

基於時序模型和圖神經網路之 NBA 季後賽勝負預測

Predicting the Outcome of NBA Playoffs Based on Time Series Model and Graph Neural Network

Yen-Tsang Wu Jenq-Haur Wang
Department of Computer Science and
Information Engineering
National Taipei University of Technology
Web Information Retrieval Lab
buddyswu@gmail.com
jhwang@ntut.edu.tw

Ning Chien
Chunghwa Telecom Laboratories Advanced
Technology Laboratory
nickchien@cht.com.tw

摘要

近年來，關於賽事結果預測的研究普遍存在三個問題。第一，許多研究使用比賽結束後的球員統計數據來預測該場比賽的結果；第二，這些研究通常基於球隊的平均表現進行分析和預測。此外，對於球賽數據預測的方式多採用傳統的統計模型，這樣的作法並未考量到數據之間的相關性。以上問題導致了賽事預測的效能偏低。本文提出了一種基於時序模型與圖神經網路的架構，用於預測季後賽球隊的勝負結果。首先，我們將球員作為圖中的節點 (nodes)，並將時序模型預測的球員表現作為節點特徵 (node features)，根據球員在球隊中的位置關係建立邊 (edges)，構建圖 (graph)。其次，利用我們提出的圖神經網路架構進行預測，其中 GAT 的注意力機制 (attention) 用於選取圖中重要的節點並計算節點表達式 (node representation)。隨後，經由 GCN 進行卷積 (convolution) 獲得特徵向量，並通過全連接層 (fully connected layer) 將節點表達轉換為圖表達 (graph representation)，最後進行賽事勝負預測。我們使用了美國職籃 (National Basketball Association, NBA) 2020-2021 球季的數據進行實驗。實驗結果顯示，使用我們提出的方法進行賽事結果預測，準確率達到 76.9%。結果表明，所提出的架構能夠有效預測比賽的勝負。

Abstract

In recent years, most studies on predicting game outcomes face two major issues. The first is the use of player statistics recorded after the game has ended to predict the outcome of that same game. The second issue is analyzing and predicting based on the average performance of the team. This paper proposes a novel approach for predicting game outcomes by integrating time series models and graph neural networks (GNNs). First, player performance predicted by time series models is treated as node features, and edges are constructed based on the players' positional relationships to form a graph. We then introduce a graph neural network architecture for prediction, where the attention mechanism of GAT is used to select important nodes in the graph and compute their representations, while GCN is employed to perform convolution to derive feature vectors. Finally, the node representations are transformed into graph representations for predicting the final game outcomes. The prediction accuracy of the proposed method reaches 76.9%, demonstrating its effectiveness.

關鍵字：選手表現預測、NBA 賽事勝負預測、圖神經網路、機器學習

Keywords: Player Performance Prediction, Graph Neural Networks, Machine Learning,

1 Introduction

隨著運動經濟的蓬勃發展，NBA 已成為全球增長最快的產業之一。然而，現有的比賽勝負預測多依賴賽後數據，忽略了賽前數據的重要性。此外，目前研究主要基於球隊的平均數據進行預測，缺乏對個別球員表現的考

量。因此，現有研究在球隊勝負預測上面臨兩大問題：(1) 賽前對球員表現的預測不足；(2) 缺乏對球員間互動的考慮。為了解決上述問題，我們提出了一種基於深度學習的 Gated Recurrent Unit (GRU) 和圖神經網路 (GNN) 的方法，以提高預測的準確性。我們的研究貢獻如下：

1. 在數據預測方面，我們使用時序模型來預測賽前球員表現，取代傳統的數學運算方式，使賽事勝負預測更貼近真實情況。
2. 本研究考量了所有上場球員的表現，並將圖神經網路應用於賽事預測，預測準確率達到 76.9%，超越了現有的研究方法。

2 Related Work

由於 NBA 賽事擁有豐富的統計數據，因此吸引了大量相關研究。Greene(Greene, 2015)利用綜合數據模型來估算大學新秀的 NBA 選秀順位，這項研究突顯了通過詳細的統計分析和數據建模來增強對 NBA 選秀中球員潛力的理解。Hu 等人(Hu et al., 2019)則透過神經網路預測 NBA 年度最有價值球員 (MVP) 的人選。Sarlis 等人(Sarlis et al., 2021)利用深度學習的方法來評估傷病如何影響球員的個人表現和整體球隊的戰績。Farghaly 等人(Farghaly & Deshpande, 2024) 則使用多種機器學習的方法探討 NBA 球員下肢肌肉拉傷的可能性，並分析了不同因素對受傷風險的影響。現今有許多的研究運用空間和時間的資料，結合深度學習、集成學習及統計方法，對季後賽結果、球員受傷與否和選秀順位進行預測。由於本研究的主題聚焦於 NBA 賽事結果預測，因此相關文獻主要集中在三個方面：籃球比賽結果預測、特徵選取，以及圖神經網路 (GNN) 在運動賽事中的應用。

