



國立中山大學資訊工程學系

碩士論文

Department of Computer Science and Engineering

National Sun Yat-sen University

Master Thesis

一個有效的動態加權集成式演算法應用於異常用電偵測

An Effective Ensemble Algorithm with Dynamic Weight for
Electricity Anomaly Detection

研究生：呂季澤

Chi-Tse Lu

指導教授：蔡崇煒 博士

Dr. Chun-Wei Tsai

中華民國 111 年 7 月

July 2022



©中華民國 111 年 7 月

呂季澤

All Rights Reserved

學年度： 110

學期： 2

校院： 國立中山大學

系所： 資訊工程學系

論文名稱(中)： 一個有效的動態加權集成式演算法應用於異常用電偵測

論文名稱(英)： An Effective Ensemble Algorithm with Dynamic Weight for Electricity Anomaly Detection

學位類別： 碩士

語文別： 中文

學號： M093040039

提要開放使用： 否

頁數： 50

研究生(中)姓： 呂

研究生(中)名： 季澤

研究生(英)姓： Lu

研究生(英)名： Chi-Tse

指導教授(中)姓名： 蔡崇煒

指導教授(英)姓名： Chun-Wei Tsai

關鍵字(中)： 集成式學習、異常偵測、超參數最佳化、超啓發式演算法。

關鍵字(英)： Ensemble learning, anomaly detection, hyperparameter optimization, metaheuristic algorithm.

摘要

在機器學習 (machine learning) 中常常針對不同類型的資料採用不同的分類方法，但即使是相同的資料集內的不同資料，有可能需要使用不同擷取特徵方式才便於使模型學習並分類。集成式學習 (ensemble learning) 主要目標為透過不同種類的方法合併模型預測結果，進而以多個模型取得比單一模型更好的預測結果。傳統的集成學習大多採用多數決 (majority rule) 或者固定公式調整權重並整合各模型的預測結果，這樣的方式在面對多變或不同資料集時效果往往較為有限。為進一步提升此問題，本論文提出一種有效動態的模型結果權重調整方法，將堆疊法 (stacking) 架構中的元學習器 (meta-learner) 以深度神經網路 (deep neural network; DNN) 取代，該網路將學習面對子模型給出的預測機率時，所應該分配的模型權重，對系統整體的預測效能有所提升。本論文也使用超啓發式演算法於最佳化深度神經網路的架構與訓練超參數，除了取代以往時間與人力成本較高的試誤法，也能使模型在最佳化後能有更好的表現，而傳統的超啓發式演算法在搜尋後期往往會因搜尋策略因為群體資訊的影響而困於區域最佳解內，因此本論文利用搜尋經濟學演算法具有劃分解空間及將許久未進入該區域搜尋解作為搜尋解方向考量特性，使模型的超參數搜尋後期能有機率跳出區域最佳解，取得相較於其他超啓發式演算法更優秀的超參數組合。本論文所提出的集成式分類系統將應用於智慧電網中的異常用電偵測，為了評估本論文提出之演算法效能，將實驗結果大致拆分為三部分進行比較，第一部分為透過搜尋經濟學演算法最佳化後的集成式分類系統與目前其他近期使用相同資料集的模型進行比較，第二部分將分析集成式架構中不同的子模型組合、前處理方式及資料集比例對於實驗結果的效能影響，第三部分將實作其他傳統的超啓發式演算法同樣對權重調整網路進行最佳化，並將其結果與搜尋經濟學演算法進行比較且分析。實驗結果顯示，不論是動態權重調整的集成式架構或以搜尋經濟學演算法對超參數進行最佳化，均能有效地提升異常用電偵測的準確率且優於其他近期相似研究。

關鍵詞：集成式學習、異常偵測、超參數最佳化、超啓發式演算法

目錄

摘要	ii
圖目錄	v
表目錄	vii
符號說明	1
第一章 簡介	1
1.1 研究動機	2
1.2 論文貢獻	3
1.3 論文架構	4
第二章 相關文獻探討	6
2.1 問題定義	6
2.2 異常用電偵測	7
2.3 集成式學習	8
2.3.1 裝袋演算法	9
2.3.2 提升演算法	10
2.3.3 堆疊演算法	11
2.3.4 動態的投票權重調整	13
2.4 模型超參數最佳化	14
2.4.1 無模型超參數搜尋	15
2.4.2 基於模型的超參數搜尋	16
2.4.3 超啓發式演算法於超參數搜尋	16
2.5 總結	17
第三章 自適應集成式異常偵測系統	19
3.1 演算法概念及架構	19
3.2 集成式分類系統	22

3.2.1	子模型建構與訓練	22
3.2.2	動態權重分配	25
3.3	權重分配最佳化	26
3.3.1	搜尋經濟學演算法	26
3.3.2	編碼及初始化	27
3.3.3	資源分配	28
3.3.4	遠見搜尋	28
3.3.5	市場調查	30
3.3.6	簡例	31
3.4	總結	32
第四章 實驗結果		33
4.1	實驗環境與參數設定	33
4.2	資料集	33
4.2.1	資料清理	34
4.2.2	資料摺疊	35
4.2.3	缺失值通道	36
4.3	實驗結果及分析	36
4.3.1	資料切割比例實驗	39
4.3.2	子模型組合實驗	39
4.3.3	區域比較實驗	41
第五章 結論與未來展望		42
5.1	結論	42
5.2	未來展望	42
參考文獻		44

圖目錄

1.1	寬且深的異常用電偵測架構 [Zheng et al., 2017]	2
1.2	論文架構圖	5
2.1	CNN-RF 架構圖	7
2.2	裝袋法簡例	9
2.3	提升法簡例	11
2.4	堆疊法簡例	12
2.5	各超參數搜尋演算法比較圖	15
3.1	集成式學習概念圖	19
3.2	動態權重調整	20
3.3	系統流程圖	21
3.4	一維卷積神經網路架構圖	22
3.5	長短期記憶架構圖	23
3.6	自注意力機制示例	24
3.7	多頭注意力模型	25
3.8	堆疊架構示意圖	26
3.9	編碼示意圖	28
3.10	分享資訊示例	29
3.11	搜尋經濟學演算法應用於超參數最佳化示例	31
4.1	正常與異常資料比較	34
4.2	使用分位數轉換法進行資料正規化	35
4.3	資料摺疊示意圖	36
4.4	缺失值特徵通道示例	37

4.5 各模型之精準召回曲線下面積	38
4.6 各演算法於超參數最佳化效能比較	39
4.7 不同區域數量與突變率效能差異	41

表目錄

4.1 實驗硬體環境	33
4.2 混淆矩陣	37
4.3 實驗結果比較圖	38
4.4 以 5:5 資料比例切割	40
4.5 以 6:4 資料比例切割	40
4.6 以 8:2 資料比例切割	40
4.7 實驗結果比較圖	41

第一章 簡介

隨著近年的科技發展與人口增加，台灣的用電量需求也隨著電子設備、用電習慣節節高升，而這些巨量的用電需求往往也造成了尖峰時段缺電導致停電的情況，在非核家園及友善環境的願景下，核能與燃煤發電比例在受限的條件下難以對電力供應產生顯著的幫助，在台灣從石化能源轉型至再生能源期間，「節約能源」變成一項重要的議題。根據行政院的智慧電網政策，智慧電網的主要目的有三點：提升再生能源穩定度、提升用戶供電品質與促使用戶節能，透過家庭中裝設的智慧電表，能源供應方能與用電戶達成雙向溝通，使以往不易觀測的用電量數值化並可以用於進行即時用電觀測、未來用電預測與異常用電偵測。

在智慧電網的環境中，智慧電表除了使用電方及供應方可以進行即時觀測用電量外，也可以隨時針對異常的用電量發出提醒或者警告，而異常用電大致可分為兩種 [Antmann, 2009]：技術電力損失 (technical losses; TL) 與非技術電力損失 (non-technical losses; NTL)。技術電力損失主要發生於電錶故障、線路損毀或者於電路中因材質的傳輸損耗，而非技術性電力損失電錶故障、線路損毀或人為的非法行為 [León et al., 2011, Moghaddass and Wang, 2017]。這些異常用電的耗損有可能會造成電力公司的額外隱性成本。另考量用電戶方面，若智慧電表的電力觀測值異常，將導致用電戶每個月必須承受多餘的用電量款項。為了減少不必要電力浪費與成本花費，本論文嘗試提出異常用電分類方法，以利更準確偵測出上述之異常用電情況。近年也有許多學者致力於發展智慧電表的異常用電偵測演算法，如 Zheng 等人 [Zheng et al., 2017] 使用既寬且深的神經網路進行異常用電分類，其中一維資料被輸入多個神經元之全連接層進行訓練，再將一維電力資料摺疊成二維輸入多層的卷積神經網路進行訓練，最後將該網路分類結果進行合併並經過一個 sigmoid 函數完成分類，分類結果中顯示該分類效能優於大部分機器學習與基礎深度學習分類模型效能，該研究也釋出文中所使用之完整資料集，並在後續衍生出許多異常用電分類相關研究。Hasan 等人在另一個研究 [Hasan et al., 2019] 中，結合了卷積神經網路與長短期記憶網路，提出了 CNN-LSTM 分

類模型，模型前期利用卷積神經網路取得資料的空間特徵，並使用長短期記憶網路取得前段網路所擷取特徵之時間特性並通過全連接層 (fully connected layer) 進行異常用電分類，該電力資料中具有嚴重的資料不平衡 (data imbalanced) 問題。

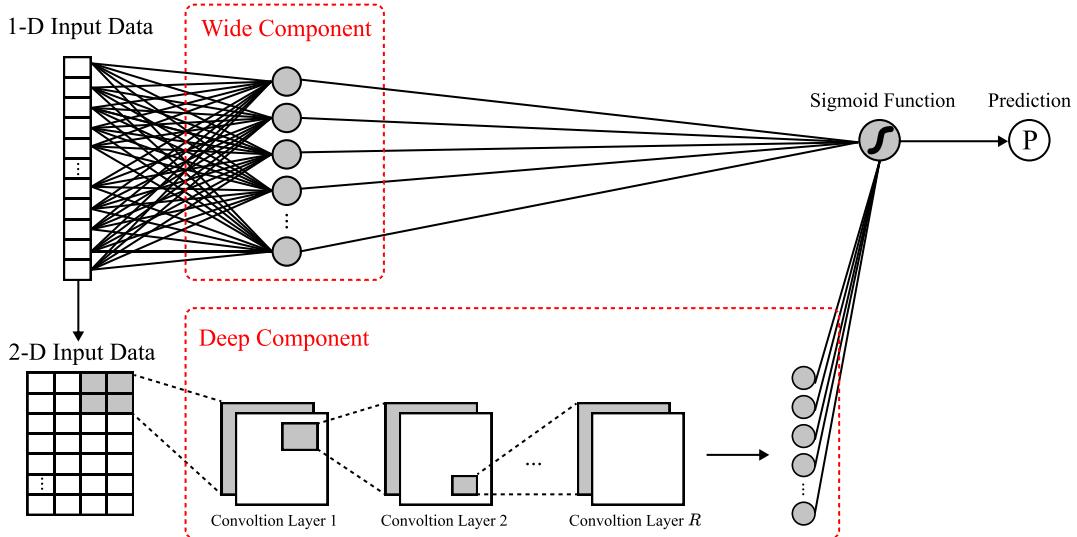


圖 1.1: 寬且深的異常用電偵測架構 [Zheng et al., 2017]

1.1 研究動機

為了提升深度學習分類方法的準確度 (accuracy)，集成學習相關技術在近年來被許多研究所使用。在集成學習的架構中，透過多個弱模型或強模型互相溝通及分享資訊，整合成一個比原先模型更加準確的多模型分類方法，而在本論文所使用的堆疊法中，針對不同特徵類型的模型將會被獨立訓練且用自身對於資料的觀點將預測結果集成至元學習器當中，即使皆為時序性資料 (time-series data)，也可能因為各種不同的長、短期因素導致該筆資料應備判斷為異常資料，倘若僅使用一種類型模型，該模型難以在觀察短期資訊的同時，記憶長期的時序性特徵。因此本論文計畫預計結合擅長記憶長期時序特徵的長短期記憶模型、擷取短期空間特徵的一維卷積神經網路模型及擷取不定時序特徵的多頭注意力機制模型，藉由整合擅長不同特徵的模型，進一步設計更高效的異常用電偵測模型。然而在這些被使用的模型當中，往往會有較適合於判斷該資料集的模型，為解決此問題，本論文在整合各模型的分類結果後，將整合結果一併輸入至深度神經網路 (deep neural network) 進行各模型的投票權重自適應調整，深度神經網路會動態地根據不同的輸入資料，調整各模型在分類結果中的影響性，也因

此堆疊法的訓練將分為兩個階段，第一階段為獨立訓練各模型，第二階段將會訓練該深度神經網路進行投票權重分配。

為了最佳化深度神經網路的投票權重分配能力進而提升異常用電偵測準確率，本論文採用超啓發式演算法 (metaheuristic algorithm) 來調整深度神經網路的架構及其超參數，在傳統超參數最佳化方法中，試誤法 (trial and error) 是最經常使用的方式之一，透過研究人員根據經驗思考並調整超參數嘗試使模型效能往更好的方向前進，但這樣的方法有兩個缺陷，耗費大量的嘗試時間與不一定能嘗試到貼近最佳解的超參數組合。超啓發式演算法使用了類似於群體生物互相分享資訊的概念，在解空間中透過不同的策略來使群體往更佳解的方向前進，其中基因演算法 [Holland, 1992] (genetic algorithm; GA) 啓發於生物的遺傳機制，不同解之間透過交配可以互相交換各自的部分解並產生下一代解，為了使解更加多樣化，生成解時將會有一定的機率突變；粒子群最佳化 [Kennedy and Eberhart, 1995] (particle swarm optimization; PSO) 利用群體智慧，每個解都會參考該迭代中最佳解位於解空間的位置 (population best) 及整個搜尋過程中群體找過的最佳解 (global best)，來引導自身往更佳解的方向前進。然而這些具有群體智慧機制的演算法在落入區域最佳解時，雖然能有一定機率跳離該解空間區域，但面對解的維度過大時效果仍相當有限。Tsai 等人 [Tsai, 2015] 提出了搜尋經濟學演算法，該演算法透過解空間的切割與期望值機制，使群體在搜尋後期依然有機會往回尋找其他解空間的可能性，也因此有機會跳脫原本的區域最佳解，找到隱藏的更加解，根據此搜尋優勢，本論文預期能利用該演算法於深度神經網路的超參數搜尋並有效的提升集成式模型的分類效能。

1.2 論文貢獻

本論文將基於集成式學習中的堆疊法 (stacking) 提出一種針對於異常用電偵測的深度學習系統，並使用搜尋經濟學演算法最佳化集成式學習當中自適應投票權重網路之超參數。集成式模型中的各個子模型均有各自擅長擷取的特徵類型，子模型經過完整訓練後對同一筆時序將有不同的分類機率分布，而透過搜尋經濟學演算法最佳化後的自適應投票權重網路將會依據各自所預測出的類別機率進行學習，並分配適當的權重給各子模型的預測結果。

- 建立一個針對於分類異常用電資料集的集成式模型，組合含有不同特徵觀點的子模型，並運用自適應投票權重網路結合各子模型之分類結果、學習且動態分配投

票權重後，進行最後的判斷，完成兩階段的分類演算法。

- 改良原先的搜尋經濟學演算法，設計新型的超參數解編碼及解空間切割方式，使該演算法能運用區域期望值使其在搜尋後期不只侷限於搜尋以往出現較優秀解的區域，更能回訪其他久未搜尋的解空間區域，並可能於其他區域搜尋到更佳解，跳出原先的區域最佳解，搜尋到比其他超啓發式演算法更好的超參數組合。
- 以竊電資料集為應用對象，透過使用分位數轉換法及建立缺失值特徵解決資料集本身含有的資料不平衡、離群值與缺失值問題，分析不同的資料前處理方式的差異並探討其原因。

1.3 論文架構

本論文的架構圖如圖 1.2，論文自第二章起將分為背景知識與相關研究、方法、實驗結果與討論、結論與未來展望。第二章將會對集成模型中的分類問題進行定義，並介紹集成模型的發展與近年的應用，其中包含了各種集成式學習的經典方法，包含了切分資料集並分開訓練數個相同模型的裝袋法 (bagging)、相同弱模型透過多次分類最後匯集分類結果而成的提升法 (boosting) 及本論文使用之二階段訓練的堆疊法 (stacking)。第三章將使用方法架構圖介紹本論文提出的自適應集成式異常偵測系統，共分為兩部分，第一部分為介紹在集成式模型中各子模型架構及該集成式架構的訓練流程，第二部分將介紹搜尋經濟學演算法的原理及如何將該演算法應用至調整深度神經網路的超參數，並使用簡單的例子演示本論文提出之演算法步驟。第四章將詳細的介紹本論文的實驗環境及各實驗的演算法、模型訓練之參數設定及所使用的資料集。本論文將比較且分析集成式架構中不同子模型與搜尋經濟學演算法中不同參數之效能差異，最後將根據本論文提出方法的最終結果與其他文獻方法進行比較與分析，並於第五章提出結論與未來展望，探討本論文後續的發展空間。

