกอบภายในทีมได้อย่างมีประสิทธิภาพภายใต้ทรัพยากรข้อมูลที่น้อยลง ซึ่งแสดงให้เห็นว่า Champion Embedding เป็นแนวทางที่มีแนวโน้มช่วยเพิ่มประสิทธิภาพและเสถียรภาพในการพยากรณ์ผลการแข่งขันในบริบทของกีฬาอีสปอร์ตได้

4.3 การทดสอบนัยสำคัญทางสถิติและการวิเคราะห์ตัวชี้วัดเพิ่มเติม

เพื่อให้เป็นไปตามข้อเสนอแนะของผู้ทรงคุณวุฒิ ผู้วิจัยได้ดำเนินการทดสอบนัยสำคัญทางสถิติเพื่อเปรียบเทียบประสิทธิภาพระหว่างรูปแบบข้อมูล 6D Embedding ร่วมกับโมเดล Soft Voting และข้อมูลดิบ (Raw Data) ร่วมกับโมเดล Random Forest ผลการทดสอบด้วย Paired t-test พบค่า p-value เท่ากับ 0.509 ซึ่งมีค่ามากกว่าระดับนัยสำคัญที่กำหนดไว้ 0.05 ผลลัพธ์นี้ระบุว่าประสิทธิภาพในการพยากรณ์ของทั้งสองรูปแบบไม่มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติ การทดสอบนี้ช่วยยืนยันว่าการลดมิติข้อมูลด้วยแคมเปญเอ็มเบดดิ้งจาก 320 คุณลักษณะ เหลือเพียง 60 คุณลักษณะ (ลดลงร้อยละ 81.25) สามารถรักษาความสามารถในการจำแนกผลการแข่งขันไว้ได้ในระดับที่เทียบเท่ากับการใช้ข้อมูลดิบทั้งหมดดังแสดงในภาพที่ 4 ซึ่งเป็นเส้นโค้ง ROC ที่มีลักษณะใกล้เคียงกัน

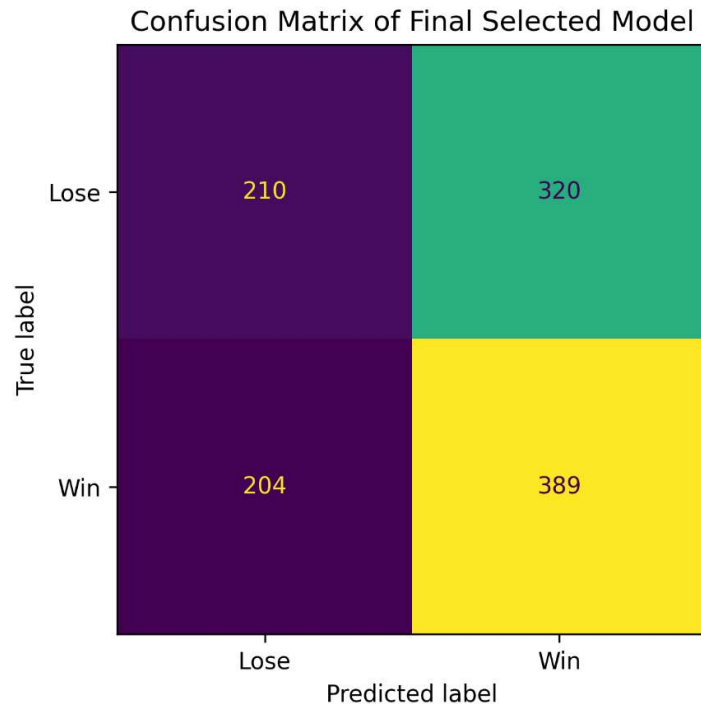


ภาพที่ 4 เส้นโค้ง ROC เปรียบเทียบระหว่างโมเดลที่ใช้ข้อมูลดิบและโมเดลที่ใช้ 6D Champion Embedding