2.1 Basketball Game Outcome Prediction

賽事結果預測可以視為一種二元分類任務。Hu 等人(Hu & Zidek, 2004)利用數值分析方法，通過統計主場和客場的勝負差異，並使用 weighted likelihood 計算 1996-1997 年冠軍賽的勝負結果。Miljković 等人(Miljković et al., 2010)將比賽數據分為主場和客場，並使用 Naïve Bayes 和 multivariate linear regression 作為分類

模型，預測準確率達到 67%。Cao 等人(Cao, 2012) 使用 2005-2010 賽季作為訓練數據，以 2010-2011 賽季為測試數據，採用 Naïve Bayes、Support Vector Machine (SVM)、Logistic Regression 等模型進行預測，準確率達到 69.67%。

Pai 等人(Pai et al., 2017)提出一個 HSVMDT 的架構，結合 SVM 和 Decision Tree，預測準確率達到 85.2%。Jain 等人(Jain & Kaur, 2017)提出了 Hybrid Fuzzy-SVM (HFSVM) 模型，用於降低數據中 noises 對 SVM 的影響，減少預測誤差，並且使用 CFS (Correlation-based Feature Selection, Hall 1999) 技術進行特徵選取，預測準確率達到 88.26%。Horvat 等人(Horvat et al., 2020)使用多個賽季的數據進行訓練，並採用 Decision Tree、KNN 演算法和 Random Forest(RF)進行實驗，預測準確率達到 60.8%。Osken 等人(Osken & Onay, 2022)使用 K-Means 和 C-Means 聚類演算法來識別球員類型，並基於成員的能力訓練預測模型，他們的方法在 NBA 的五個賽季中實現了 76% 的預測準確率。Wang 等人(Wang, 2023) 則是進行了特徵工程，透過分析和選擇關鍵指標來提高模型預測準確性，研究發現數據中，投籃命中率、三分球命中率和搶斷數是預測比賽結果的關鍵指標。他們使用 Random Forest 和深度神經網路 (DNN) 來進行預測，最佳準確率達到 74%。Adam 等人(Adam et al., 2024) 利用比賽上半場的統計數據，並使用 SVM 來預測最後勝負結果，預測準確率達到 66.67%。

2.2 GNNs Methodology Used in Sports

圖神經網路 (Graph Neural Networks, GNNs) 是一種用於處理圖結構數據的神經網路模型，通過利用圖中節點 (nodes) 與邊 (edges) 之間的關聯性和相互依賴性進行信息傳遞與學習。由於體育比賽中的數據是高度多樣化的，包括時間序列、空間位置以及球員之間的互動。GNN 能夠將這些多模態數據有效地整合，處理體育運動中的圖結構數據，捕捉球員、球隊和比賽之間的複雜關聯，從而提供更準確和有效的預測與分析。

Xenopoulos 等人(Xenopoulos & Silva, 2021) 使用 GNN 預測美式足球 (NFL) 和電子競技遊戲 (CSGO) 的比賽結果，分別將損失 (LOSS) 減少了 9% 和 20%。這表明 GNN 在

處理這類高互動性和多維數據方面具有顯著優勢。Zhao 等人(Zhao et al., 2023) 的研究結合了 GCN 與 Random Forest 算法，以提升對 NBA 比賽結果的預測準確性，預測準確率達到了 71.54%。Luo 等人 (Luo & Krishnamurthy, 2023) 則提出了一種名為 GATv2-TCN 的深度學習方法，將圖注意力網路 (Graph Attention Networks, GAT) 與時間卷積網路 (Temporal Convolutional Networks, TCN) 結合，用於預測運動表現。他們的研究探討了如何通過這些模型捕捉時間序列與圖結構之間的動態關聯，從而提高預測精度。

2.3 Prediction Based on Feature Selection

由於 NBA 數據屬性繁多，特徵選取 (feature selection) 在比賽勝負預測中至關重要。Thabtah 等人(Thabtah et al., 2019b)除了傳統數據以外，更加入了主場 (home) 和客場 (road) 因素，分別使用 Multiple Regression(Berger, 2003)、CFS(Hall, 1999)和 RIPPER algorithm(Cohen, 1995)等方法挑選特徵。研究將這些方法選取的特徵分成五組進行實驗，發現使用 RIPPER algorithm 與 Naïve Bayes 結合的模型表現最佳，準確率達 83%。另外，作者在實驗中還發現這五組特徵中都有一個共同的特徵，即「DRB」(Defensive Rebounds, 防守籃板)，因此推論這是對 NBA 賽事勝負預測中最重要的一項特徵。以上研究都表明，特徵選取在提升比賽預測效能中扮演了至關重要的角色，對比賽結果預測有顯著影響。

3 Methodology

本研究所提出的方法主要分為兩個部分：第一是**球員表現預測 (Player Performance Prediction)**：由於我們無法在比賽開始前就獲得球員的表現數據，因此，我們需要透過球員過去的表現預測出該場比賽的球員數據。第二是**季後賽勝負預測 (Game Outcomes Prediction)**：在這一部分，我們使用圖神經網路 (GNN) 來進行勝負預測。如前面章節所提，輸入的特徵至為關鍵，因此在進行預測前，通過特徵選取 (feature selection) 來找出球員表現中對比賽結果最具影響的關鍵特徵。這些特徵將被用作 GNN 的輸入，進行賽事結果的預測。系統流程如圖 1 所示。