一個有效的自適應集成式演算法應用於異常用電偵測

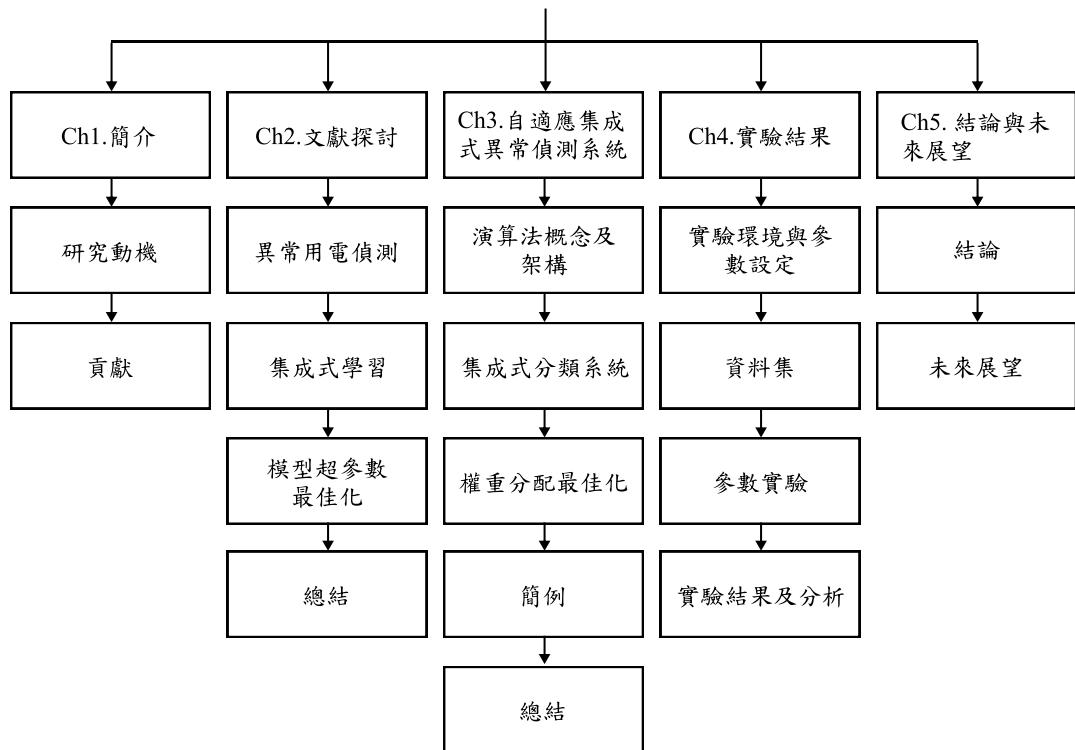


圖 1.2: 論文架構圖

第二章 相關文獻探討

2.1 問題定義

為了比較各方法間的效能差異並分析導致該差異的原因，本論文將異常偵測問題視為一分類問題 [Aziz et al., 2020]，其中自適應投票權重網路的問題定義公式如 公式 2.1 所示：

$$\lambda^* = \arg \max_{\lambda} \text{PRAUC}(\mathcal{A}_\lambda(\mathcal{M}_1^{\theta_1^*}(\mathcal{D}^t), \mathcal{M}_2^{\theta_2^*}(\mathcal{D}^t), \dots, \mathcal{M}_n^{\theta_n^*}(\mathcal{D}^t))), \quad (2.1)$$

其中 PRAUC 為精準召回曲線下面積 (area under curve; AUC)，也是本論文用於評斷分類模型優劣的最主要依據， \mathcal{D} 為完整的時序用電量資料集，本論文切分為訓練集 \mathcal{D}^a 及測試集 \mathcal{D}^t 。因此公式 2.1 可用於表達為：集成學習中的各子模型 \mathcal{M}_1 到 \mathcal{M}_n 透過 公式 2.2 進行模型分類訓練取得最佳的神經元權重 θ^* ，測試集 \mathcal{D}^t 輸入訓練完成的各子模型 $\mathcal{M}_1^{\theta_1^*}$ 至 $\mathcal{M}_n^{\theta_n^*}$ 後，將各模型的分類結果輸入至使用超啓發式演算法找到的超參數組合 λ 進行模型訓練的自適應投票權重網路 \mathcal{A}_λ ，而分類結果中的 PRAUC 越大，代表該分類方法越佳。其中各子模型及自適應投票權重網路訓練的損失函數 (loss function) 定義如 公式 2.2 所示：

$$\theta^* = \arg \min_{\theta} \mathcal{L}(\theta, c, \mathcal{D}^a) = \sum_{j=1}^{|\mathcal{D}^a|} \left(-\mathcal{P}_{\theta}^{x_j}(c_j) + \log \sum_{i=1}^k \exp(\mathcal{P}_{\theta}^{x_j}(i)) \right), \quad (2.2)$$

其中 θ^* 代表能該模型能在該次訓練中找到最低損失值 \mathcal{L} 的模型權重，而每次驗證該權重所得的損失函數值須輸入當前的神經元權重 θ 、當前批次大小 (batch size) 的訓練資料 \mathcal{D}^a 及其真實標籤 c 。損失函式中的 $\mathcal{P}_{\theta}^{x_j}(c_j)$ 代表基於當前的模型權重 θ ，模型判斷當前筆資料 x_j 屬於真實類別 c_j 的機率，並使用公式後半計算模型對於總類別數 k 中每類的機率值與真實類別的機率負數相加，即可得到該模型當前對驗證資料集的損失函數值，而損失函數值越低，代表模型的分類結果越趨近於真實，使模型可以往效能更佳的方向前進。

2.2 異常用電偵測

基於近年的深度學習快速發展，有越來越多的研究 [Mohammed et al., 2021, Xiao et al., 2018, Zhang et al., 2021] 以開發各種不同的深度學習模型為主，並試圖打造能於異常用電偵測問題上取得優秀的表現。而這些研究也顯示了深度學習的方法相較於傳統的機器學習方法，例如支持向量機、隨機森林等，深度學習模型能取得更好的表現。為了能更加的提升異常用電的偵測準確度，許多研究 [Buzau et al., 2019, Livieris et al., 2020] 也開始結合兩種以上的分類器或特徵擷取方式，透過各種深度學習方法間對資料的不同特徵角度，模型可以從不同觀點去審視並思考資料的空間特徵或者時序特徵，其中 Hasan 等人 [Hasan et al., 2019] 於文獻中提出了基於卷積神經網路與長短期記憶模型，該文獻中先透過卷積神經網路擷取空間性特徵，將擷取後的特徵攤平，最後使用長短期記憶網路學習在二維的電力資料中，各資料區塊間的時序性特徵，並輸出結果至最後的神經元層完成分類。實驗結果也顯示相較於傳統的支持向量機及邏輯回歸獲得的 69% 及 79% 準確率，卷積神經網路與長短期記憶組合成的模型以 89% 的準確率大幅度的領先。Li 等人 [Li et al., 2019] 結合卷積神經網路與隨機森林，提出 convolutional neural network-random forests (CNN-RF) 模型，如圖 2.1 所示，該模型使用隨機森林取代了原本卷積神經網路末端的輸出層，隨機森林以全連結層的神經元輸出作為輸入值進行最後分類，實驗中作者也嘗試以相同架構結合卷積神經網路與其他的機器學習方法，如支持向量機、梯度提升決策樹 (gradient boosting decision tree; GBDT)，結果也顯示使用 bagging 機制的隨機森林可以取得比其他機器學習方法更好的分類結果。

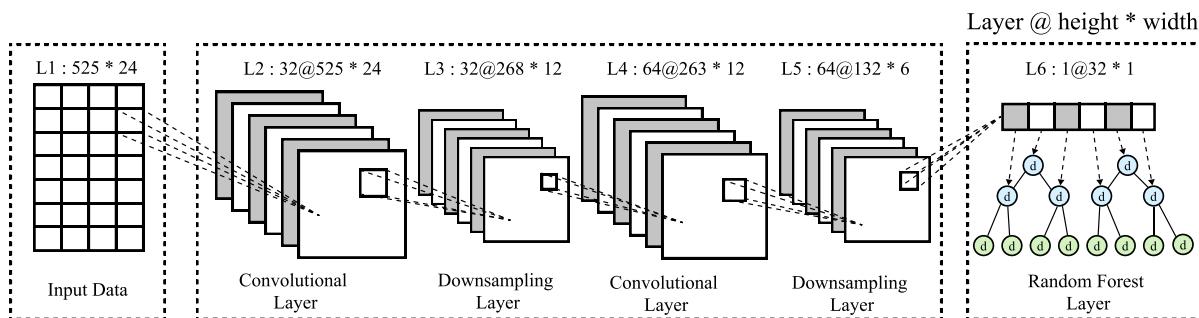


圖 2.1: CNN-RF 架構圖

電力資料集通常為智慧電表以一段時間為單位蒐集而成，這些資料通常以一維資料但長時間的資料型態為主，許多文獻 [Chen et al., 2018, Feng et al., 2020] 也根據此

特性，將一維的電力資料進行資料摺疊，使二維的深度學習方法可以透過擷取二維特徵的方式對資料進行學習，而這些文獻中大多也探討了以週、半月、月分別代表 7 天、14 天、28 天為單位進行資料摺疊，對於分類結果的差異。也因為二維卷積核 (convolution kernel) 必須擷取相鄰值特徵的特性，以不同單位對資料進行摺疊意味著欲使卷積模型觀察每週相似天、每半月相似天或者每月相似天的變化特徵。Chen 等人 [Chen et al., 2018] 提出 convolution-based deep neural network modeling periodic traffic data (PCNN) 方法，該文獻將電力資料的時序特徵分為兩個種類：局部連貫性與週期性。局部連貫性是在發生與以往不同的電力數值時，通常會大幅度的影響接續幾個時間點的電力數值，例如晚上 6 點突然的電力需求可能導致接下來 7 點至 10 點的電力異常需求；週期性則是以每週或每天為週期，各時間點相似的電力固定變化趨勢。該模型將時序電力資料摺疊為二維矩陣形式，並使用進行上述兩種時序特徵的擷取。並輸入至全連接層進行未來的電力預測。該文獻的實驗結果中也顯示，當作者以 10 分鐘、15 分鐘、30 分鐘及 60 分鐘的時間間隔摺疊時序電力資料，會對於預測結果會產生不同的影響，同時也證明了比起不使用摺疊方法處理資料集輸入至機器學習或者深度學習的模型進行預測，使用摺疊後的資料可以取得更好的預測結果。Feng 等人 [Feng et al., 2020] 提出了 TextCNN 模型進行異常用電的分類，該研究分析了使用卷積神經網路、深度神經網路及 TextCNN 的差別，為了解決電力資料的不平衡，該文獻利用了時間窗擷取經過摺疊並複製經過折疊後的時序資料並輸入至主要提出之 TextCNN，與傳統卷積神經網路不同的是，TextCNN 使用的卷積核並非如 CNN 般長跟寬皆固定，卷積核的高將根據資料摺疊後的資料週期而改變，換而言之，該模型可以一次觀察到所有相似的時間點中的時序週期性關係。

2.3 集成式學習

集成式學習最早於研究 [Dasarathy and Sheela, 1979] 中提及與集成學習相似的演算法，文中作者 Dasarathy 與 Sheela 提出了一個初步的集成式系統，該系統將資料的特徵切割並分別使用不同的分類器訓練。1990 年 [Hansen and Salamon, 1990] Hansen 及 Salamon 更提出一個完整的集成式系統，文中也比較一般的分類演算法與集成式的分類方法，實驗結果也表明透過集成式學習可以有效地提升整體的分類準確率。同年 [Schapire, 1990] Schapire 也提出的一個嶄新的集成式學習方法，並於文中驗證了多個弱分類器確實可以達到與強分類器相同的效能。在上述的經典文獻提出後，集成學習在

近 20 年內逐步成長，而本章節將探討不同類型的集成式學習文獻與提出相關評述。近年的集成學習大致可以分為三類，分別為裝袋法 (bootstrap aggregating; bagging)、提升法 (boosting)，與堆疊法 (stacking)，這三種方法透過不同的模型間合作方式提升整體的分類或者預測效能，本章節將這三種相似但不同的集成學習方法分開介紹並探討該方法近年的相關文獻。

2.3.1 裝袋演算法

集成學習中的裝袋概念最早由 Breiman 提出 [Breiman, 1996]，該方法以公民投票的概念設計，通常將訓練集進行切割，分別輸入至各自獨立但具有相同結構的弱分類器進行訓練，各模型皆訓練完成後，將測試集輸入至各模型取得各自的分類結果，而各模型的分類結果即為該模型進行投票的對象，在裝袋法中每個模型的分類結果均具有相同的權重，最後以多數決的方法決定該集成分類系統最重的分類結果輸出。如圖 2.2 所示，將經過前處理後的資料進行切割為資料集 1 至資料集 3，接著獨立的輸入至不同的分類器進行訓練並進行類別預測，最後的結果輸入將進行多數決而統計過後的最終結果即為該集成式系統的分類結果。

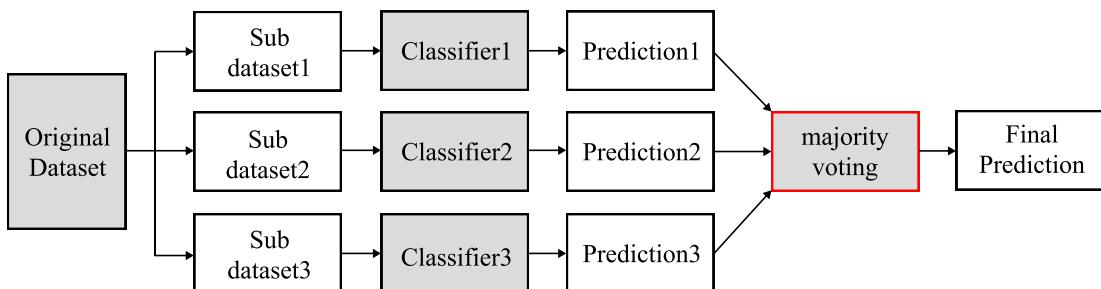


圖 2.2: 裝袋法簡例

在裝袋法提出之後，Breiman 延伸了自己的研究並提出了隨機森林 (random forests) [Breiman, 2001]，該方法結合多個分類與迴歸樹 (classification and regression tree; CART) 進行分類，透過隨機切割資料集用以訓練各獨立的分類與迴歸樹，並透過多數決投票決定最後的結果，而該研究也證明了，比起獨立一個的決策樹 (decision tree)，多棵樹獨立訓練並進行投票分類可以有效的提升整體的分類效能而透過隨機切割資料集分類多棵樹，也可以因為各樹學到資料集中不同組合、不同分布的樣本而避免只有一棵樹時造成的過度擬合問題。隨機森林的成功，使裝袋法也開始被關注與分析 [Bühlmann and Yu, 2002]，越來越多的學者嘗試將不同的機器學習分類方法結合裝袋

法。Kim 等人 [Kim et al., 2002] 嘗試將支持向量機 (support vector machine; SVM) 與裝袋法進行結合並首次提出由模型組合的雙層結構，經過資料集的隨機裝袋與各支持向量機模型的獨立訓練後，將各模型的輸出結果輸入至第二層的支持向量機，而該層的支持向量機模型的學習方向主要為根據第一層模型的輸出，決定本輪的分類結果。Ha 等人 [Ha et al., 2005] 使用了裝袋神經網路 (neural network)、單一神經網路與傳統的邏輯迴歸方法用於預測消費者消費習慣並比較及分析了使用線性參數的傳統邏輯迴歸與非線性的神經網路，實驗結果顯示透過不同獨立子資料集訓練的多層感知器 (multilayer perceptron; MLP) 收斂速度明顯快於其他僅使用單一分類器進行分類的演算法，雖然裝袋多層感知器 (bagged multilayer perceptron; BMLP) 無法在全部的模型中擁有最好的結果，但穩定性明顯比取得最好結果的單一多層感知器 (single multilayer perceptron; SMLP) 好非常多，該篇研究開啟了裝袋法與深度學習領域結合的先例，直到近期，裝袋法可以避免過度擬合的特性除了經常被用於強化異常偵測的任務 [Alvear-Sandoval and Figueiras-Vidal, 2018] 也常常用於分類不平衡的資料集，Wang 等人 [Wang et al., 2017] 提出了 bagging of extrapolation borderline-SMOTE SVM (BEBS)，該方法啟發於 synthetic minority oversampling technique (SMOTE)，先利用多個支持向量機對原始不平衡資料集進行分類並標記出哪些為資料集中的少數，接著使用 SMOTE 生成這些被標記為少數的資料。實驗結果顯示 BEBS 與其他常見的資料擴增方式或者沒有使用資料擴增的方法比較，在 G-means、F1-score、精準度上均能更勝一籌。

2.3.2 提升演算法

提升法 (boosting) 為集成學習中的一大分類，如 圖 2.3 其主要概念為建構多個弱分類器，每個分類器皆從資料集中抽樣進行訓練，每個弱分類器會將自己的分類結果輸入至下一個弱分類器進行訓練，而下個分類器將從上個分類器分類錯誤的資料中抽取一部分作為自己迭代的訓練資料藉此補足上個分類器的缺失並提升整體對於該資料集的分類能力，最後透過每個分類器投票並以多數決方式來決定最終的分類結果。