นอกจากนี้ ผู้วิจัยได้วิเคราะห์ประสิทธิภาพผ่านตัวชี้วัดเพิ่มเติม ได้แก่ ค่าความแม่นยำ (Precision) ค่าความระลึก (Recall) และค่า F1-score เพื่อให้เห็นภาพรวมของการทำงานของโมเดลในทุกมิติ โดยแสดงผลลัพธ์เฉลี่ยจากการทดลอง 10 Seeds ดังตารางที่ 4 และจากการแจกแจงรายละเอียดการพยากรณ์ในภาพที่ 5 พบว่าโมเดลที่นำเสนอมีความสามารถในการทำนายผลชนะ (Class 1) ได้แม่นยำกว่าการทำนายผลแพ้ ซึ่งสะท้อนถึงความสามารถในการเรียนรู้องค์ประกอบของทีมที่มีประสิทธิภาพในช่วง Draft Phase แม้จะมีทรัพยากรข้อมูลที่จำกัด

ตารางที่ 4 เปรียบเทียบตัวชี้วัดประสิทธิภาพเชิงลึก (Mean $\pm$ Std)

รูปแบบข้อมูลและโมเดล	AUC	Accuracy	Precision	Recall	F1-score
6D Embedding + Soft Voting	0.5421 $\pm$ 0.0161	0.5329 $\pm$ 0.0123	0.5451 $\pm$ 0.0088	0.6981 $\pm$ 0.0249	0.6121 $\pm$ 0.0135
Raw Data + Random Forest	0.5462 $\pm$ 0.0176	0.5385 $\pm$ 0.0170	0.5508 $\pm$ 0.0122	0.6816 $\pm$ 0.0286	0.6092 $\pm$ 0.0182
Majority Class Baseline	0.5000 $\pm$ 0.0000	0.5280 $\pm$ 0.0000	0.5280 $\pm$ 0.0000	1.0000 $\pm$ 0.0000	0.6911 $\pm$ 0.0000



ภาพที่ 5 เมทริกซ์ความสับสน (Confusion Matrix) ของโมเดลพยากรณ์สุดท้ายในการทำนายผลชนะและแพ้

5. สรุปผลและข้อเสนอแนะ

งานวิจัยนี้ศึกษาแนวทางการใช้ Champion Embedding ผ่านโครงสร้าง Autoencoder เพื่อสกัดคุณลักษณะสำคัญและลดมิติของข้อมูลในช่วง Draft Phase ก่อนนำไปประมวลผลด้วยโมเดลการเรียนรู้ของเครื่อง โดยผลการดำเนินงานและข้อเสนอแนะสามารถสรุปได้ดังนี้

5.1 สรุปผลการวิจัย

จากการทดลองเปรียบเทียบขนาดมิติข้อมูลแฉกเป็นตั้งแต่ 2 ถึง 16 มิติ พบว่า 6D Embedding เป็นขนาดที่เหมาะสมที่สุด (Sweet Spot) เนื่องจากสามารถรักษาสมาดุลระหว่างการเรียนรู้รายละเอียดของแฉกเป็นและความสามารถในการทั่วไป (Generalization) ได้ดีที่สุด โดยช่วยลดความเสี่ยงจากภาวะการเรียนรู้ไม่เพียงพอ (Underfitting) ในมิติต่ำ และภาวะการเรียนรู้จดจำข้อมูลมากเกินไป (Overfitting) ในมิติสูงได้อย่างมีประสิทธิภาพ

ในด้านประสิทธิภาพการพยากรณ์ ผลการทดลองซ้ำจำนวน 10 รอบ (10 Random Seeds) เพื่อความน่าเชื่อถือทางสถิติ ยืนยันว่าการใช้ 6D Embedding ร่วมกับโมเดล Soft Voting ให้ค่า AUC เฉลี่ย 0.5421 ± 0.0161 และค่าความถูกต้อง (Accuracy) เฉลี่ย 0.5329 ± 0.0123 แม้ผลลัพธ์ดังกล่าวจะมีค่าใกล้เคียงกับการใช้ข้อมูลดิบ (Raw Data) ที่ให้ค่า AUC 0.5523 แต่จากการทดสอบนัยสำคัญด้วย Paired t-test พบค่า p-value เท่ากับ 0.509 ซึ่งระบุว่าประสิทธิภาพของทั้งสองรูปแบบไม่มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติที่ระดับ 0.05 ผลลัพธ์นี้พิสูจน์ให้เห็นว่าแนวทางที่นำเสนอนี้มีข้อดีที่สำคัญคือสามารถลดมิติข้อมูลลงได้มหาศาลถึงร้อยละ 81.25 (จาก 320 มิติ เหลือเพียง 60 มิติ) ซึ่งช่วยลดสัญญาณรบกวน (Noise) และความซับซ้อนของข้อมูลในช่วง Draft Phase โดยยังคงรักษาความสามารถในการจำแนกผลการแข่งขันไว้ได้ในระดับเดียวกับการใช้ข้อมูลทั้งหมด