圖 1. 系統流程圖

我們的方法主要分為以下三個步驟：球員表現預測、特徵選取和比賽勝負預測。

3.1 Player Performance Prediction Model

在比賽尚未開打前，我們並無法得知球員的表現，因此，對球員表現的準確預測成為預測比賽結果的關鍵因素。透過觀察，我們可以發現球員的表現往往具有維持一段時間的趨勢，所以我們將預測球員表現當作時序特徵預測任務。

首先，我們將所有數據進行標準化 (normalization)，接著使用基於時序數據的 many-to-one GRU(Dey & Salem, 2017)來預測球員表現。該模型以多場比賽的數據作為輸入，輸出預測單一特徵的值，用來推測下一場比賽中每位球員的表現。GRU 模型的 input 為球員過去的表現數據，輸入的數據必須轉換成 (n, m, k) 的形式才能進行訓練，其中 n 代表輸入的比賽場次(batch size)， m 為 sliding window 的大小， k 為球員表現的特徵數量，將轉換好的球員數據輸入模型進行訓練，最後經過一個線性層 (Linear Layer) 將模型的輸出層向量轉換成實數空間，進行預測。最終，我們可以得到第 $n+1$ 場比賽中每位球員的表現預測。由於總共有 21 項不同的球員表現特徵，因此我們需要執行 21 次預測來完成一場比賽完整的球員數據預測。提出的整體架構如圖 2 所示。

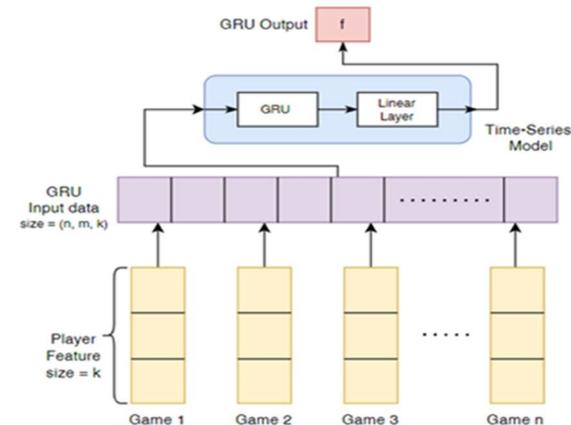


圖 2. 選手表現預測架構圖

我們每次只預測一種特徵，並使用 Sliding Window 的方式，將前 n 場比賽當作訓練來預測第 $n+1$ 場比賽中球員的表現。圖 3 為利用前 3 場 (即 g_1, g_2, g_3) 的球員表現預測第 4 場 (即

g_4)的球員表現的例子。其中 g_i 代表第 i 場比賽，而 $i \in [1, n]$ 。



圖 3 sliding window 示意圖

我們使用 Mean-Square Error(MSE)來評估時序模型，即預測值與實際值之間距離的平方和。

3.2 Feature Selection

NBA 官方公布了豐富的球員數據，這些數據被分為傳統數據和進階數據。傳統數據是球員在比賽中的真實表現，例如得分、籃板、助攻等具體統計項目，這些數據可以直接反映球員在場上的實際表現，因此能夠被時序模型用來進行預測。進階數據則是根據傳統數據進行進一步計算和推導而得出，例如使用命中率、效率值等複合指標來評估球員的表現。本研究採用傳統數據進行實驗，因為它直接反映了球員的比賽表現，詳細說明如表 1。

英文縮寫	中文名稱	英文縮寫	中文名稱
MP	上場時間	TS%	真實命中率
FG	命中球數	eFG%	有效命中率
FGA	總出手球數	FTr	罰球製造率
FG%	命中率	ORB%	進攻籃板率
TP	三分進球數	DRB%	防守籃板率
TPA	三分總數	TRB%	總籃板率
TP%	三分命中率	AST%	助攻率
FT	罰球進球數	STL%	抄截率
FTA	罰球總數	BLK%	阻攻率
FT%	罰球命中率	TOV%	失誤率
ORB	進攻籃板	USG%	球權佔有率
DRB	防守籃板	ORtg	進攻率
TRB	總籃板	DRtg	防守率
AST	助攻	TOV	失誤
STL	抄截	PF	犯規次數
BLK	阻攻	PTS	得分
+/-	場上效率值		

表 1. 數據列表

球員特徵的選取將在數據預測完成後進行篩選，目的是確認每項球員特徵對於賽事勝負預測的重要性。本文使用 SVM 來進行特徵選取。

3.3 Playoff Outcomes Prediction

籃球是一種團隊運動，所有球員彼此相互影響，因此，為了找出影響勝負的關鍵球員以及球員之間的互動影響，本文提出圖注意力卷積網路(Graph Attention Convolution Neural Network, GATCN)，一種結合 GAT(Veličković et al., 2017b)和 GCN(Li et al., 2018)的新架構以進行賽事預測，如圖 4 所示：