但在原始的 boosting 版本中雖然每次都從資料集中進行隨機抽樣訓練，還是存在著嚴重的過度擬合問題。adaptive boosting (adaboost) [Freund et al., 1996] 為 Freund 等人提出的最經典 boosting 方法之一，該方法有效的解決該問題，如 演算法 1 所示， D 代表要輸入至 adaboost 進行訓練的資料集，每筆資料均含有資料 x 及其真實標籤 y ， $w_{t,i}$ 代表第 i 筆資料在迭代 t 中的權重，權重越大也意味著該筆資料在當前迭代中可以獲得分類器更多的注意而有更大的機率會被分類正確，整個演算法進行的迭代次數與所建

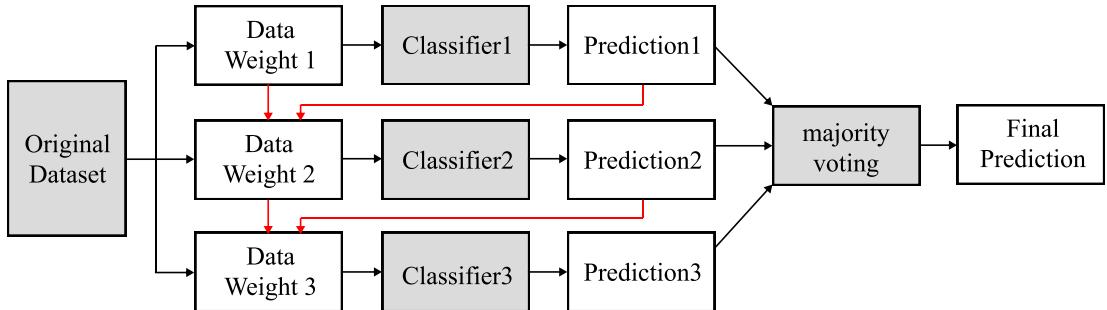


圖 2.3: 提升法簡例

立的 K 個分類器數量相同，每個迭代只訓練一個分類器，經過了 K 個迭代後每個分類器皆會被訓練過。在每個分類器分類完成後，會計算當迭代分類器所得到的錯誤率(error rate)，並用以計算分類器對每筆資料的影響係數 α 及更新每筆資料的權重 $w_{t,i}$ 。

演算法 1 Adaboost

- 1: Given dataset $D = (x_1, y_1), (x_2, y_2) \dots, (x_n, y_n)$, $t = 0$
 - 2: Initialize weights of each sample $w_{t,i} = 1/n$, $i = 1, 2 \dots, n$
 - 3: build K weak classifiers
 - 4: **while** for $t = 1$ to K **do**
 - 5: train weak classifier $C_t(D)$
 - 6: calculate the error rate of C_t : $\varepsilon_t = \frac{\sum_{i=1}^N w_{t,i} l(C_t(x_i) \neq y_i)}{Z_t}$
 - 7: **if** $\varepsilon_t > 0.5$ **then**
 - 8: break
 - 9: **else**
 - 10: $\alpha_t = \frac{1}{2} \ln(\frac{1-\varepsilon_t}{\varepsilon_t})$
 - 11: update sample weight $w_{t+1,i} = w_{t,i} \exp(-y_i C_t(x_i) \alpha_t)$
 - 12: **end if**
 - 13: **end while**
 - 14: **return** summarization of predictions by K weak classifiers $H(x) = \text{sign}(\sum_{i=1}^K \alpha_i C_i(x))$
-

該演算法透過每個資料的錯誤率與該迭代中判斷錯誤的樣本在下一迭代中建立一個可以彌補上個分類器無法準確分辨的區域，考量到有些難以分類的資料若勉強分類可能導致較高的錯誤率，在演算法最後回傳結果時，將根據使用錯誤率 ε_t 計算出的 α 來影響最後各分類器投票的權重，該方法除了解決了傳統的提升法的過度擬合問題，還使用了各分類器的盲區概念來建立下個弱分類器，使整體的分類效能可以有效的提升。

2.3.3 堆疊演算法

堆疊法 (stacking) [Wolpert, 1992] 為被規類為元學習 (meta-learning) 的一種方法

[Sagi and Rokach, 2018]，如圖 2.4 所示，堆疊法主要可以分為兩個階段，分類器訓練階段及元學習器 (meta-learner) 訓練階段，在資料前處理時，通常會將資料切割成兩個部分，第一部分用於訓練該集成系統中的子模型，另一部分則運用於訓練第二階段的元學習器。於訓練第一階段時，會獨立的用相同的資料集對各子模型進行分類訓練，該階段的訓練完成時，會凍結該階段各子模型的網路權重，以利於第二階段的元學習器訓練，第二階段的元學習器訓練，主要旨在訓練一分類器如何根據當筆欲分類資料，各子模型給出的分類結果，學習如何給出最終分類結果。而拆分資料集的意義為，若有一子模型於第一階段時因過度訓練導致對訓練集產生過度擬合之狀況，若於第二階段用同樣的訓練集輸入至子模型並提供分類結果給元學習器進行訓練，過度擬合的該子模型可能會使元學習器過度依賴但一子模型的分類結果，而該分類結果在測試集的準確率可能非常低落，導致集成系統整體的分類效能大幅下降。

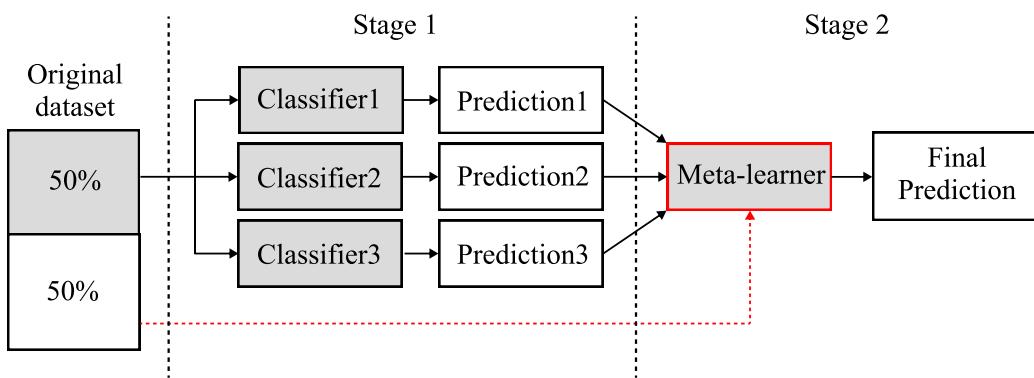


圖 2.4: 堆疊法簡例

在部分研究中，堆疊法可以結合不同類型模型特性經常被使用於提升分類效能，Aydin 等人 [Aydin and Gungor, 2018] 嘗試將各種不同的機器學習透過堆疊法方式進行組合並比較各組合之間的模型效能差異，其中包含邏輯回歸、隨機森林及 k -近鄰演算法 (k -nearest neighbors; KNN)，實驗結果顯示在用於電力非技術損失的偵測問題上，使用了上述三種機器學習方法的集成式學習模型可以取得最好的結果，並明顯優於 adaboost 及貝式網路 (bayesian network)，該文獻也嘗試了只使用邏輯回歸及隨機森林的集成式學習模型，結果大幅差於使用三種機器學習方法，間接地證明了堆疊法可以基於不同類型或數量的子模型而提升分類準確率。在運用堆疊法結合不同演算法並提升效能的文獻中，Khraisat 等人 [Khraisat et al., 2020] 結合了單一類別支持向量機 (one-class support vector machine) 及決策樹運用於入侵偵測系統 (intrusion detection system; IDS)，實驗結果也證明了堆疊法可以有效的使各模型發揮自身所長，有效的提升整體

分類效能並使結果明顯優於僅使用單一分類模型。Ouyang 等人 [Ouyang et al., 2018] 提出了三階段的堆疊集成式模型 three-stage multi-view stacking ensemble (TMSE)，第一階段稱為資料延伸階段，由於使用的資料具有嚴重的缺失問題，該階段使用不同的方法填補缺失值並分別輸入至第二階段中不同的子模型當中。該方法的第二階段由多組的子模型組合而成，各子模型均包含 xgboost、隨機森林、邏輯回歸及神經網路，運用不同方法填補缺失值可形成不同的資料型態，而不同的資料型態運用於訓練同樣的模型組合，各模型組合的各子模型也會用不同的特徵角度 (feature view) 來對資料進行分類，最後使用 xgboost 及邏輯回歸學習各組之預測並給出最終分類結果。

在醫學領域的方面，堆疊法也於部分文獻中被提及且使用，Rahman 等人 [Rahman et al., 2019] 使用 variational mode decomposition (VMD) 跟 discrete wavelet transform (DWT) 分析腦電圖，將堆疊模型應用於被大家廣泛使用的資料集中並分類出 focal EEG 及 non-focal EEG。Rahman 等人也針對堆疊模型中不同機器學習方法的順序及組合，進行分類效能的比較，從其實驗結果的準確率可以看出不同的堆疊組合及順序，均可以形成不一樣的分類結果，但其比較結果亦說明任何堆疊方法比起使用單一分類演算法，可以得到更好的分類結果。Yang 等人 [Yang et al., 2021] 在另外一個研究則使用集成學習中的堆疊方法進行帕金森氏症的分類，先對 T1W1-MRI、DTI 及各臨床資料進行特徵擷取並經過特徵混合、特徵篩選後，輸入由支持向量機、隨機森林、k-近鄰演算法及人工神經網路 (artificial neural network; ANN) 所組成的集成模型，並將各演算法對該資料集的分類結果在模型訓練第二階段輸入元學習器中，該學習器將會學習如何針對各演算法的特性去選擇該次分類應該採取的分類結果，以達成基於各演算法創造高準確率的分類結果。

2.3.4 動態的投票權重調整

加權多數演算法 (weighted majority algorithm) [Littlestone and Warmuth, 1994] 概念主要由 Littlestone 等人提出，在集成學習中經常會有多個分類器進行投票統整最終分類結果的情況，假設對於使用的訓練資料集沒有任何的先備知識，若一昧的平均各分類器的預測結果，將會使分類器中訓練較差或較不適合該資料集的分類器對最終的分類結果產生過大的影響，因此加權多數演算法的意義在於動態的調整每一個分類器預測結果之權重，使票票不等值，表現較好的分類器將會有更高的權重，也就能更大幅度的影響最終的分類結果，同樣地在訓練階段時表現較差的分類器將會獲得較小的權重值，使其分類錯誤的結果不至於過度干擾整體的分類準確率。Brunese 等人

[Brunese et al., 2020] 於文獻中也解釋了硬投票 (hard voting) 與軟投票 (soft voting) 的差別，硬投票的定義即為先判定每個分類器的預測結果在進行平均，舉例來說，若第一個分類器認為該筆資料為陽性的機率為 0.4，第二個分類器同為 0.4，而第三個分類器為 0.9，閾值 (threshold) 設為 0.5。硬投票的分類結果即為 $(0 + 0 + 1)/3$ ，而軟投票的分類結果為 $(0.4 + 0.4 + 0.9)/3$ ，經過計算硬投票的平均機率小於閾值會分類為陰性，而軟投票的結果高於閾值將分類為陽性。此外該文獻也基於軟投票機制為各分類器加上不同的權重，使原本較為優秀的模型在投票時能具有更高的影響性。實驗結果也顯示，在癌症診斷的資料集中，經過適當的分類器權重參數調整，可以取得相較於其他方法更優秀的表現。Ayan 等人 [Ayan et al., 2020] 將分類器的權重以 0 至 1 的小數編碼為一組解，並使用基因演算法 (genetic algorithm) 對該權重進行最佳化，基因演算法啓發於生物的遺傳機制，不同解之間透過交配可以互相交換各自的部分解並產生下一代解，為了使解更加多樣化，生成解時將會有一定的機率突變，而解的優劣將以各解的適應值 (fitness) 作為評斷的標準，該研究以均方誤差 (mean square error) 作為衡量解的標準，藉此使解的搜尋方向往更佳的方向邁進。實驗中，Ayan 挑選了表現最佳的三個卷積神經網路為基底的模型作為集成學習的子模型，分別為 Inception-V3、Xception 及 MobileNet，而基因演算法為三個模型找到的權重分別為 0.509、0.463 及 0.921，而相較於原本尚未最佳化權重的集成學習模型，平均提升了 0.5% 的準確率。

2.4 模型超參數最佳化

深度學習模型中往往可以顯著的影響模型最終的訓練成果，因此如何找到適合一個最適合該模型的超參數也變成了近期許多研究主要探討的目標 [Yi and Khac-Hoai, 2020, Tsai and Fang, 2021]，在這些更有效率地超參數最佳化演算法提出之前，都是由研究人員不斷的嘗試超參數與訓練，這樣的過程往往需要消耗大量的人力與時間。本章節將搜尋超參數的方法分為無模型的超參數搜尋與基於模型的超參數搜尋 [Yang and Shami, 2020]，其中無模型的超參數搜尋通常為較傳統的超參數搜尋方法，例如網格搜尋法 (grid search) [Shekar and Dagnew, 2019] 與窮竭搜尋法 (Exhaustive Search)，這些方法的搜尋過程都是於搜尋前就已經先行設定，例如網格搜尋法會在各維度中一定的超參數範圍內，按照設定好的步伐測試每一步的超參數組合。窮竭搜尋法則是將每個超參數中的所有可能都嘗試過，依此找出對於該模型架構較為適合的超參數組合。基於模型的超參數搜尋通常會產生一個或多個代理模型，該代理模型會嘗

試不同的超參數組合，其中與無模型搜尋方法不同的是，該代理模型會根據每次給出的超參數組合所回饋的獎勵或適應值來計畫與改變下一次的超參數搜尋方向，如此一來可以有效的在比網格搜尋法或者窮竭搜尋法減少大部分搜尋次數下找到近似於最佳解的超參數組合。

2.4.1 無模型超參數搜尋

網格搜尋法為超參數最佳化中最傳統的搜尋法之一，在該搜尋演算法會先設定一個固定的超參數搜尋範圍並依照一定的步伐劃出每一個需要驗證的超參數組合，如圖 2.5 中左圖所示，透過嘗試每一個在網格上的超參數組合得出解空間更好的超參數組合，也因該方法往往需要嘗試過多的超參數組合，找尋超參數的過程往往較為冗長，也近期的研究中也較常使用於搜尋架構較為簡單的分類演算法 [Klatzer and Pock, 2015, Shekar and Dagnew, 2019]，而 Li 等人 [Li et al., 2017] 也提出透過類似於採樣方式進行超參數組合搜尋的方法 hyperband，該方法強調於不找尋過多的解，定義計算資源若為 B (budget)，而後選解的數量若為 n ，則每個超參數組合可以使用的預算即為 $\frac{B}{n}$ ，假設選定多個均勻候選解，經過驗證後會淘汰比例 η 的數量，多次迭代後即可獲得表現最為優秀超參數組合。

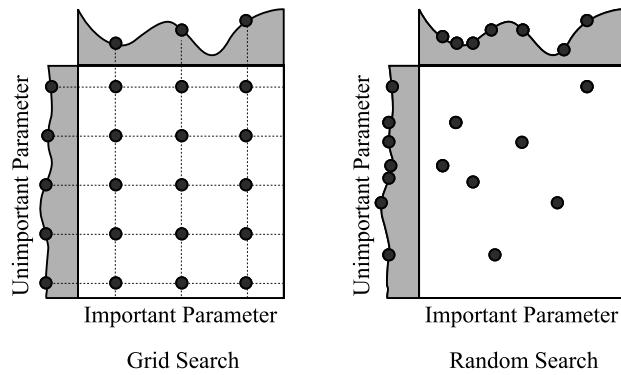


圖 2.5: 各超參數搜尋演算法比較圖

Bergstra 與 Bengio [Bergstra and Bengio, 2012]提出透過隨機搜尋法 (random search; RS)，來搜尋不同的神經網路模型超參數。如圖 2.5 中右圖，雖然隨機搜尋法能得到比手動調整超參數更佳的超參數組合，其完全隨機的特性使每個迭代皆為獨立的搜尋，也讓搜尋無法透過收斂得到更佳的結果。Andonie 等人在另一個研究 [Andonie and Florea, 2020]，改良隨機搜尋不會收斂的特性，以不同超參數對於資料集的重要程度進行排序，並將最重要的超參數作為權重最高的維度，並提出一個權重隨

機搜尋方法 (weighted random search; WRS) 來依序改變每一個維度在迭代的超參數內容。Andonie 等人透過連續型均勻分布 (uniform distribution) 決定超參數每個維度改變的機率，越重要的超參數代表該組維度可能透過較小的超參數調整卻得到極大的效能差異，也代表該維度必須擁有越大的改變解機率，也能投入更多資源在搜尋該維度的解空間。研究結果顯示透過可收斂的搜尋方法，權重隨機搜尋方法能在 CIFAR-10 資料集中得到遠高於傳統隨機搜尋法更佳的效能。在十分類的測試資料集也得到 85% 的分類準確率，且他們發現卷積神經網路的第一個全連接層的神經元數量將大幅改變模型的分類能力。