5.2 ข้อเสนอแนะ

เพื่อให้ระบบพยากรณ์ผลการแข่งขันมีความแม่นยำและสามารถนำไปประยุกต์ใช้งานจริงในระดับมืออาชีพได้ดียิ่งขึ้น ผู้วิจัยมีข้อเสนอแนะสำหรับการศึกษาในอนาคตดังนี้ ประการแรก ควรมีการขยายขอบเขตของข้อมูลโดยการผนวกปัจจัยด้านบริบทการแข่งขันเชิงลึกอื่น ๆ เข้ามาเป็นส่วนหนึ่งของเวกเตอร์คุณลักษณะ เช่น สถิติส่วนบุคคลและประสิทธิภาพของผู้เล่น (Player Stats) รูปแบบการประสานงานภายในทีม (Team Synergy) และลำดับการเลือกหรือห้ามเลือกแชมเปียน (Pick/Ban Order) เนื่องจากปัจจัยเหล่านี้มีอิทธิพลต่อผลการแข่งขันนอกเหนือไปจากคุณลักษณะพื้นฐานของแชมเปียนเพียงอย่างเดียว ประการต่อมา ควรพิจารณาประยุกต์ใช้สถาปัตยกรรมโครงข่ายประสาทเทียมรูปแบบอื่น เช่น Graph Neural Networks (GNN) เพื่อจำลองความสัมพันธ์ของแชมเปียนภายในทีมในรูปแบบโครงสร้างกราฟ ซึ่งจะช่วยให้กระบวนการสร้างเอ็มเบดดิ้งสามารถสะท้อนความสัมพันธ์เชิงกลยุทธ์และองค์ประกอบทีมที่ซับซ้อนได้ลึกซึ้งกว่าการใช้โอโตเอนโค้ดเดอร์แบบดั้งเดิม ประการสุดท้าย เนื่องจากเกม League of Legends มีพลวัตสูงและมีการปรับสมดุลของแชมเปียน (Patch Update) อยู่เสมองานวิจัยในอนาคตควรนำข้อมูลการแข่งขันจากฤดูกาลล่าสุดมาทดสอบรวมกับการใช้เทคนิคการแบ่งข้อมูลตามลำดับเวลา (Temporal Split) หรือตามรอบแพตช์ เพื่อให้โมเดลมีความทันสมัยและสามารถรักษาความแม่นยำในการพยากรณ์ได้ภายใต้สภาพแวดล้อมการแข่งขันที่เปลี่ยนแปลงไปในระยะยาว

6. เอกสารอ้างอิง (References)

- [1] Coherent Market Insights. (2026, March 12). *Esports market size, share and opportunities, 2026-2033*. <https://www.coherentmarketinsights.com>
- [2] Jalovaara, P. (2024). *Win probability estimation for strategic decision-making in esports* (Master's thesis). Aalto University.
- [3] Sapienza, A., Goyal, P., & Ferrara, E. (2018). Deep neural networks for optimal team composition. *arXiv preprint arXiv:1805.03284*.
- [4] Inzitari, T., Lyons, B., & Islam, M. N. A. (2022). Predicting pick-ban sequence in League of Legends games. <https://doi.org/10.1145/1122445.1122456>
- [5] Kim, D. H., Lee, C., & Chung, K. S. (2020). A confidence-calibrated MOBA game winner predictor. In *2020 IEEE Conference on Games (CoG)*.
- [6] Błaszczuk, K. W., & Szajerman, D. (2023). Champion recommendation in League of Legends using machine learning. In *The International Conference on Computational Science*.
- [7] Do, T. D., Wang, S. I., Yu, D. S., McMillian, M. G., & McMahan, R. P. (2021). Using machine learning to predict game outcomes based on player-champion experience in League of Legends. In *The 16th International Conference on the Foundations of Digital Games (FDG) 2021*. <https://doi.org/10.1145/3472538.3472579>
- [8] Junior, J. B. S., & Campelo, C. E. C. (2023). League of Legends: Real-time result prediction. *arXiv preprint arXiv:2308.06646*.
- [9] Jang, J., Woo, J. Y., & Kim, H. K. (2022). Action2score: An embedding approach to score player action. *arXiv preprint arXiv:2205.00000*.