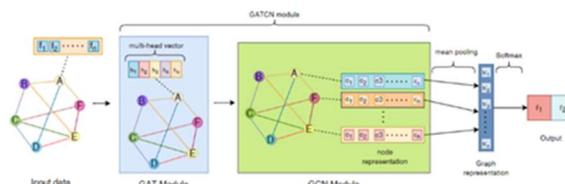


圖 4. GATCN 架構圖

我們首先將球員表現轉換成圖結構形式，並將其作為 GATCN 模型的 input。接著，通過 GAT 計算每個節點（球員）與其 1-hop 鄰居之間的特徵向量。GAT 能夠根據節點之間的重要性分配注意力係數，並使用這些係數來更新每個節點的特徵向量，然後，透過 GCN 將節點之間的區域連結和全域連結性進行聚合，以得出新的節點表達式。接著，通過 mean pooling，將更新後的節點表達式轉換成整體的圖表達式，最後，經由 softmax 函數，我們得到最終 output，即該場比賽的勝負預測。

3.3.1 Player-to-Player Graph

為了能夠精準計算每位球員對比賽的影響值，我們提出了一種 Player-to-Player Graph 的建圖方式。我們將每場比賽中上場的所有球員作為節點，根據他們之間的關聯性來構建出一張圖。這種建圖方式能夠將球員的互動特徵轉換為圖結構，我們更能夠有效捕捉球員之間的複雜關聯性和互動模式，反映球員的團隊合作以及個別球員在比賽中的影響力。首先，給定一張圖 $G=(V,E)$ ， G 代表一場比賽， V 為球員 p 集合，表示為 $V = \{p_1, p_2, \dots, p_n\}$ ， $n \in \mathbb{R}$ ，其中 n 為一支球隊有上場的球員數量，邊 $E = \{e_1, e_2, \dots, e_n\}$ ， $n \in \mathbb{R}$ 則為球員跟球員間的關係，由於 NBA 官方並未公布每位先發和替補球員的上場時間，因此本研究會以同隊的球員建邊形成全連接圖，如圖 5 所示。

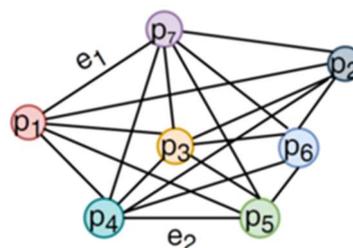


圖 5. Player-to-Player graph 示意圖

球員之間的邊會以相鄰矩陣 A 表示，對於每個相鄰矩陣元素 A_{ij} 代表球員 p_i 和 p_j 之間的邊，其中 $i, j \in \mathbb{R}$ 。我們將同隊的球員之間建邊，如公式(1)所示。若 p_i 和 p_j 是同隊的關係，則相鄰矩陣 A_{ij} 的值为 1，若不是同隊的關係則 A_{ij}

為 0，即 $A_{ij} = A_{ji}$ ，形成無向圖(undirected graph)，節點的特徵 (node feature) 是該球員在該場比賽中的表現數據。

$$f(x) = \begin{cases} 1, & \text{if } (p_i, p_j) \in E \text{ and } i \neq j \\ 0, & \text{others} \end{cases} \quad (1)$$

3.3.2 Graph Attention Convolution Network

我們提出的 GATCN 結合了 GAT 和 GCN 所運用到的技術，GATCN 在 propagation 過程中會運用到自注意力機制(self-attention)，該機制通過計算每個節點與其鄰居節點之間的注意力係數，來判斷節點的重要性。如公式 2 所示。

$$\alpha_{ij} = \frac{\exp(\text{LeakyReLU}(\bar{a}^T[\mathbf{w}\bar{h}_i \parallel \mathbf{w}\bar{h}_j]))}{\sum_{k \in N_i} \exp(\text{LeakyReLU}(\bar{a}^T[\mathbf{w}\bar{h}_i \parallel \mathbf{w}\bar{h}_k]))} \quad (2)$$

α_{ij} 為節點 (i, j) 的注意力係數， $i, j \in \mathbb{R}$ ，其中 \bar{h}, \bar{a} 的定義如之前所述，以 LeakyReLU(Xu et al., 2015) 當作啟動函數調整權重，我們運用 multi-head self-attention，讓各個 head 能夠關注更多區域和全域的訊息，為了將所有 head 得出的特徵向量結合，形成 multi-head，我們透過公式 3 的計算，將所有向量做串聯。

$$h'_i = \parallel_{k=1}^K \sigma \left(\sum_{j \in N_i} \alpha_{ij}^k W^k h_j \right) \quad (3)$$