2.4.2 基於模型的超參數搜尋

貝式最佳化 (Bayesian optimization; BO) [Snoek et al., 2012] 是基於統計模型所建立的超參數最佳化方法，相較於隨機搜尋法，貝式最佳化將解空間的每一組解利用指定的數學分布進行抽樣後，計算出每一個超參數解的真實適應值，再使用所有初始樣本的適應值來估計超參數解空間的大致樣貌。貝式最佳化的每次迭代都會先計算不同超參數解的收穫函數 (acquisition function)。透過解空間估計的更新以及收穫函數的計算，貝式最佳化以逐漸收斂的方式找到較佳的超參數組合。Yi 和 Bui [Yi and Bui, 2020] 將貝式最佳化與 meta-learning 進行結合，並提出一個 HyperNet 神經網路架構，利用貝式最佳化的特性來搜尋長短期記憶模型的各項超參數。該研究將此神經網路架構應用於韓國的高速公路流量預測系統，以每分鐘作為時間間隔，並分別切割 66 天與 16 天為訓練資料集與測試資料集。研究結果顯示透過貝式的超參數最佳化，預測模型能在不同路口產生的測試資料集皆得到趨近於真實資料的高速公路流量預測，且實驗顯示以貝式最佳化取代隨機搜尋法，更有機會得到適合此資料集的超參數設置。

2.4.3 超啓發式演算法於超參數搜尋

近年有部分研究嘗試以超啓發式演算法，應用於機器學習或深度學習的超參數最佳化。Young 等人 [Young et al., 2015] 將基因演算法 (genetic algorithm; GA) 作為卷積神經網路調整超參數的搜尋策略，提出 MENNDL (multi-node evolutionary neural networks for deep learning) 模型，並將 CIFAR-10 作為衡量模型效能的訓練與測試資料集。Young 等人指出透過演化式的超參數搜尋，找到的最佳超參數組合往往與起始組合大不相同，例如卷積神經網路的 filter 數量可能大幅影響模型的分類效能，因此它將隨著搜尋的次數提升而大幅上升，而卷積核的大小則可能因為更細微的調整而降低，使最終

的結果得到更低的卷積核大小。研究結果顯示以基因演算法所找到的超參數組合能勝於手動調整超參數，且大幅節省開發者調整參數的資源，且能找到比隨機搜尋法更佳的神經網路架構。Soon 等人 [Soon et al., 2018] 使用粒子群最佳化 (particle swarm optimization; PSO) 進行卷積神經網路的架構調整，並結合圖像處理技術來建立一個自動車牌辨識系統。Soon 等人將系統架構分成圖像切割、廠牌偵測、卷積神經網路建立與超參數最佳化四個步驟，並以一個 13 分類的汽車廠牌圖像庫作為訓練與測試的資料集。研究結果顯示透過粒子群最佳化的超參數調整，此系統將能大幅贏過未調整過的卷積神經網路分類模型及其他傳統的圖像偵測方法，並達到 99.1% 的分類準確率。Tsai 等人 [Tsai et al., 2020] 使用模擬退火法 (simulated annealing; SA) 調整深度神經網路的超參數架構，使超參數之數值，能隨著搜尋過程逐漸收斂於特定的解空間區域。Tsai 等人將此方法調整每一個全連接層的神經元數量，並以常態分布來決定每次進行神經元數量改變時的數值，應用於台中公車的乘客人數預測。研究結果顯示透過超參數最佳化，模擬退火法能在四種不同路線公車資料集皆得到最低的均方根誤差 (root-mean-square error)，且可以獲得比支援向量機(support vector machine; SVM) 與隨機森林法 (random forest; RF) 等傳統機器學習方法較好的準確率，給予較準確的乘客人數。Serizawa 與 Fujita [Serizawa and Fujita, 2020] 提出了線性遞減權重的粒子群最佳化 (linearly decreasing weight particle swarm optimization; LDWPSO)，主要概念為將粒子群最佳化的慣性速度權重更改為隨著迭代次數的提升而逐漸減少，藉此使每個粒子在前期與後期所搜尋的方向有所變化，進一步達到全域搜尋與區域搜尋的目的。作者以 MNIST 與 CIFAR-10 作為此方法的訓練及測試資料集，分別在這兩種資料集得到 98.95% 與 68.92% 的分類準確率。研究結果顯示以 LDWPSO 找到的最佳超參數，模型在訓練的準確率與誤差值在訓練初期的 epoch 就會得到更接近訓練後期的數值。這代表超參數的選擇將可能直接改變模型訓練的效能，若使用的超參數組合不佳，將可能需要耗費更多的 epoch 數量來得到相應的分類或預測能力。

2.5 總結

現實中的時序電力資料大多具有幾個特點：週期不定的時序性特徵、不平衡的資料與大量的資料缺失值，在本章的第一節當中，許多文獻透過許多不同的方式以擷取電力資料中長短不一的時序性特徵，包含混合長短期記憶模型與卷積神經網路、進行不同時間間隔的資料摺疊便於使用擅長擷取二維特徵的深度學習模型學習時序特徵，

為了避免混合模型中不同種類的特徵進入全連接層時被較凸顯的特徵類型所稀釋，本論文將提出一個基於集成式學習的異常用電偵測方法，根據本章第二節集成式學習相關文獻，集成式學習的架構，無論裝袋法、提升法或者堆疊法均具有提升分類結果的可能性。本論文試想，若能建構不同種類的子模型，例如擅長記憶長期時序特徵的長短期記憶模型、擅長擷取短期空間特徵的一維卷積神經網路模型及擷取不定時序特徵的多頭注意力機制模型，並透過一深度神經網路作為堆疊法中的元學習器用於學習第二節末小節提及之加權多數演算法，預期能有效的解決時序電力資料中不定的時序性特徵問題。為了使作為元學習器的深度神經網路能於電力資料集取得更佳的表現，本論文將使用超啓發式演算法於該元學習器的超參數最佳化，近期超參數最佳化文獻如本章第三節所示，大多的超啓發式演算法在面臨區域最佳解時，雖然有群體智慧的機制能有一定機率跳離該解空間區域，但面對解的維度過大時效果仍相當有限，本論文使用之搜尋經濟學演算法透過切割解空間與期望值機制，使搜尋方向有一定機率能跳回強制劃分的解空間進行搜尋，該機制雖然犧牲了些許的收斂速度，但比起其他超啓發式演算法，可以大幅度的增加搜尋解過程的多樣性，也因此有機會可以找到更好的超參數組合。

第三章 自適應集成式異常偵測系統

3.1 演算法概念及架構

本論文提出之系統可以分為兩個部分，分別為「集成式分類系統」與「權重分配最佳化」，其中集成式分類系統如圖 3.1 所示，透過集成學習中的堆疊法，我們可以將擅長於不同特徵類型的模型組合，以提升分類效能。假設從圖 3.1 中的任何一條路線出發，輸入的資料集均只能透過一種類型的模型進行預測或分類，若使用卷積神經網路僅能根據空間性特徵進行該資料集的分類，或使用遞迴神經網路僅能從時序性的角度來審視該筆資料。然而即使在一個資料集中，每筆資料可以用以判斷該筆資料類別的特徵均不相同，有時需要鄰近的資料變化特徵，有時可能需要每隔一段時間的週期性特徵，因此透過元學習器整合三個不同類型模型的預測結果，預期可以有效的提升效能。

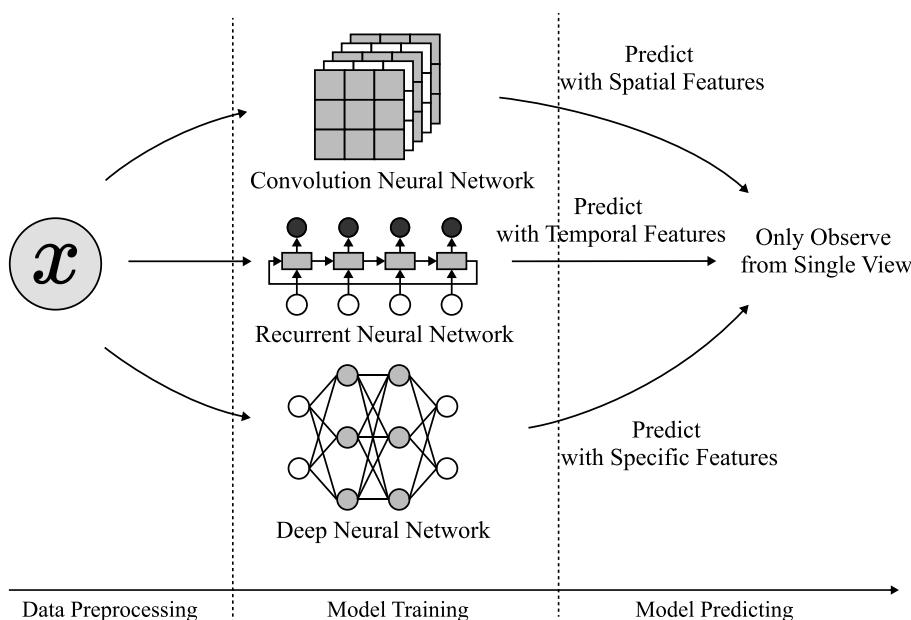


圖 3.1: 集成式學習概念圖

在集成學習中要如何分配各分類器的投票權重為一大重點，如圖 3.2 所示，左上角圖 (a) 中的不同顏色區塊代表各分類器於該資料集中認為是異常或者非異常的超平面，不同的模型根據自身不同的特徵觀點 (feature-view) 會有不同的分類結果。圖 (b) 為使用集成學習並配置各模型相同的投票權重，假設被包含在內的為模型判斷為異常的部分，因此圖中若有兩個以上模型的判斷範圍重疊，由於有相同的投票權重，根據多數決原則集成學習系統將會視其為異常，反之若只有一個模型以下認為該筆資料為異常將會被視為正常資料。而在本論文中提出的動態權重分配方法預期如圖 (c) 所示，與相同權重的圖 (b) 不同，元學習器將會學習如何調整各模型的權重分配，使整個系統的分類效能能達到最大化的提升。

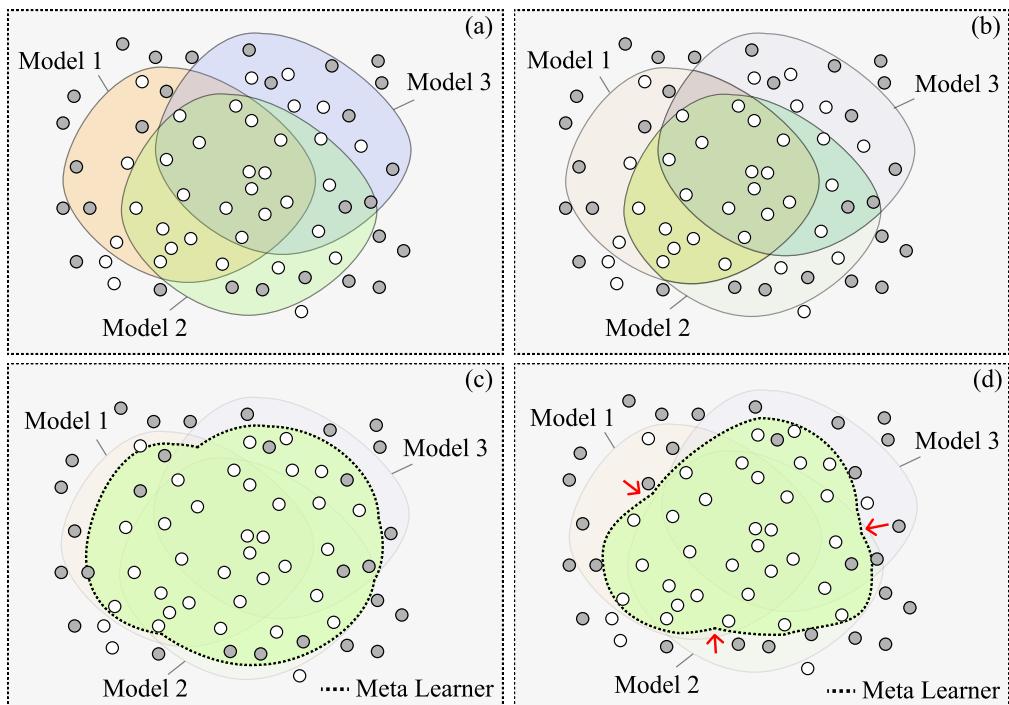


圖 3.2: 動態權重調整

針對本論文作為集成式系統中元學習器的深度神經網路，本論文欲使用超啓發式演算法來調整該神經網路的架構與超參數，其效果示例預期如圖 3.2 中的右下角圖 (d) 所示，超啓發式演算法將透過演算法的三步驟：改變解 (transition)，衡量解 (evaluation) 及評估解 (determination)。運用類似於生物群體中的群體智慧，使多個候選解往更佳解在每個迭代中根據群體中較好解資訊，逐步逼近區域最佳解或全域最佳解，而經過超啓發式演算法的超參數最佳化，元學習器的動態權重分配效能將會更加提升，並間接的影響集成式分類系統的效能。然而經典的超啓發式演算法於搜尋解的後期容易困於區域最佳解且由於群體中的解過於相似，產生許多重複的不必要計算，Tsai 發現該現

象並提出了搜尋經濟學演算法 [Tsai, 2015]，該演算法以投資者對金融市場進行投資為啓發，各投資者會觀察金融市場中何者具有潛力並計算期望值，並根據期望值來選擇較適合的投資標的。搜尋經濟學演算法的機制由兩個主要概念組成：區域劃分與期望值計算。

- **區域劃分 (region division)**：在搜尋經濟學演算法中，解空間將會根據固定特色解被切割成數個區域 (region)，以本論文的深度神經網路為例，若根據模型的學習率為切割區域的依據，區域 1 的解中的學習率固定為 0.01，區域 2 的解的學習率固定為 0.001，依此類推，每個區域中的解均有該區域的固定特色，而投資者將會根據期望值而投資各區域的樣品 (sample)。
- **期望值計算 (expected value calculation)**：期望值為搜尋經濟學演算法中投資者選擇要投資哪個區域的根據，期望值會根據區域中解驗證的結果而變化，解的適應值越高將導致其所屬之區域期望值越高，另外期望值也將根據投資次數與迭代數量進行調整，若有一區域許久未被投資，該區的期望值將會提高，這也是搜尋經濟學演算法於後期有機率跳脫區域最佳解的重要原因之一。

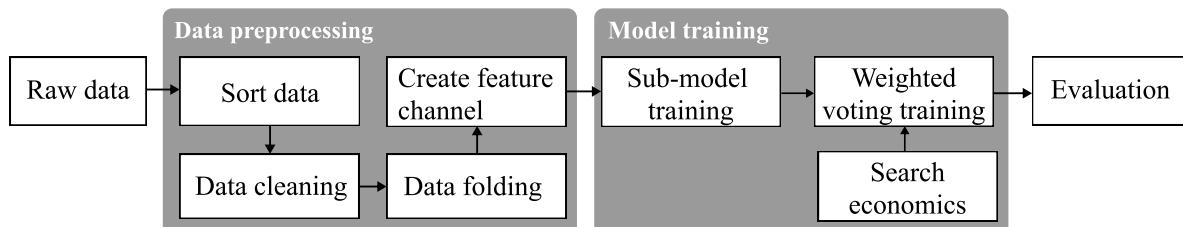


圖 3.3: 系統流程圖

圖 3.3 為本論文方法之系統架構圖，首先先進行資料前處理，該部分細節將詳述於小節 4.2，會透過資料排序、資料清理、資料折疊及建構特徵通道方法來修復資料集的同時，穩定實驗結果。在模型訓練部分將分為兩個部分，分別為子模型建構與訓練及動態權重調整與最佳化。在子模型建構與訓練中，將獨立建立一維卷積神經網路、長短期記憶模型與多頭注意力卷積神經網路，這三個模型將輸入同樣的資料集並分開獨立進行模型訓練，直至三者均能取得自身最佳的分類效能。在模型訓練的第二部分，將訓練作為元學習器的深度神經網路並透過搜尋經濟學演算法進行神經網路架構與超參數的最佳化，使其能在某一起參數組合下，分配最適當的權重給各模型之預測結果，最後進行模型驗證、參數實驗及結果分析。

3.2 集成式分類系統

3.2.1 子模型建構與訓練

本論文所提出之集成模型由多個深度學習模型所建構，分別為一維卷積神經網路、長短期記憶模型及多頭注意力卷積神經網路。其中一維卷積神經網路主要擷取電力資料中的短期時序特徵，透過一個可固定大小的濾波器 (filter) 在多維的電力資料上進行滑動，由於濾波器的擷取範圍必須相連，在卷積神經網路中擷取的特徵主要為鄰近時間之特徵相關性。

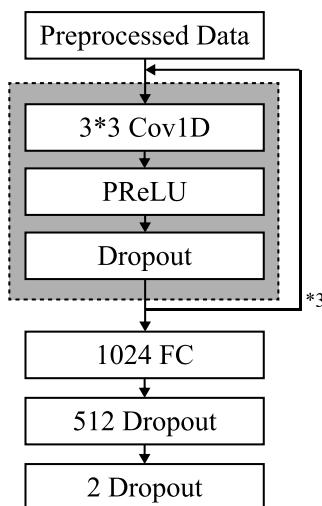


圖 3.4: 一維卷積神經網路架構圖

圖 3.4為本論文所採用之卷積神經網路模型架構，輸入層部份將接收具有兩維特徵的電力資料，分別為電力用量通道及缺失值特徵通道，接著使用分別為 50 個與 100 個且卷積核大小為 3 的卷積層。經過卷積層的特徵擷取後，連接至多層的全連接層進行分類。而層與層之間將加入激勵函數與丟棄層，激勵函數部分使用 PReLU 激勵函數 [He et al., 2015]，將輸出值轉換為非線性，相較於傳統 ReLU [Glorot et al., 2011]，PReLU 使用了自適應變數 a_i ，該變數將以動量方法進行更新，當輸入值為負的狀況下可以避免梯度爆炸 (gradient exploding)，又可有效地學習到該資料集所適合的 a_i 值，以取得較佳的分類能力。丟棄率則設定為丟棄 20% 至 60% 之特徵傳遞值。

集成模型的第二部份採用長短期記憶模型，該模型為傳統遞迴神經網路 (recurrent neural networks; RNN) 的改良，如圖 3.5，長短期記憶網路透過精巧的記憶控制單元設

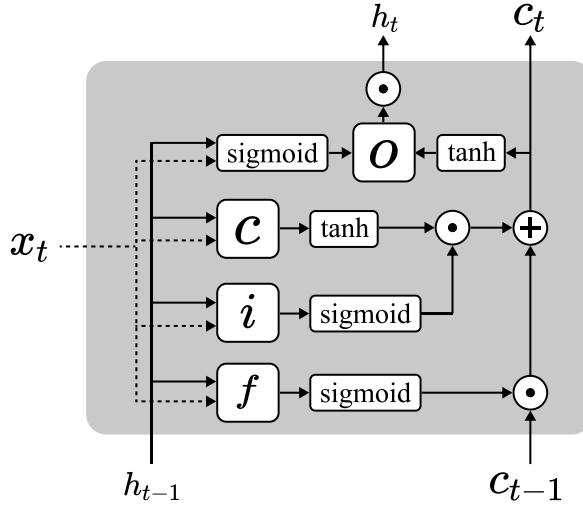


圖 3.5: 長短期記憶架構圖

計，使該模型可以不只強調於短期記憶，同時可以留下長期記憶的資訊。本論文將使用該模型的時序記憶特性，加入集成模型以擷取卷積神經網路較不擅長的長期時序特徵。長短期記憶網路的主要結構如圖3所示，大致分為三個部份，其中包含輸入閘道 (input gate)、遺忘閘道 (forget gate) 與輸出閘道 (output gate)。這三個閘道主要目的為控制每個迭代的隱藏狀態 (hidden state) 及記憶細胞狀態 (memory cell state)。其中隱藏狀態主要為傳遞各時間之短期記憶特徵，而記憶細胞狀態負責記憶長期時序特徵資訊。

$$i_t = \sigma(W_x^i x_t + W_h^i h_{t-1} + b_i), \quad (3.1)$$

$$f_t = \sigma(W_x^f x_t + W_h^f h_{t-1} + b_f), \quad (3.2)$$

$$o_t = \sigma(W_x^o x_t + W_h^o h_{t-1} + b_o), \quad (3.3)$$

$$\tilde{c} = \sigma(W_x^c x_t + W_h^c h_{t-1} + b_c), \quad (3.4)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}, \quad (3.5)$$

$$h_t = o_t \odot \phi c_t, \quad (3.6)$$

輸入層訊號 i_t 如公式 3.1，主要使用神經元權重 W_x^i 與偏差值 b_i 的訓練，學習到應該要從當前的輸入隱藏狀態及輸入資料中保留何項資訊並保存到記憶細胞狀態之中。遺忘層訊號 f_t 如公式 3.2，使用神經元權重值 W_x^f 與偏差值 b_f 並透過 sigmoid 函數使大部分的閘門控制訊號為值 0 或 1，使用該控制訊號來控制上一個時間點的記憶細胞狀態中何項特徵對於未來的時序分類或者預測是較無影響力。在該輪的遺忘閘道將會從記憶細胞狀態中移除，經過輸入閘道及遺忘閘道後，記憶細胞狀態將會以 公式 3.4 的該輪

所輸入的新資訊 \tilde{c}_t 與遺忘閘道所完成的遺忘後剩餘記憶細胞狀態資訊，透過公式 3.5 得出當前時間的新記憶細胞狀態 c_t ，輸出閘道訊號 o_t 與輸入閘道及遺忘閘道相同，如公式 3.3，使用輸出權重值 W_x^o 為控制該輪更新過後的記憶細胞狀態中有何項資訊是需要被輸出至下一階段的長短期記憶模型當中，公式 3.6 中，下一個時間的隱藏狀態 h_t 將使用當前時間之記憶細胞狀態 c_t 與輸出閘道訊號 o_t 進行矩陣乘積運算，並作為下一個時間的隱藏狀態輸入。本論文所使用之長短期記憶模型將使用上述介紹之兩層長短期記憶層，輸入層維度分別為 1,029 及 64，輸出層分別為 64 及 32，在模型最後輸入 2 層分別具有 32 及 2 個神經元的全連接網路。

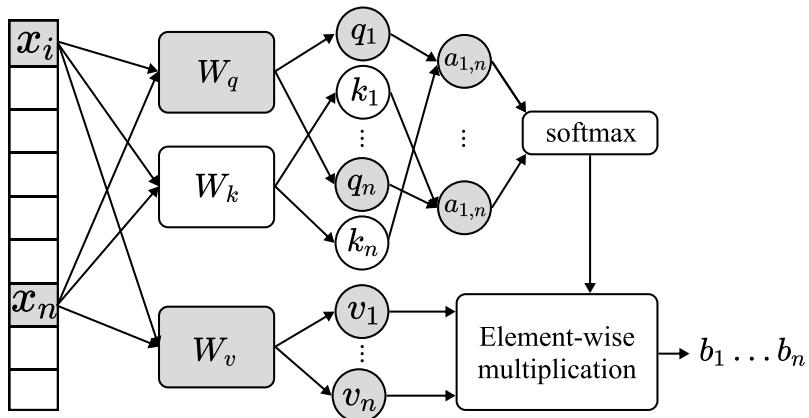


圖 3.6: 自注意力機制示例

注意力機制運用類似於人類的視覺注意力原理找出各時間點或各特徵間的關聯性，並算出注意分數 (attention score) 以告訴其他學習層該筆資料的相關性特徵，如圖 3.6 所示，注意力機制網路中共有查詢權重 W^k 、關鍵權重 W^q 及值權重 W^v 需要進行訓練，以一時間序列的資料為例，每個時間點的資料點都將乘上代表查詢矩陣 W^q 、關鍵矩陣 W^k 及值矩陣 W^v 。並獲得該時間點自己的查詢值及關鍵值後，與序列中其他值進行點乘積 (dot-product) 運算得到注意分數 α ，以圖 3.6 為例，透過 q_1 與 k_n 進行運算，即可得到 x_1 與 x_n 的注意分數 a_n ，依此計算方式，可以使用權重矩陣獲得每個時間點對於其他時間點特徵的注意力分數，將該注意力分數經過 softmax 函數後將各'值與各自的 v 進行矩陣乘積運算，最後所得之值 b_1 到 b_n 即為該注意力層所輸出之相關性特徵值，該特徵值的意義為，從某一個角度的觀點來看，該時間點對於其他時間的相關性如何，而研究 [Vaswani et al., 2017] 所強調之多頭注意力機制即為使用多個權重矩陣 W^k 、 W^q 及 W^v 對於各時間數據進行矩陣計算，從產生出的多個 q 值、 k 值及 v 值進行上述計算，達成從不同的觀點來定義該資料集的「相關性」這個特徵，因此稱為

多頭注意力機制。

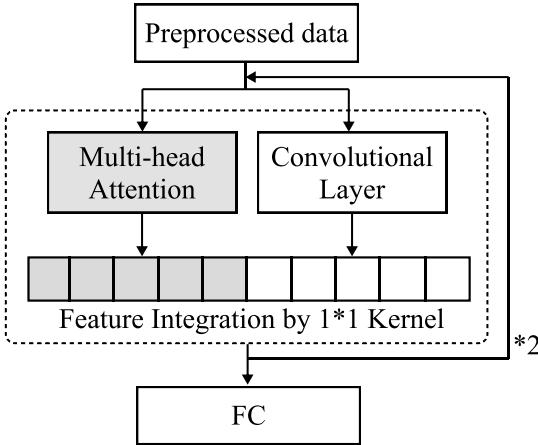


圖 3.7: 多頭注意力模型

本論文使用之多頭注意力卷積神經網路架構如 圖 3.7，在模型前期的特徵擷取階段，分別將前處理過後的資料輸入至具有膨脹卷積 (dilated convolution) 機制的二維卷積層及多頭注意力機制層，接著將兩者擷取之特徵銜接並輸入至過濾器核心大小為 1 的卷積層中進行兩者結果的特徵擷取並整理輸出以利下一輪的混合模型進行輸入最後連接至兩層使用 dropout 的全連結層進行分類。二維卷積神經網路使用二層具有 32 個卷積核的卷積層，注意力層則為 7 頭注意力層。

3.2.2 動態權重分配

圖 3.8 為本論文主要提出之集成學習總模型結構，將使用上述提及的三種深度學習網路架構，堆疊並以各模型的不同特徵角度觀察資料集，擷取不同的特徵所做出的分類結果進行投票分類。而該分類結果的投票權重將以動態的投票權重學習方法進行分配，分配權重的深度神經網路將依照不同模型所預測出的類別機率動態的決定該筆資料的投票權重該如何進行分配，因此該模型能基於各模型的分類能力，整合出更佳的分類結果與效能，進行更準確的異常用電分類。

輸入資料部份將以進行缺失值遮罩填補及分位數轉換法，並根據不同的模型輸入需求，摺疊成一維或者二維的資料型態輸入。整體分類架構如 圖 3.8 所示，訓練第一階段將會先進行三個分類模型各自的模型訓練，如圖所示，此方法將訓練研究方法中主要介紹的三種模型，分別為長短期記憶模型、一維卷積神經網路模型與卷積神經網路-注意力機制混合模型。三個訓練均使用 公式 2.2 的交叉熵函數，做為模型訓練方向並進行損失值反向傳遞訓練，集成模型的第二階段將會把上述所完成訓練之分類模型

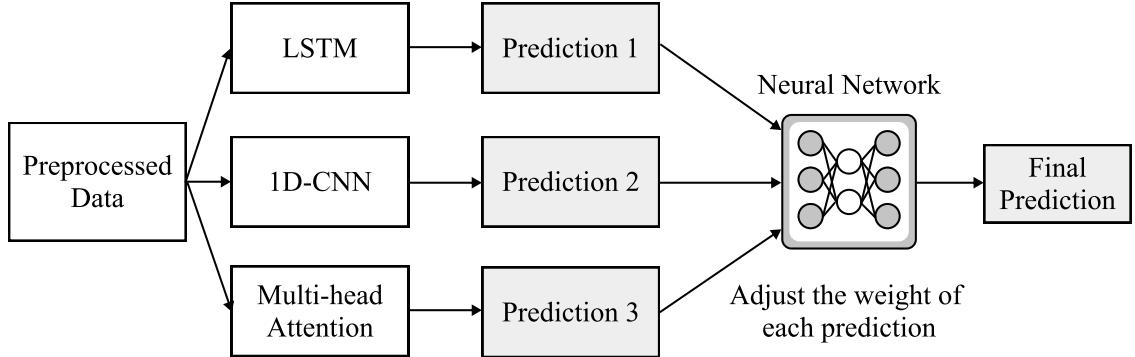


圖 3.8: 堆疊架構示意圖

進行權重凍結 (weight freeze)，使其保持在訓練完成時良好的分類效能，不被權重網路訓練的影響導致分類效能變差。權重固定後，將使用 公式 2.2 中的交叉熵函數對於權重網路進行訓練與回饋。由於在實驗過程中發現在後面的訓練迭代，常常會有過度擬合的情況發生，因此在所有模型訓練階段，將會保存覆蓋精準召回曲線下最高面積的模型並在驗證時讀取該模型之權重進行測試集分類模型效能評量。

3.3 權重分配最佳化

本論文將以搜尋經濟學演算法應用於元學習器的動態權重調整網路最佳化，本論文所使用的元學習器將以深度神經網路作為學習架構，而以往的研究人員設定深度神經網路架構或其超參數時，大多使用試誤法根據經驗進行參數嘗試，而這樣的方法往往不能有效的找到比較好的超參數組合，而近期研究同小節 2.4 所述，多使用貝式最佳化、基因演算法或粒子群最佳化等演算法進行超參數最佳化，然而雖然這些方法均能於解空間中有效地找到更佳解，但搜尋後期容易因搜尋機制困於區域最佳解。搜尋經濟學演算法透過資源分配、遠見搜尋及市場調查能有效的保持高效搜尋的效能下，在搜尋後期有大量機會跳脫出區域最佳解，本章節將切分演算法流程、編碼、資源分配、遠見搜尋、市場調查及應用於超參數最佳化的範例進行搜尋經濟學演算法的方法介紹。

3.3.1 搜尋經濟學演算法

經濟搜尋演算法為 Tsai 等人於 2015 年 [Tsai, 2015] 提出的超啓發式演算法，流程

及步驟如演算法 2 所示，該演算法會先初始化在演算法搜尋過程中會使用到的參數，分別為投資者 (searcher) 數量 s 、欲切割之區域 (region) 數量 r 、市場中的樣品 m ，並將市場中各樣品的隨機初始超參數組合與各區域用於計算投資者對各區域投資機率的係數 t_h^a 及 t_h^b 。在初始化結束後，將進行該演算法的主要三階段「資源分配」將進行一次，「遠見搜尋」與「市場調查」則因設定的迭代次數而異。

演算法 2 Search Economics for Hyperparameters Optimization

```
1: /* Initialization */
2: Initialize the content of hyperparameters
3: Initialize the number of  $s, r, m$ 
4: Initialize the ratio  $t_h^a, t_h^b$  for each region
5: /* Resource Arrangement */
6:  $s, r, m = \text{ResourceArrangement}()$ 
7: while the termination criterion is not met do
8:   /* Vision Search */
9:    $v = \text{Transition}(s, r, m)$ 
10:   $E = \text{Evaluation}(s, r, m, v)$ 
11:   $s, m = \text{Determination}(E, s, r, m, v)$ 
12:  /* Marketing Research */
13:   $t_h^a, t_h^b, r_h^b = \text{Marketing Research}()$ 
14: end while
15: return the best solution
```

3.3.2 編碼及初始化

深度神經網路的超參數組成本論文將其分為兩類，分別為架構參數及訓練參數，其中架構參數主要改變深度神經網路的樣貌，例如神經元層數、各層所含有的神經元數量及使用的激勵函數等；訓練參數主要調整在訓練框架中的學習率、批次大小等影響梯度傳遞之參數。這兩類參數對於深度神經網路的訓練都有著重大的影響，在面對不同的資料集時，所需要的超參數組合也不盡相同，也就是說某一超參數組合在 A 資料集能取得趨近於全域最佳解的效能，相同的超參數組合應用於 B 資料集時，未必能取得與 A 資料集同樣優秀的結果。也基於此原因，本論文採用超啟發式演算法對其進行最佳化，並預期最佳化主要提出之集成式分類系統在異常用電偵測任務上的效能。

本論文的解編碼示例如圖 3.9 所示，使解可以用以調整深度神經網路的神經元層數、各層的神經元數量、批次大小及學習率，圖中為一組解的範例，該解將會出現於投資者所攜帶的樣品與各區域中的樣品。

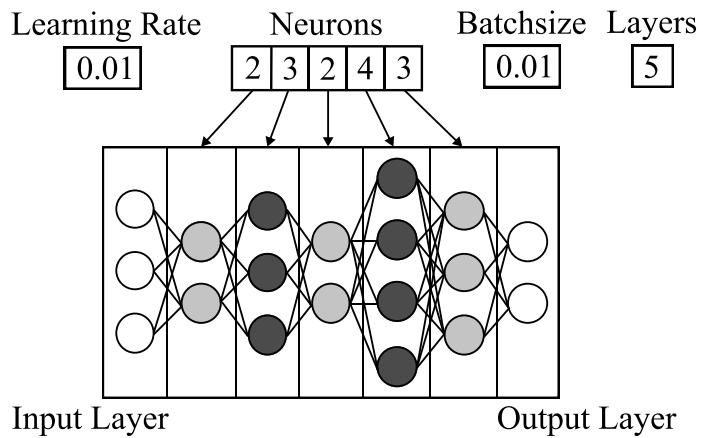


圖 3.9: 編碼示意圖

3.3.3 資源分配

在資源分配階段，將會平均的部屬各投資者與樣品的分布狀態，進行「解空間切割」與「樣品分配」，解空間切割階段，本論文將以模型學習率作為各區域切割的依據，雖然不同的神經元層數與神經元數量對於深度神經網路的訓練影響較為顯著，但若以神經元層數作為切割依據，將可能使各解空間區域的超參數組合平均適應值差異過大，例如神經元限定為一層的區域產生的樣品適應值，由於神經元僅有一層的模型效能大多明顯小於一層以上的神經元結構，因此該區域的模型將會高機率的弱於擁有兩層以上神經元層結構的深度神經網路。而若基於學習率切割解空間，每一個區域都有可能產生所有神經元層數量限制內的候選解，並依據不同的神經元層數與其對應的神經元數量，搭配適合的學習率可以使該深度神經網路有更好的效能，再者，在不同的學習率中也有大部分學者普遍認為較為泛用的學習率，該學習率中產生的大多候選解可能優於其他學習率，但其差距並無以神經元層做為切割區域依據大，因此在解的搜尋後期，在其他區域找到比目前的區域最佳解更好的解機率也會較高。