\parallel 為串聯符號， K 為 head 數量， σ 為 sigmoid 啟動函數， α_{ij}^k 為節點 (i, j) 在第 k 個 head 的注意力係數， W^k 為第 k 個 head 的線性轉換權重矩陣， h'_i 為 K 個 head 串連起來的新注意係數，代表各個球員對於一場比賽的權重。Graph 具有區域連接(local connected)的特性，我們利用卷積的方式來聚合特徵，由於此階段的特徵已經由 GAT 計算過，因此包含和 1-hop 鄰居之間的注意力權重，接著透過 GCN 聚合所有鄰居和自身節點的特徵訊息，更新後的點特徵向量就具備全域信息。為了進行勝負預測，我們沿用 Monti 等人(Federico Monti & Bronstein, 2019)對於將點特徵轉換成圖特徵的方法，透過 mean pooling 的方式將每個 batch 的點特徵向量轉換成圖特徵向量，如公式(4)所示，

$$x_i = \frac{1}{N_i} \sum_{n=1}^{N_i} h'_i \quad (4)$$

N_i 為節點 i 鄰居節點數量， x_i 為圖特徵向量，代表比賽中所有球員的資訊，最後再經過 softmax 的轉換，會得出一個二維向量，代表的意義為比賽勝利或失敗的機率。在訓練的過程中以交叉熵(cross-entropy)作為損失函數。為避免 over fitting 的狀況發生，我們會計算出每一次訓練過程的 loss 值 H_i ，其公式如(5)所示，

其中 C 為類別數，在此為勝負兩種維度， n 為所有資料的筆數， y_c 為類別做 one-hot encode 後的第 i 筆數據的標籤， $r_{c,i}$ 為第 i 筆資料屬於 c 類的機率

$$H_i = \sum_{c=1}^C \sum_{i=1}^n -y_c \log_2(r_{c,i}) \quad (5)$$

4 Experiment and Results

在這項研究中，我們的目標是準確預測 NBA 季後賽的結果。接下來的章節將詳細說明我們所使用的數據集，實驗程序和結果。

4.1 NBA Dataset

本研究實驗的資料集是從 www.basketball-reference.com 中，通過網路爬蟲抓取 NBA 2020-2021 賽季的公開數據而來。該賽季因為受到新冠肺炎(COVID-19)影響，每支球隊只進行 72 場例行賽，因此例行賽總數共有 1080 場，而季後賽則有 91 場。每場比賽有兩支球隊參與，因此每場比賽會產生出一勝一負的數據，因此總數為 2160 場。資料集包含了每場比賽中球員的傳統數據、進階數據、球隊平均數據、勝負(label)。為了避免重複預測，我們採用主場球員表現做為預測對象，訓練集和驗證集則依照 8:2 的比例切割，詳細數據如表 2 所示：

	訓練集	驗證集	測試集
勝	509	127	53
敗	423	107	38
總數	932	234	91

表 2. 資料集統計表

4.2 Evaluation Metrics

我們以準確率(accuracy)、精確率(precision)、召回率(recall)和 f1-score 進行模型的效能評估。

4.3 Player Parameter

由於我們的模型是基於球員表現進行勝負預測，因此每場參與比賽球員的數量會是預測的重要參數之一，我們透過統計 2016-2019 賽季共 681 場的季後賽上場人數作為參考，我們發現板登球員上場時間平均在 15-20 分鐘之間。根據 NBA 統計顯示，每場比賽只使用 7 個球員的場次最多，達到 280 場，其次是使用 8 名球員的比賽場次，共有 203 場。因此，我們將模型的球員數量設為 7 跟 8 人，進行實驗比較。

4.4 Compare models

為了驗證提出模型的效能，我們使用以下模型作為比較對象：

- **Baseline**：我們基於兩篇論文的方法進行融合，以產生一個基準模型。首先，根據 Jones(Jones, 2016)提出 3-game-average 的方法，預測出球員表現後，再使用 XGBOOST 做特徵選取，計算平均成為隊伍整體數據。接著我們使用 Thabtah (Thabtah et al., 2019a)等人提出的 ANN 模型進行比賽結果預測。
- **GAT**：由 Veličković 等人(Veličković et al., 2017a)所提出的架構，將圖神經網路加上注意力機制，使得模型能夠關注單一個點和其 1-hop 鄰居，並計算出注意力係數更新點特徵向量。
- **GCN**：由 Monti 等人(Federico Monti & Bronstein, 2019)提出一種基於 GCN 的圖分類模型，並運用於真假新聞判斷上。其架構的核心為使用兩層的卷積層與全連接層，將 GCN 輸出的 64 維向量降至 2 維，並透過 softmax 計算真假類別的機率。
- **DGCNN**：由 Zhang 等人 (Zhang et al.)提出的 GCN 改良模型，總共有四層 GCN，透過 sort pooling 的方法將每一層的輸出，藉由節點在圖中的結構角色(structural roles)進行排序，然後將 pooling 完的結果輸入傳統 1-D convolution 進行卷積，最後經過線性層輸出結果。

4.5 Experimental Results

如表 3 所示，我們提出的 GATCN 模型，在各項評估指標上都有優異的表現。

Model	Accuracy	Precision	Recall	F1-score
Baseline	0.593	0.886	0.602	0.717
SVM+GAT	0.736	0.754	0.784	0.769
SVM+GCN	0.736	0.792	0.763	0.777
SVM+DGCNN	0.747	0.773	0.788	0.781
GATCN	0.652	0.769	0.666	0.714
SVM+GATCN	0.769	0.811	0.796	0.803