3.3.4 遠見搜尋

遠見搜尋為搜尋經濟學演算法中最重要的階段，該階段將以搜尋者尋找更好的區域進行投資作為出發點，本節將遠見搜尋拆分為「資訊分享與創新」、「期望值計算」及「樣品及區域選擇」階段進行說明。

1. 資訊分享與創新：資訊分享為遠見搜尋的第一個階段，搜尋者會攜帶自身搜尋過最佳的樣品到各個區域的樣品進行資訊的交換與更新，該資訊分享的動作也是在

該演算法中解逐漸變好的關鍵，由於各區域中的解皆有各自的特色部分解，因此當搜尋者攜帶自身找過最佳解與各區域交換資訊並產生新解時，各區域能基於自身區域的特色延續最佳解特性，並藉此往該區域中更好的解方向前進。

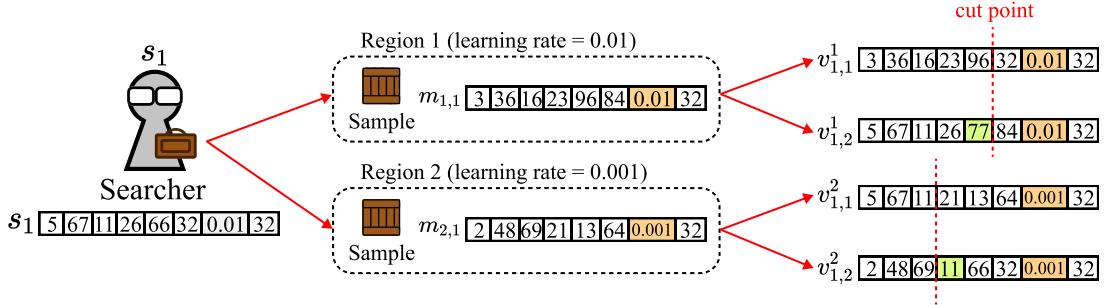


圖 3.10: 分享資訊示例

如圖 3.10 所示，該例為搜尋者攜帶自身最佳樣品 s_1 到區域 1 及區域 2 進行投資，從圖中可以發現區域 1 及區域 2 中解的黃色部分為固定解，也就是該區的特色部分解。在本論文中搜尋者的解與各區域的解將使用基因演算法中的單點交配方法交換彼此的解資訊，樣品 s_1 與區域中樣品 $m_{1,1}$ 及 $m_{2,1}$ 將透過一個隨機產生、最大值為解長度的正整數作為單點交配的分割點進行前後段的資訊交換。經過單點交配後，每組新產生的解將根據突變機率在解長度內隨機進行突變，如圖 3.10 中解 $v_{1,2}^1$ 的綠色部分，若以原始兩解交配規則，該維度的值應為 s_1 的 66，但由於突變機制，綠色部分解隨機改變為 77，突變機制對於搜尋經濟學演算法尤其重要，因為在演算法中所有的解內資訊都是由其他解交配而來，突變機制使演算法能往未知的解空間進行探索，以致搜尋到更好的超參數組合。

2. 期望值計算：在市場投資中搜尋者與各區域的樣品進行解資訊的交換，接著搜尋者將會計算各區域的投資潛力並前往投資，投資潛力主要由三個部分所構成，如公式 3.7 所示，由歷史投資頻率 T_h 與區域內的最佳評估值 M_h 組成。

$$e_{n,h} = T_h M_h, \quad (3.7)$$

歷史的投資頻率 T 計算方式如公式 3.8， t_h^b 為區域 h 未被投資的迭代次數， t_h^a 代表區域 h 被投資的迭代次數，因此若區域 h 在整個解搜尋的過程中被投資的次數較其他區域多，歷史的投資頻率值將會較其他區域低，進而導致其他區域的被投資機率可能變高，該公式也反映出搜尋經濟學演算法的核心機制，當某一區域許

久未被搜尋者所投資，該區域的 T 值將會越來越大，在經過許多迭代後也將有更大的機率回來搜尋，這個區域的解並有可能發現從未發現過的更加解。

$$T_h = \frac{t_h^b}{t_h^a}, \quad (3.8)$$

最後則為區域內的最佳評估值 M_h ，公式 3.9 中 $f(r_h^b)$ 為歷史中於區域 h 找到過的最佳適應值，分母則為當前所有區域所有樣品的適應值總和，而若該值越大也代表著該區域在整個搜尋過程中所找到過的最佳解比當前所有解好得多，對於搜尋者來說也更具投資潛力。

$$M_h = \frac{f(r_h^b)}{\sum_{l=1}^h \sum_{i=1}^j f(m_{l,i})}, \quad (3.9)$$

在期望值計算中，搜尋經濟學演算法分別從「投資次數」與「歷史解優劣」兩個方面來評估各區域的投資潛力，選擇一個好的投資區域也意味著可以使演算法明確的知道若欲找到比當前最佳解更好的解，應該往哪個區域搜尋，但同時也根據投資次數來給予不被看好的解機會，增加解多樣性的同時，也提高跳脫區域最佳解的機會。

3. 樣品及區域選擇：在搜尋者對各區域進行投資後，每個搜尋者對每個樣品將產生兩個解，因此每個區域將會產生比原先多一倍的樣品數量，本論文在此將使用比較選取法 (tournament selection) 進行各區域新一批樣品的篩選，將根據參數設定隨機從該區域所有舊樣品及新樣品中選取數個，接著比較各樣品的適應值大小，適應值較大的樣品將會被留下作為新迭代的新樣品，在有效保留優秀解的同時，由於比較對象為隨機挑選，適應值並非名列前茅的樣品也有機會被留於作為下一迭代的新樣品。

3.3.5 市場調查

經過了遠見搜尋後，搜尋者對於各區域的投資次數或者找到的最佳解均可能改變，因此在市場調查階段將會更新所有區域的被投資次數、不被投資次數及目前找到的最佳解。搜尋者在遠見搜尋階段決定新投資區域後，該區域的被投資次數將會增加，而為確保公式 3.8 不變成分母為 0 的無意義公式，被投資次數將會變為 1。在該階段更新市場資訊後，將有助於下一迭代各搜尋者的投資策略與解的搜尋方向，以找到更佳的超參數組合。

3.3.6 簡例

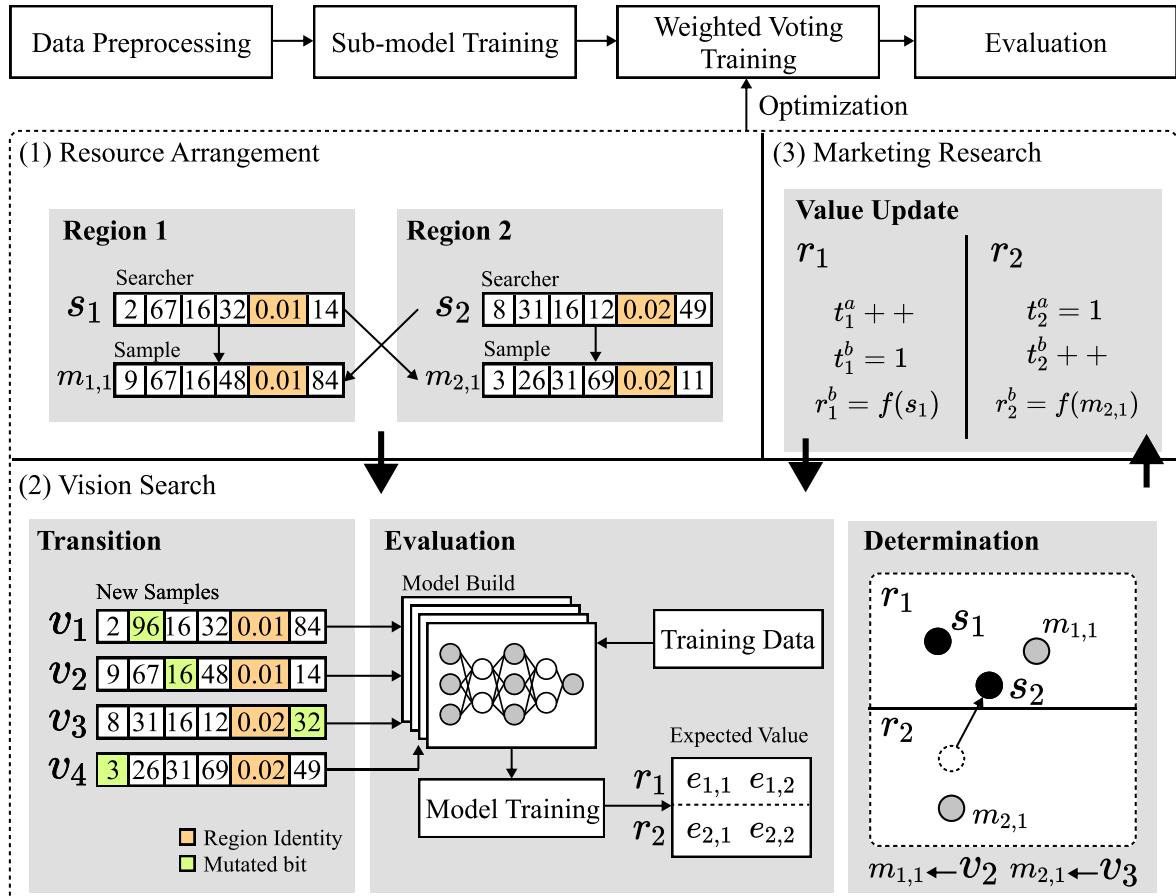


圖 3.11: 搜尋經濟學演算法應用於超參數最佳化示例

本小節將提供簡單的例子示範搜尋經濟學演算法搜尋深度神經網路的超參數組合，如圖 3.11，首先在資源分配階段，本範例將每個區域的樣品數設為 1， $m_{1,1}$ 及 $m_{2,1}$ 代表區域 1 中的第一個解及區域 2 中的第一個解。在整個解空間中共有兩位平均分配給各區域的搜尋者，分別為 s_1 及 s_2 ，而資源分配中的每個解都進行隨機初始化，以利於開始搜尋解時的解多樣性。在資源分配完成後，每位搜尋者將會用自身所攜帶的樣品去與區域內的解進行解資訊交換，本範例所使用的資訊交換方式為單點交配方法，如圖 3.11 中的遠見搜尋階段，每位搜尋者與每個區域的樣品將各產生兩個超參數解，每組解在進行單點交配後將根據突變機率 (mutation rate) 隨機進行突變，如圖中左下角，每段解中的綠色區塊為突變後所產生的超參數，該區塊的值將以隨機於解空間限制內取值方式產生，並根據該區域所定義的解特色進行強制修改，如圖中解的淡橘色區塊，將根據搜尋者前往何者區域進行投資時決定要賦予哪區解空間的特色解。在

解資訊交換完成後，將進行各新解的適應值計畫階段，將根據各超參數解內容進行深度神經網路建立與訓練，待訓練完成後將根據適應值計算出各搜尋者對於各區域該迭代的期望值，其中 $e_{1,1}$ 代表第 1 位搜尋者對於第 1 區域的期望值、 $e_{1,2}$ 、 $e_{2,1}$ 及 $e_{2,2}$ 亦然。計算完搜尋者對於每個區域的期望值後，將進行下一迭代的各搜尋者的投資區域選擇與各區域中的樣品替換，如該範例中兩位搜尋者 s_1 及 s_2 都選擇第一區域 r_1 作為下一次的投資目標。最後將於市場調查中更新各區域所被投資、未被投資的次數以用於計算下一迭代的期望值，並使許久未被投資的區域可以增加下次被投資的機率。

3.4 總結

在本章節中，首先提到了本論文使用集成式架構應用異常用電偵測動機並分別介紹了三個集成式系統中的模型架構及其原理，透過不同特徵觀點的模型，可以根據不同資料中不同的關鍵特徵給出自己的預測結果，輸入至用於動態調整各模型給予的預測結果權重的元學習器，以深度神經網路作為架構的元學習器可以有效的學習在各種預測結果下，分配適當的權重給各模型，並有效的提升分類結果。然而深度神經網路的神經元層數、神經元數量及訓練框架下的學習率與批次大小也大大的影響了權重分配的效能，針對此議題，本論文採用搜尋經濟學演算法作為最佳化深度神經網路架構的方法，搜尋經濟學演算法相較於其他傳統的超啟發式演算法多了跳脫區域最佳解的機制，期中的區域期望值將以解空間中許久未被搜尋的次數作為參考，提高演算法回頭搜尋該解空間區域的機率，藉此有機會找到搜尋解過程中的盲點並發現更優良的超參數組合。

第四章 實驗結果

4.1 實驗環境與參數設定

本論文進行所有實驗所使用的程式語言為 Python 3.7.12，程式撰寫環境為 Ubuntu 20.04.3 LTS。本研究所使用的深度學習環境為 PyTorch 1.9.0 及 cuda 11.1，詳細環境如表 4.1。

表 4.1: 實驗硬體環境

	個人電腦
作業系統	Ubuntu 20.04.3 LTS
處理器	AMD Ryzen™ 7 5800X 3.80GHz
顯示卡	Nvidia Geforce RTX 3080 (12 GB)
記憶體	32 GB

4.2 資料集

所有的實驗及模型訓練均使用 Zheng 等人釋出的資料集 [Zheng et al., 2017] 所釋出之資料集，該資料集中包含 42,372 筆含有 1,034 天的用戶電力流量數據，在 42,372 用戶的用電紀錄中有 3,615 戶具有竊電行為，而在本章節實驗結果中，資料集均進行比率為 60、20、20 的隨機切割，其意義為 60% 作為訓練集，20% 作為訓練過程中的驗證集與 20% 作為最後的模型效能測試集。

實際用電資料如圖 4.1 所示，圖 (a) 為資料集中正常用電用戶在 1,034 天中的用電量變化，而圖 (b) 為異常用電戶在 1,034 天的用電量變化，該二圖也可以明確地展示本論文所使用之資料集所含有的問題，在圖 (a) 的約前 200 天可能由於線路故障、電表異常或者人為因素導致用電資料遺失，造成許多資料缺失值 (missing data)。而從圖 (b) 可

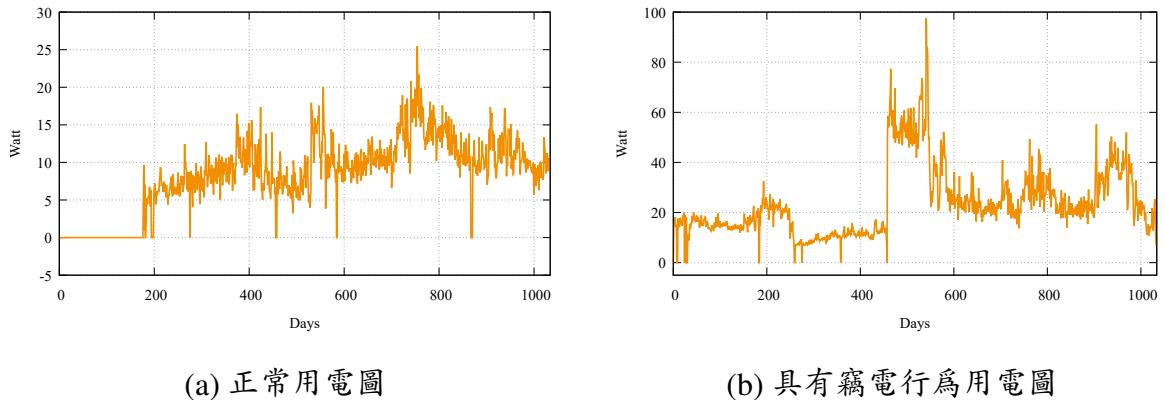


圖 4.1: 正常與異常資料比較

以看出約在 400 至 500 天時用電量每天都有不同於以往的用電量變化，該特徵需使用深度學習模型進行長時間或者短時間的特徵變化並加以分辨輸入用電量資料為正常或異常用戶。而綜觀兩張用電量變化圖，可以發現兩用戶的縱軸尺度不同，即最大用電量和最小用電量範圍每戶皆不盡相同，該尺度差可能造成正規化或者模型擷取用電趨勢變化的困難。, 本論文實驗主要訓練目標為使分類模型能夠盡量的分類出這些竊電用戶並盡量減少正常用戶的誤判比例。小節 4.2 將詳細說明本論文用於解決上述問題的方法及資料前處理方式。

4.2.1 資料清理

在本論文的資料清理方法中可以分為資料排序及分位數轉換，該電力原始資料集的用電量原為未經日期排序的時序型資料，而時序型資料中最為重要的判斷依據為在每個時間點不同的用電量變化特徵，更可以分為長時間的週期性用電變化與短時間突如其來的用電需求變化，若資料集的用電量排序有誤，前後的用電量交錯，將導致上述的用電量變化特徵變得難以擷取，進而導致分類效能大幅度降低。因此本論文在資料前處理的開始，會先將資料集依用電日期進行排序，整理好用電量之間的先後關係。

除了用電量未排序外，不同用戶之間導致的用電差也形成較棘手的資料分布型態，如圖 4.2 中的圖 (a)，由於每戶之間的用電基本需求與習慣不同，這種每戶的最大用電量與最小用電量不同的現象，也導致了資料集中的離群值 (outlier)，例如一般家庭的用平均用電量約為 20 至 40 度 (千瓦) 不等，但若為店家或工廠的用電需求，每日可能為 100 至 1000 度 (千瓦) 以上，因此若家庭的戶數明顯高於店家或工廠，店家或工廠的用電資料在整個資料集中就會變成離群值。針對離群值問題，本論文使用分位數轉

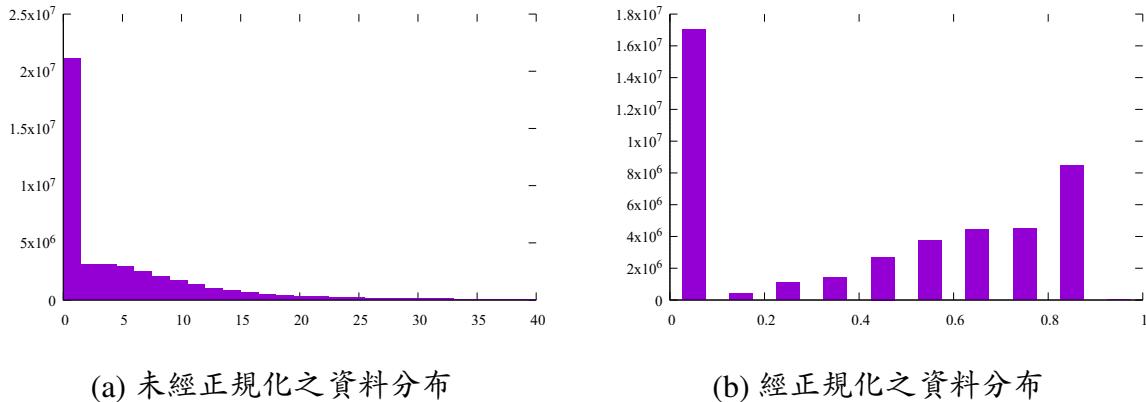


圖 4.2: 使用分位數轉換法進行資料正規化

換法 (quantile transformation) 對資料集進行正規化，如圖 4.2 中的圖 (b)，分位數轉換法用非線性的方式將資料映射到資料集中較常出現的高斯分布中，除了可以將資料均縮限於 0 到 1 之間，進以提升分類演算的準確性，該方法也非常適合於處理有離群值問題的資料集。

4.2.2 資料摺疊

為了使一維的時序型用電量資料能被二維卷積方式所擷取特徵，本論文將時序型用電量資料折疊成二維資料，如 圖 4.3 所示，二維資料除了便於使卷積神經網路的二維卷積核所擷取特徵外，還可以根據不同的時間間隔以更凸顯資料的週期性變化並讓擅長於擷取相鄰區域的卷積神經網路學習到自定義週期之間的變化特徵，該方法於近期的許多文獻 [Chen et al., 2018, Feng et al., 2020, Chen et al., 2018, Feng et al., 2020] 中顯示能明顯強化模型分類效能。電力資料中的時序特徵分為兩個種類：局部連貫性與週期性。局部連貫性是在發生與以往不同的電力數值時，通常會大幅度的影響接續幾個時間點的電力數值，例如晚上 6 點突然的電力需求可能導致接下來 7 點至 10 點的電力異常需求；週期性則是以每週或每天為週期，各時間點相似的電力固定變化趨勢。該模型將時序電力資料摺疊為二維矩陣形式，並使用進行上述兩種時序特徵的擷取。並輸入至全連接層進行未來的電力預測。該文獻的實驗結果中也顯示，當作者以 10 分鐘、15 分鐘、30 分鐘及 60 分鐘的時間間隔摺疊時序電力資料，會對於預測結果會產生不同的影響，同時也證明了比起不使用摺疊方法處理資料集輸入至機器學習或者深度學習的模型進行預測，使用摺疊後的資料可以取得更好的預測結果。

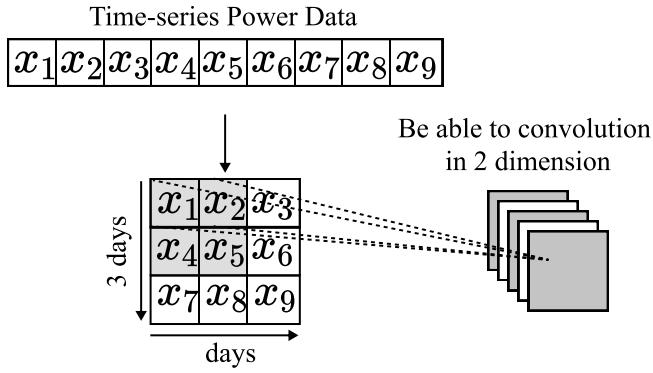


圖 4.3: 資料摺疊示意圖

4.2.3 缺失值通道

本論文使用的用電量資料集中，可能因為設備故障，線路毀損或人為因素導致了大量的資料缺失值，這些缺失值可能發生在 1,034 天用電資料的任何一小段時間並造成該部分缺失值的大量資訊遺失，這些資訊遺失可能會導致深度學習模型無法正確地擷取真實的用電量變化特徵並學習錯誤而使最後分類結果的效能下降，針對本論文使用的資料集含有約 25% 的資料缺失，將使用兩部分的缺失值處理方法來最大化減少資訊遺失所帶來的效能下降問題。如圖 ?? 所示，首先將缺失值部分用 0 進行取代，並建立一個缺失值特徵通道，來使深度學習模型學習缺失值的相關資訊。其主要理念為利用缺失值特徵來作為資料的特徵，用以協助模型判斷該筆資料是否為異常資料，換而言之，本論文使用建立缺失值特徵通道的方式以化危機為轉機，原本丟失的將近 25% 的特徵資訊，轉換為凸顯該筆資料的缺失特徵，深度學習模型可以根據不同的缺失值狀況做為參考進行判斷，例如異常用戶經常於前 200 天有大量的用電量缺失值，模型將學習到這個習慣，並用以判斷後續資料，其中使用 0 作為填補原始資料缺失值能更加凸顯該段資料的差異性，搭配缺失值通道能取得更好的效果。

4.3 實驗結果及分析

在本小節中將測試並比較其他文獻方法與各種前處理方式、搜尋經濟學演算法中的區域切割方式對於本論文實驗結果的影響性。比較主要對象包含：較為傳統的一維卷積神經網路 [LeCun et al., 1995]、長短期記憶網路 [Hochreiter and Schmidhuber, 1997]，及近期用於異常用電偵測方法：Zheng 等人提出的寬且深的卷積神經網路 [Zheng et al., 2017]、Bello

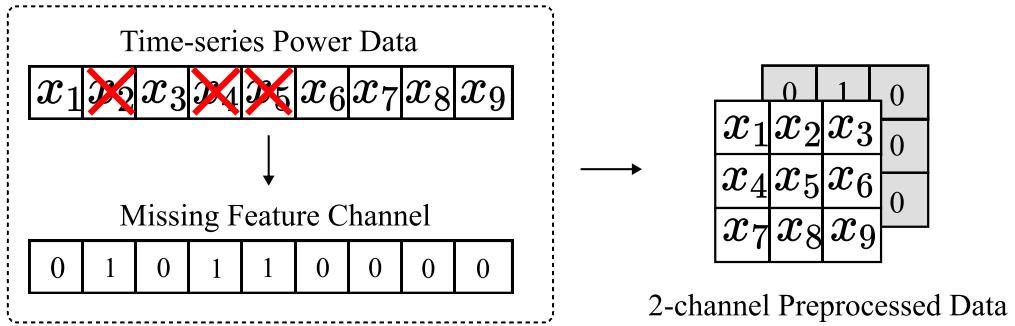


圖 4.4: 缺失值特徵通道示例

等人提出的注意力機制網路 [Bello et al., 2019] 及 Finardi 等人提出的多頭注意力卷積神經網路 [Finardi et al., 2020]。由於本論文所使用的用電量資料集具有資料不平衡問題，若單純使用準確率作為模型的評斷依據將不夠明確，因此在本論文之實驗結果中，將以 ROC 曲線下面積 (ROC-AUC)、精準-召回曲線下面積 (PR-AUC) 及 F1 得分等較針對異常資料的指標作為分類模型的評判依據。F1 得分的計算方式如公式 4.1：

$$F1\text{-score} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}, \quad (4.1)$$

其中精準度的計算方式如公式 4.2，True Positive (TP) 與 False Positive (FP) 可參考表 4.2 混淆矩陣，精準度代表的意義為在所有的模型判斷為異常的資料中，有多少是真的異常的資料，

表 4.2: 混淆矩陣

	Predicted Positive	Predicted Negative
Actual Positive	TN	FP
Actual Negative	FN	TP

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (4.2)$$

而召回率的定義如 公式 4.3，其中 True Positive (TP) 與 False Negative (FN) 同樣參考於表 4.2 混淆矩陣，其代表在所有異常的資料中，模型實際判斷出的異常資料在資料集中所有異常資料比率為何，

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (4.3)$$

透過精準度與召回率，可以得知 F1-score 即為該兩個分類效能指標的調和平均數 (harmonic mean)，能作為判斷模型對於異常資料的判斷能力，並尤其適合於資料不平衡的資料集中。

表 4.3: 實驗結果比較圖

Classification Algorithm	ROC-AUC	PR-AUC	F1-score	Accuracy
LSTM [Hochreiter and Schmidhuber, 1997]	85.82%	50.96%	48.39%	92.76%
1D-CNN [LeCun et al., 1995]	89.21%	62.64%	55.57%	93.99%
MHA-CNN [Finardi et al., 2020]	91.53%	65.50%	54.84%	93.99%
AA-CNN [Bello et al., 2019]	88.10%	58.91%	48.83%	93.56%
W-DCNN [Zheng et al., 2017]	86.52%	51.84%	47.18%	92.76%
Proposed Method	91.62%	67.66%	56.56%	94.30%

本論文提出之模型與其他論文使用方法比較結果如表 4.3 所示，從表中可以看到集成式分類系統的效能明顯優於傳統的一維卷積神經網路、長短期記憶網路及近期提出用於異常用電偵測方法，該結果也進一步驗證，集成式分類系統相較於使用單一模型進行分類，可以透過不同特徵觀點的模型提升整體效能，從圖 4.5 中的精準召回曲線也可以看出，集成式分類系統在所有的分類方法中涵蓋了最大的面積，也就是說在各種不同的閾值 (threshold) 下本論文提出之分類方法能取得最佳的分類結果。

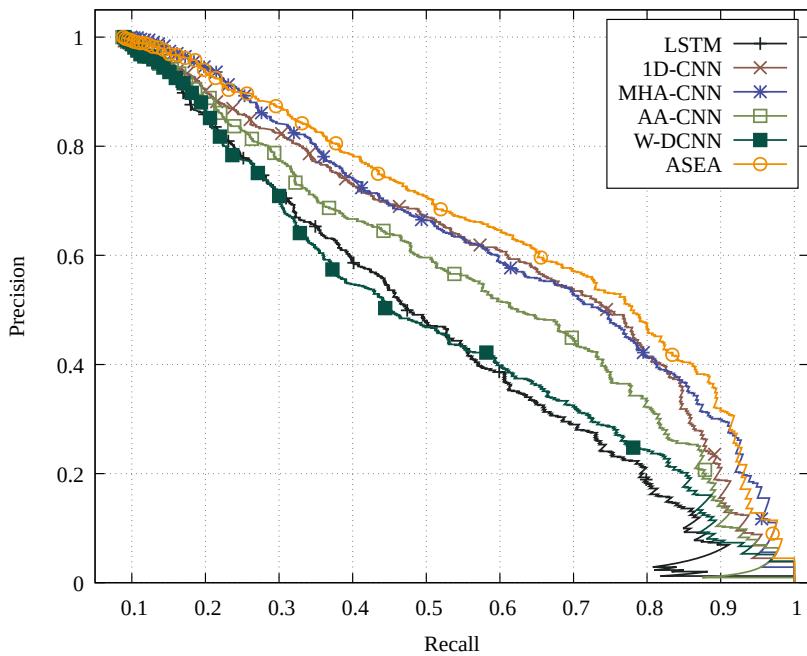


圖 4.5: 各模型之精準召回曲線下面積

深度神經網路的超參數最佳化如圖 4.6，該實驗比較了傳統經典的基因演算法與粒子群最佳化，圖中的收斂曲線為在相同的子模型參數下，進行五次的深度神經網路超參數最佳化結果平均，從收斂圖中可以看出，搜尋經濟學演算法找到的解在精準召回曲線下面積，優於其他兩種經典的超啟發演算法。

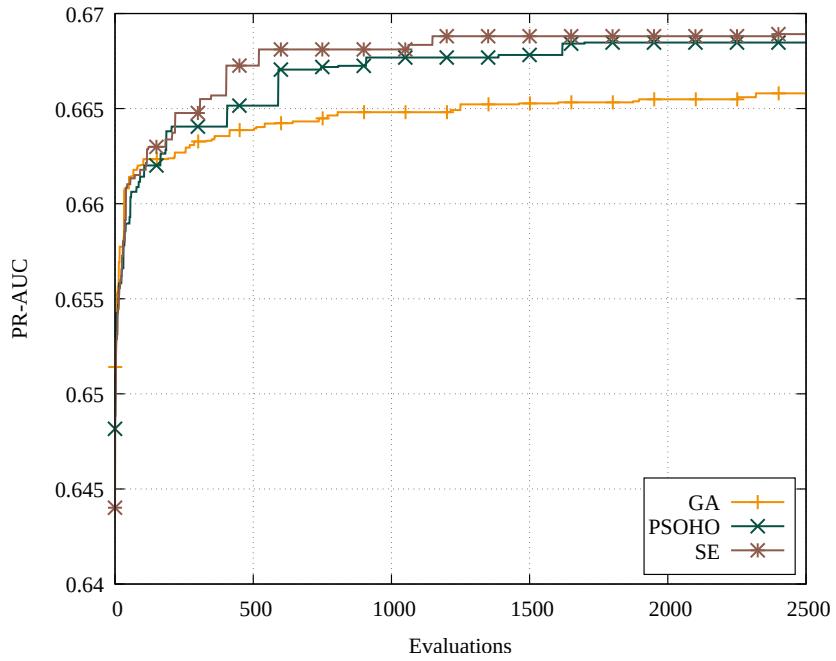


圖 4.6: 各演算法於超參數最佳化效能比較

4.3.1 資料切割比例實驗

資料前處理對於深度學習的分類模型結果有著重大的影響，本論文所使用的資料集中包含了離群值、資料未排序及資料缺失等問題，在小節 4.1 中也清楚地說明了本論文針對這些資料集問題的應對方式，本小節將以本論文所使用的資料前處理方式，進行資料比例切割、資料排序、資料折疊及分位數正規化前後的結果比較與分析。表 4.4、表 4.5 及表 4.6 分別展示了訓練集與測試集根據 8:2、6:4、5:5 進行資料集的切割後，輸入各模型進行訓練並驗證結果，其中在每種切割比例組合下，訓練集當中的 20% 會用於驗證集，從該三表可以看出，在本論文使用的資料集下，訓練集的多寡直接的影響到模型的分類結果且明顯成正比。由此可知資料集中的資料具備的資料獨特性，大多的資料均有其值得模型學習的特徵，導致在資料量越大時，越多的資訊量會使分類結果更加準確。

4.3.2 子模型組合實驗

本小節將針對集成式架構中，集成不同分類模型組合的結果進行實驗與分析，表 4.7 為在本論文主要使用的三個子模型所有組合的訓練結果，從表中可以發現當使用兩個模型時，測試結果均優於僅使用單一模型，其中一維卷積神經網路與多頭注意卷

表 4.4: 以 5:5 資料比例切割

Classification Algorithm	ROC-AUC	PR-AUC	F1-score	Accuracy
LSTM [Hochreiter and Schmidhuber, 1997]	81.17%	40.20%	36.14%	92.19%
1D-CNN [LeCun et al., 1995]	85.26%	50.79%	47.81%	93.23%
MHA-CNN [Finardi et al., 2020]	87.69%	54.30%	47.13%	93.33%
AA-CNN [Bello et al., 2019]	84.67%	48.98%	42.60%	93.13%
W-DCNN [Zheng et al., 2017]	82.34%	43.90%	41.46%	92.19%
Proposed Method	87.90%	56.31%	47.70%	93.67%

表 4.5: 以 6:4 資料比例切割

Classification Algorithm	ROC-AUC	PR-AUC	F1-score	Accuracy
LSTM [Hochreiter and Schmidhuber, 1997]	83.90%	46.20%	43.49%	92.72%
1D-CNN [LeCun et al., 1995]	87.68%	55.66%	50.36%	93.58%
MHA-CNN [Finardi et al., 2020]	89.04%	58.21%	50.17%	93.71%
AA-CNN [Bello et al., 2019]	86.78%	53.66%	43.71%	93.36%
W-DCNN [Zheng et al., 2017]	85.00%	47.96%	44.09%	92.67%
Proposed Method	89.75%	60.82%	52.14%	94.00%