表 3. 實驗結果總表

當我們使用 sliding window = 3 作為 GRU 的設置對球員的表現進行預測，然後按照每場比賽的 8 位球員建圖。接著，通過使用 SVM 挑選出的 8 個特徵作為 GATCN 的輸入進行訓練時，我們的方法在準確率、召回率和 F1-score

的表現都是最好，分別達到 0.769、0.811 和 0.803。實驗結果證明，經過特徵選取後的 GATCN 模型在預測球隊勝負時有較好的表現。我們提出的模型以 GAT 聚合(aggregate)每個節點的 one-hop 鄰居，透過加權平均更新節點特徵，再使用 GCN 進行 Laplace matrix 運算將所有鄰居的特徵聚合，比起只單純使用 GAT 或 GCN 的模型，GATCN 在預測準確率上分別贏過 GAT 和 GCN 模型 3.3%。這充分證明了 GATCN 能夠有效融合兩者的優點，從而提高模型性能。我們也觀察到，DGCNN 模型在 F1-score 有 0.781 的優異表現。DGCNN 通過多層的 GCN 計算出節點與鄰居的關係，並使用 sort pooling 來對節點特徵先排序再 pooling 的方法，對於提升 GCN 是有效益的，但由於缺少了注意力係數的幫助，在準確率仍低於 GATCN 模型 2%。

5 Analysis and Discussion

在這個章節，我們對球員的特徵選取，時序模型以及其他的研究進行討論與分析。

5.1 Analysis of Feature Selection

在構建球員表現預測模型時，挑選哪些數據作為球員的特徵是影響模型性能的關鍵因素。為了選找出最重要的特徵，我們通過 SVM 計算特徵的重要性來進行特徵選取。

如圖 6 所示，SVM 計算出的特徵值範圍從 -1 到 1，趨近於 1 的特徵表示該特徵與比賽勝利正相關；而趨近於 -1 的特徵則表示該特徵與比賽勝利負相關。根據這些計算結果，我們挑選出出[TRB, STL, PTS, FG%, BLK, FT%, +/-, 3P%, ORB, DRB, FGA, TOV, FG, FTA, 3P, FT]作為一場比賽的特徵。其中 TRB 的數值是最大的，代表總籃板和球隊贏球最為關鍵；而 ORB 的數值是最小的，代表進攻籃板和球隊輸球的關係最相關，而[MP, AST, PF, 3PA]則因數值都趨近於 0，表示無關於比賽勝利和失敗。



圖 6. SVM 特徵分數

在過去的研究中，常見的特徵選取方法包括卡方檢驗 (Chi-Square)、Random Forest 和 XGBoost 等。這些方法通常會通過計算特徵的分數來篩選出對模型最有用的特徵。根據的 Chen 等人(Chen et al., 2021)的研究，本研究選擇特徵分數高於平均值的特徵進行比較，以便挑選出最具影響力的特徵。

除了球員的個人數據之外，建邊的球員的數量也會是重要的參數，正如前述章節所提到，7 到 8 名球員是比賽中最常使用的人數。因此，我們在實驗中選擇 7 和 8 人的球員數量來進行實驗。為了找出最佳的特徵，我們將以上四種特徵選取方法，使用由不同 sliding windows 大小產生的球員數據以及不同的選手數量與 GATCN 結合，驗證其效果。其中 sliding windows 以場數表示，實驗結果如下：

場數	人數	Method	Accuracy	Precision	Recall	F1-score
2	8	Baseline	0.652	0.692	0.692	0.692
2	8	SVM	0.681	0.717	0.730	0.723
2	8	XGBOOST	0.637	0.679	0.692	0.685
2	8	RF	0.582	1.000	0.582	0.736
2	8	Chi-square	0.593	0.962	0.593	0.733
3	8	Baseline	0.695	0.769	0.714	0.740
3	8	SVM	0.769	0.811	0.796	0.803
3	8	XGBOOST	0.758	0.811	0.781	0.796
3	8	RF	0.681	0.660	0.760	0.707
3	8	Chi-square	0.692	0.735	0.735	0.735
4	8	Baseline	0.565	1.000	0.565	0.722
4	8	SVM	0.637	0.679	0.692	0.685
4	8	XGBOOST	0.626	0.660	0.686	0.673
4	8	RF	0.582	1.000	0.582	0.736
4	8	Chi-square	0.593	0.962	0.593	0.733

表 4. Feature selection 之實驗結果

從表 4 可以看出，當使用 8 位球員建圖並且以 3 場比賽作為 sliding windows 的場數時，模型在準確率、精確率、召回率和 F1-score 上的表現優於使用 2 場或 4 場比賽，其中 4 場的效果是最差的。而 XGBoost 和 Random Forest 在 2 場的情況優於 4 場，這表明球員的表現並沒有長期的規律可供 GRU 進行學習。3 場比賽作為輸入是最好的。而以 SVM 作為特徵選取的方法時，模型的準確率、召回率和 F1-score 分別是 0.769、0.796 和 0.803，是所有特徵選取的方法中最好的。而在精確率的表現上，使用 2 場和 4 場作為訓練的選手表現，並通過 Random Forest 進行特徵選取後達到 1。但觀察其結果發現是全部預測比賽勝利，因此造成精確率最高，但準確率卻是最低的 0.582。

在特徵選取方法的部分，使用 SVM 表現都是最好的，而使用 XGBOOST 則都是第二。進一步觀察，兩者分析出的特徵極為相近，其中 TRB, STL, PTS, FG%, BLK, FT%, +/-, 3P% 等 8 項是相同的，因此它們在各項評估標準中

的得分也相對接近。XGBOOST 是透過計算 gain 來評估特徵的重要度，減少模型的 entropy，使得重要信息能夠被放大，得出的特徵重要度更為精準，這是 XGBoost 在特徵選取方面的表現更優異的原因。相比之下，Random Forest 由於每次生成的樹會因為 gini index 的不同會影響樹葉的分裂，導致 Random Forest 的特徵選取在此實驗中的表現不如其他方法。至於 Chi-Square，則更適合用於分析類別變數的相關性，但由於輸球和贏球的特徵在本研究中的差異不明顯，因此無法透過計算期望值的方式有效找出關鍵特徵，所以表現不如其他三種方法。

5.2 Analysis of Time Series Model

我們對著名的時序模型 GRU 和 LSTM 進行效能比較。實驗中，我們將兩個模型的場數參數設為 3，球員參數設為 8，並進行 50 epochs 的訓練，optimizer 則是使用 adam(Diederik, 2014)，Loss function 為 MSE。在兩者預測出球員表現後，我們將這些結果輸入 GATCN 模型，進行比賽勝負的預測。結果如表 5 所示：

Method	Accuracy	Precision	Recall	F1-score
LSTM	0.7033	0.7358	0.7500	0.742
GRU	0.7692	0.8113	0.7963	0.8037

表 5. GRU 和 LSTM 效果比較表

從實驗結果，我們可以發現 GRU 在第 15 epochs 時，所有特徵都已經達到收斂，而 LSTM 的 FT 則在第 20 epochs 才收斂。在使用 SVM 進行特徵選取後的實驗結果中，GRU 在四項評估指標上均優於 LSTM。這結果顯示 GRU 用更新閥取代 LSTM 的輸入閥和遺忘閥的作法，有顯著的提升效果。GRU 的優勢在於其更新閥設計，使得在計算單元狀態時能夠將隱藏層的資訊納入其中，從而傳遞了更多的全局信息。相比之下，LSTM 將隱藏層和單元狀態分開計算，導致它只能傳遞部分信息，因此造成了準確率和效能上的差異。此外，由於 GRU 模型結構更精簡，模型訓練的時間比 LSTM 縮短了 2.5%。

5.3 Analysis of GATCN

為了驗證我們所提之 GATCN 的效能，我們比較 Jain 等人(Jain & Kaur, 2017)和 Zhao (Zhao et al., 2023)等人提出的模型。由於 Jain 等人提出 HFSVM 的模型使用 NBA 2015-2016 賽季數據進行實驗，因此我們也使用相同的賽季數據

來測試我們的模型。實驗結果，GATCN 的準確率達到了 0.9302，顯著高於 HFSVM 的準確率為 0.8826。而與我們同樣基於圖神經網路的 Zhao 等人提出的模型，在同樣賽季中的最高準確率為 0.707。這意味著我們的模型對比 Jain 等人的模型準確率提升了 4.8%，而對比 Zhao 等人的模型則提升了 22%。由此可以說明在 NBA 勝負預測任務中，比起透過用團隊數據的模型相比，球員表現所提供的資訊能夠更精準的預測比賽的勝負。實驗準確率如圖 7 所示。

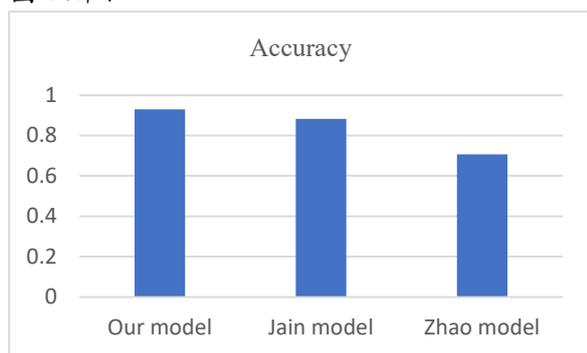


圖 7. 三個模型在 2015-16 賽季準確率比較圖

6 Conclusion and future

在本研究中，我們提出一個針對 NBA 季後賽的比賽結果預測方法。首先透過 GRU 預測選手未來的表現，並把所有球員當作 node，建立一個全連接圖。接著，通過 SVM 進行特徵選取，最後我們運用本文提出的圖注意力卷積網路（GATCN）進行預測。相較於僅使用球員表現數據進行預測，我們的方法更多了球員和球員之間的結構特徵，使得模型在賽事勝負預測中更加精確。

實驗結果顯示，本研究所提出的 GATCN 在準確率、召回率和 F1-score 達到 0.769、0.796 和 0.803，都超越了其他 state-of-the-art 的方法。

Acknowledgments

The authors would like to thank the supports from the National Science and Technology Council, Taiwan under the grant numbers: NSTC113-2221-E-027-096, and NSTC113-2634-F-027-001-MBK.

References

Adam, C., Pantatosakis, P., & Tsagris, M. (2024). On predicting an NBA game outcome from half-time statistics.

Berger, D. E. J. U. C. G. U. (2003). Introduction to multiple regression.

Cao, C. (2012). Sports data mining technology used in basketball outcome prediction.

Chen, W.-J., Jhou, M.-J., Lee, T.-S., & Lu, C.-J. J. E. (2021). Hybrid basketball game outcome prediction model by integrating data mining methods for the national basketball association. 23(4), 477.

Cohen, W. W. (1995). Fast effective rule induction. In *Machine learning proceedings 1995* (pp. 115-123). Elsevier.

Dey, R., & Salem, F. M. (2017). Gate-variants of gated recurrent unit (GRU) neural networks. 2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS),

Diederik, P. K. J. (2014). Adam: A method for stochastic optimization.

Farghaly, O., & Deshpande, P. (2024). Leveraging Machine Learning to Predict National Basketball Association Player Injuries. 2024 IEEE International Workshop on Sport, Technology and Research (STAR),

Federico Monti, F. F., Davide Eynard, Damon Mannion, & Bronstein, M. M. (2019). Fake News Detection on Social Media using Geometric Deep Learning. *ICLR*.

Greene, A. C. (2015). The Success of NBA Draft Picks: Can College Careers Predict NBA Winners?

Hall, M. A. (1999). *Correlation-based feature selection for machine learning* The University of Waikato].

Horvat, T., Havaš, L., & Srpač, D. J. S. (2020). The impact of selecting a validation method in machine learning on predicting basketball game outcomes. 12(3), 431.

Hu, F., & Zidek, J. V. J. L. N.-M. S. (2004). Forecasting NBA basketball playoff outcomes using the weighted likelihood. 385-395.

Hu, J., Zhang, H., & Qiu, J. (2019). Prediction of MVP attribution in NBA regular match based on BP neural network model. Proceedings of the 2019 international conference on artificial intelligence and advanced manufacturing,

Jain, S., & Kaur, H. (2017). Machine learning approaches to predict basketball game outcome. 2017 3rd international conference on advances in computing, communication & automation (ICACCA)(Fall),

Jones, E. S. (2016). Predicting Outcomes of NBA Basketball Games.

Li, Q., Han, Z., & Wu, X.-M. (2018). Deeper insights into graph convolutional networks for semi-supervised learning. Proceedings of the AAAI conference on artificial intelligence,

Luo, R., & Krishnamurthy, V. J. a. p. a. (2023). Who You Play Affects How You Play: Predicting Sports Performance Using Graph Attention Networks With Temporal Convolution.

- Miljković, D., Gajić, L., Kovačević, A., & Konjović, Z. (2010). The use of data mining for basketball matches outcomes prediction. *IEEE 8th international symposium on intelligent systems and informatics*,
- Osken, C., & Onay, C. J. H. (2022). Predicting the winning team in basketball: A novel approach. *8(12)*.
- Pai, P.-F., ChangLiao, L.-H., Lin, K.-P. J. N. C., & Applications. (2017). Analyzing basketball games by a support vector machines with decision tree model. *28*, 4159-4167.
- Sarlis, V., Chatziilias, V., Tjortjis, C., & Mandalidis, D. J. I. S. (2021). A data science approach analysing the impact of injuries on basketball player and team performance. *99*, 101750.
- Thabtah, F., Zhang, L., & Abdelhamid, N. (2019a). NBA Game Result Prediction Using Feature Analysis and Machine Learning. *Annals of Data Science*, *6(1)*, 103-116.
- Thabtah, F., Zhang, L., & Abdelhamid, N. J. A. o. D. S. (2019b). NBA game result prediction using feature analysis and machine learning. *6(1)*, 103-116.
- Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2017a). Graph attention networks. *ICLR*
- arXiv preprint arXiv:1710.10903*.
- Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. J. a. p. a. (2017b). Graph attention networks.
- Wang, J. (2023). Predictive Analysis of NBA Game Outcomes through Machine Learning. *Proceedings of the 6th International Conference on Machine Learning and Machine Intelligence*,
- Xenopoulos, P., & Silva, C. (2021). Graph neural networks to predict sports outcomes. *2021 IEEE International Conference on Big Data (Big Data)*,
- Xu, B., Wang, N., Chen, T., & Li, M. (2015). Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*.
- Zhang, M., Cui, Z., Neumann, M., & Chen, Y. (2018). An end-to-end deep learning architecture for graph classification. *Proceedings of the AAAI conference on artificial intelligence*,
- Zhao, K., Du, C., & Tan, G. J. E. (2023). Enhancing basketball game outcome prediction through fused graph convolutional networks and random forest algorithm. *25(5)*, 765.