積神經網路對於分類結果的影響較大。當三個子模型均加入集成時，可以達到最佳效能，相較於次優的一維卷積神經網路與多頭注意卷積神經網路組合，ROC 曲線下面積、精準召回曲線下面積、F1-score，均可以再次提升，也證明資料集中有長短期記憶模型所擅長分類的資料，以補足其他兩個模型的特徵盲點，達到更好的分類效能。雖然三個模型的準確並非最優秀的，但於異常偵測任務中，主要目標為：1.最大化異常資料中被模型偵測到的比例，2.最小化正常資料中被模型誤判的比率，這兩項所代表的指標精準度與召回率可參考小節 4.3，其調和平均數 F1-score 本論文認為其更能客觀的表示與評斷模型的異常偵測能力。

表 4.6: 以 8:2 資料比例切割

Classification Algorithm	ROC-AUC	PR-AUC	F1-score	Accuracy
LSTM [Hochreiter and Schmidhuber, 1997]	85.33%	50.71%	49.50%	92.99%
1D-CNN [LeCun et al., 1995]	89.33%	62.05%	54.11%	93.94%
MHA-CNN [Finardi et al., 2020]	91.12%	64.72%	55.11%	93.85%
AA-CNN [Bello et al., 2019]	88.10%	58.91%	48.83%	93.56%
W-DCNN [Zheng et al., 2017]	86.52%	51.84%	47.18%	92.76%
Proposed Method	91.62%	67.27%	58.73%	94.07%

表 4.7: 實驗結果比較圖

Classification Algorithm	ROC-AUC	PR-AUC	F1-score	Accuracy
LSTM	85.85%	51.38%	47.72%	92.78%
1D-CNN	89.32%	62.05%	54.11%	93.94%
MHA-CNN	91.53%	65.50%	54.84%	94.09%
LSTM+1D-CNN	90.09%	63.91%	55.84%	79.56%
LSTM+MHA-CNN	91.38%	65.82%	56.21%	93.94%
1D-CNN+MHA-CNN	91.54%	67.20%	57.34%	94.21%
LSTM+1D-CNN+MHA-CNN	91.63%	67.25%	58.53%	94.06%

4.3.3 區域比較實驗

在本小節的實驗中，將探討超參數最佳化編碼中以不同超參數作為搜尋經濟學演算法的區域切割依據及不同的區域數量進行實驗，比較且分析其中的差異，對於搜尋經濟學演算法來說，區域的劃分方式與數量會對演算法搜尋的效率與效果造成相當的影響，如圖 4.7 中，比較搜尋經濟學演算法中以學習率做為切割依據，劃分不同數量及其降低突變率至 50% 的效果，從收斂結果可以看出當在本論文所使用的深度神經網路之上，以學習率作為切割依據並將區域數設為 2 可以達到搜尋解的最佳效能。

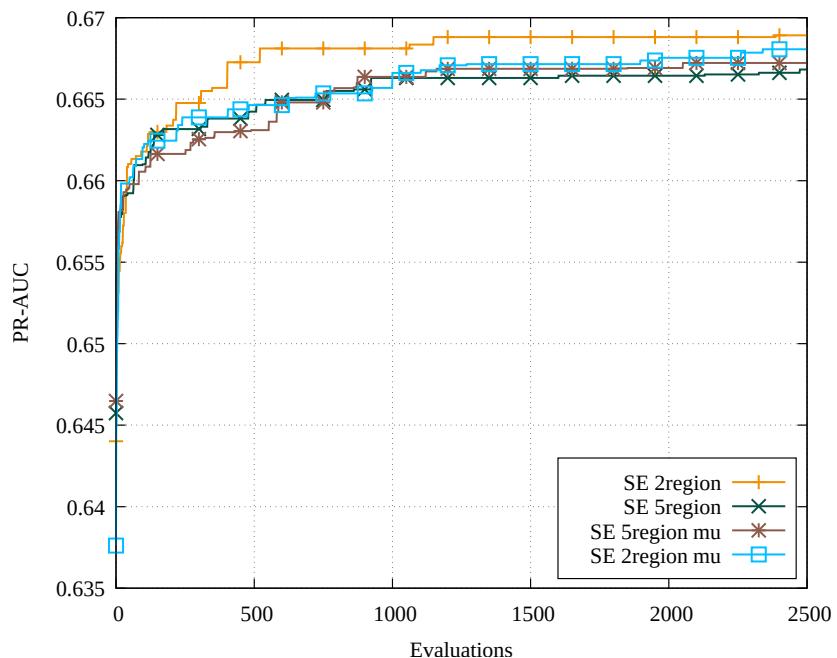


圖 4.7: 不同區域數量與突變率效能差異

第五章 結論與未來展望

5.1 結論

本論文提出了一種基於集成式學習的異常用電偵測系統，該系統主要分為兩大貢獻，第一為透過集成式學習中的堆疊法，可以將具有不同特徵觀點的模型集成於一個模型，並透過深度神經網路學習應該如何在沒有各模型的能力先備知識下，分配各模型給出預測結果之權重。該動態的權重分配也不同於傳統的使用固定公式或者以平均的方式計算各模型的權重組成，可以根據不同的輸入資料，得到各模型輸出該筆資料為正常及異常的機率，深度神經網路將可以以該機率作為訓練網路的資料，進行動態的權重分配訓練。該架構也於實驗結果小節 4.3 中顯示了確實能提升整體的分類效能。為最佳化使動態權重分配，本論文以超啓發式演算法中的搜尋經濟學演算法作為最佳化系統的框架，並設計編碼與該演算法獨有的區域切割方式，實驗結果小節 4.3.3 也實驗了不同的區域切割方式將導致不同的動態權重分配效果。

5.2 未來展望

本論文提出了一種集成式的異常用電偵測系統並基於搜尋經濟學演算法最佳化架構中用於分配各子模型權重值的深度神經網路，該分類架構已於實驗結果中驗證其優於近期相似文獻的方法，在研究過程中也發現在未來可以發展或改良的部分：

1. 在超參數組合的搜尋過程中，常常面臨需要進行上百次甚至上千次的模型驗證，每次驗證均為一輪完整的模型訓練，在整個超參數搜尋經常花費較長的時間。在後續的研究中可針對模型驗證速度進行嘗試。
2. 本論文的編碼以深度神經網路中的隱藏層層數、各層內的神經元數、批次大小及學習率作為一個可行解，為利於各迭代的資訊交換，採用的編碼為固定長度

解，導致若隱藏層的層數小於解中可代表的各層神經元數，後續若突變到未使用的神經元數部分解，將可能使該部分解對於模型驗證無任何影響進而浪費一次模型驗證機會。

在搜尋解的過程中，若模型驗證於未來獲得加速，演算法在相同的時間內可進行更多迭代的超參數搜尋，或許也能嘗試加入不同的超參數種類進行搜尋，例如需要加入激勵函數層與否、丟棄層的丟棄比率等，該改動能使超參數最佳化的精度提高，並有望再次提升整體分類結果。

參考文獻

- [Alvear-Sandoval and Figueiras-Vidal, 2018] Alvear-Sandoval, R. F. and Figueiras-Vidal, A. R. (2018). On building ensembles of stacked denoising auto-encoding classifiers and their further improvement. *Information Fusion*, 39(1):41–52.
- [Andonie and Florea, 2020] Andonie, R. and Florea, A.-C. (2020). Weighted random search for CNN hyperparameter optimization. *arXiv preprint arXiv:2003.13300*.
- [Antmann, 2009] Antmann, P. (2009). Reducing technical and non-technical losses in the power sector. Technical report.
- [Ayan et al., 2020] Ayan, E., Erbay, H., and Varçın, F. (2020). Crop pest classification with a genetic algorithm-based weighted ensemble of deep convolutional neural networks. *Computers and Electronics in Agriculture*, 179(1):105809–105819.
- [Aydin and Gungor, 2018] Aydin, Z. and Gungor, V. C. (2018). A novel feature design and stacking approach for non-technical electricity loss detection. In *Proceedings of IEEE Innovative Smart Grid Technologies-Asia*, pages 867–872.
- [Aziz et al., 2020] Aziz, S., Naqvi, S. Z. H., Khan, M. U., and Aslam, T. (2020). Electricity theft detection using empirical mode decomposition and K-Nearest neighbors. In *Proceedings of International Conference on Emerging Trends in Smart Technologies*, pages 1–5.
- [Bello et al., 2019] Bello, I., Zoph, B., Vaswani, A., Shlens, J., and Le, Q. V. (2019). Attention augmented convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3286–3295.
- [Bergstra and Bengio, 2012] Bergstra, J. and Bengio, Y. (2012). Random search for hyperparameter optimization. *Journal of Machine Learning Research*, 13(2):1–25.

- [Breiman, 1996] Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.
- [Breiman, 2001] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.
- [Brunese et al., 2020] Brunese, L., Mercaldo, F., Reginelli, A., and Santone, A. (2020). An ensemble learning approach for brain cancer detection exploiting radiomic features. *Computer Methods and Programs in Biomedicine*, 185(1):105150.
- [Bühlmann and Yu, 2002] Bühlmann, P. and Yu, B. (2002). Analyzing bagging. *The Annals of Statistics*, 30(4):927–961.
- [Buzau et al., 2019] Buzau, M.-M., Tejedor-Aguilera, J., Cruz-Romero, P., and Gómez-Expósito, A. (2019). Hybrid deep neural networks for detection of non-technical losses in electricity smart meters. *IEEE Transactions on Power Systems*, 35(2):1254–1263.
- [Chen et al., 2018] Chen, M., Yu, X., and Liu, Y. (2018). PCNN: Deep convolutional networks for short-term traffic congestion prediction. *IEEE Transactions on Intelligent Transportation Systems*, 19(11):3550–3559.
- [Dasarathy and Sheela, 1979] Dasarathy, B. V. and Sheela, B. V. (1979). A composite classifier system design: Concepts and methodology. *Proceedings of the IEEE*, 67(5):708–713.
- [Feng et al., 2020] Feng, X., Hui, H., Liang, Z., Guo, W., Que, H., Feng, H., Yao, Y., Ye, C., and Ding, Y. (2020). A novel electricity theft detection scheme based on text convolutional neural networks. *Energies*, 13(21):5758–5775.
- [Finardi et al., 2020] Finardi, P., Campiotti, I., Plensack, G., de Souza, R. D., Nogueira, R., Pinheiro, G., and Lotufo, R. (2020). Electricity theft detection with self-attention. *arXiv preprint arXiv:2002.06219*.
- [Freund et al., 1996] Freund, Y., Schapire, R. E., et al. (1996). Experiments with a new boosting algorithm. In *Proceedings of International Conference on Machine Learning*, volume 96, pages 148–156.
- [Glorot et al., 2011] Glorot, X., Bordes, A., and Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of International Conference on Artificial Intelligence and Statistics*, pages 315–323.

- [Ha et al., 2005] Ha, K., Cho, S., and MacLachlan, D. (2005). Response models based on bagging neural networks. *Journal of Interactive Marketing*, 19(1):17–30.
- [Hansen and Salamon, 1990] Hansen, L. K. and Salamon, P. (1990). Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10):993–1001.
- [Hasan et al., 2019] Hasan, M., Toma, R. N., Nahid, A.-A., Islam, M., and Kim, J.-M. (2019). Electricity theft detection in smart grid systems: A CNN-LSTM based approach. *Energies*, 12(17):3310.
- [He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of IEEE International Conference on Computer Vision*, pages 1026–1034.
- [Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.
- [Holland, 1992] Holland, J. H. (1992). Genetic algorithms. *Scientific American*, 267(1):66–73.
- [Kennedy and Eberhart, 1995] Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of International Conference on Neural Networks*, volume 4, pages 1942–1948.
- [Khraisat et al., 2020] Khraisat, A., Gondal, I., Vamplew, P., Kamruzzaman, J., and Alazab, A. (2020). Hybrid intrusion detection system based on the stacking ensemble of c5 decision tree classifier and one class support vector machine. *Electronics*, 9(1):173–191.
- [Kim et al., 2002] Kim, H.-C., Pang, S., Je, H.-M., Kim, D., and Bang, S.-Y. (2002). Support vector machine ensemble with bagging. In *Proceedings of International Workshop on Support Vector Machines*, pages 397–408.
- [Klatzer and Pock, 2015] Klatzer, T. and Pock, T. (2015). Continuous hyper-parameter learning for support vector machines. In *Proceedings of Computer Vision Winter Workshop*, pages 39–47.

- [LeCun et al., 1995] LeCun, Y., Bengio, Y., et al. (1995). Convolutional networks for images, speech, and time series. *The handbook of Brain Theory and Neural Networks*, 3361(10):1995–2009.
- [León et al., 2011] León, C., Biscarri, F., Monedero, I., Guerrero, J. I., Biscarri, J., and Millán, R. (2011). Variability and trend-based generalized rule induction model to NTL detection in power companies. *IEEE Transactions on Power Systems*, 26(4):1798–1807.
- [Li et al., 2017] Li, L., Jamieson, K. G., DeSalvo, G., Rostamizadeh, A., and Talwalkar, A. (2017). Hyperband: Bandit-based configuration evaluation for hyperparameter optimization. In *Proceedings of International Conference on Learning Representations*, pages 1–15.
- [Li et al., 2019] Li, S., Han, Y., Yao, X., Yingchen, S., Wang, J., and Zhao, Q. (2019). Electricity theft detection in power grids with deep learning and random forests. *Journal of Electrical and Computer Engineering*, 2019(1):1–11.
- [Littlestone and Warmuth, 1994] Littlestone, N. and Warmuth, M. K. (1994). The weighted majority algorithm. *Information and Computation*, 108(2):212–261.
- [Livieris et al., 2020] Livieris, I. E., Pintelas, E., Stavroyiannis, S., and Pintelas, P. (2020). Ensemble deep learning models for forecasting cryptocurrency time-series. *Algorithms*, 13(5):121–142.
- [Moghaddass and Wang, 2017] Moghaddass, R. and Wang, J. (2017). A hierarchical framework for smart grid anomaly detection using large-scale smart meter data. *IEEE Transactions on Smart Grid*, 9(6):5820–5830.
- [Mohammed et al., 2021] Mohammed, M., Mwambi, H., Mboya, I. B., Elbashir, M. K., and Omolo, B. (2021). A stacking ensemble deep learning approach to cancer type classification based on TCGA data. *Scientific Reports*, 11(1):1–22.
- [Ouyang et al., 2018] Ouyang, Z., Sun, X., Chen, J., Yue, D., and Zhang, T. (2018). Multi-view stacking ensemble for power consumption anomaly detection in the context of industrial internet of things. *IEEE Access*, 6(1):9623–9631.

- [Rahman et al., 2019] Rahman, M. M., Bhuiyan, M. I. H., and Das, A. B. (2019). Classification of focal and non-focal eeg signals in VMD-DWT domain using ensemble stacking. *Biomedical Signal Processing and Control*, 50(1):72–82.
- [Sagi and Rokach, 2018] Sagi, O. and Rokach, L. (2018). Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1249.
- [Schapire, 1990] Schapire, R. E. (1990). The strength of weak learnability. *Machine Learning*, 5(2):197–227.
- [Serizawa and Fujita, 2020] Serizawa, T. and Fujita, H. (2020). Optimization of convolutional neural network using the linearly decreasing weight particle swarm optimization. *arXiv preprint arXiv:2001.05670*.
- [Shekar and Dagnew, 2019] Shekar, B. and Dagnew, G. (2019). Grid search-based hyperparameter tuning and classification of microarray cancer data. In *Proceedings of International Conference on Advanced Computational and Communication Paradigms*, pages 1–8.
- [Snoek et al., 2012] Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, 25(1):1–9.
- [Soon et al., 2018] Soon, F. C., Khaw, H. Y., Chuah, J. H., and Kanesan, J. (2018). Hyperparameters optimisation of deep CNN architecture for vehicle logo recognition. *IET Intelligent Transport Systems*, 12(8):939–946.
- [Tsai, 2015] Tsai, C.-W. (2015). Search economics: A solution space and computing resource aware search method. In *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, pages 2555–2560.
- [Tsai and Fang, 2021] Tsai, C.-W. and Fang, Z.-Y. (2021). An effective hyperparameter optimization algorithm for DNN to predict passengers at a metro station. *ACM Transactions on Internet Technology*, 21(2):1–24.
- [Tsai et al., 2020] Tsai, C.-W., Hsia, C.-H., Yang, S.-J., Liu, S.-J., and Fang, Z.-Y. (2020). Optimizing hyperparameters of deep learning in predicting bus passengers based on simulated annealing. *Applied Soft Computing*, 88:106068.

- [Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30(1):1–11.
- [Wang et al., 2017] Wang, Q., Luo, Z., Huang, J., Feng, Y., and Liu, Z. (2017). A novel ensemble method for imbalanced data learning: bagging of extrapolation-SMOTE SVM. *Computational Intelligence and Neuroscience*, 2017(1):1–11.
- [Wolpert, 1992] Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*, 5(2):241–259.
- [Xiao et al., 2018] Xiao, Y., Wu, J., Lin, Z., and Zhao, X. (2018). A deep learning-based multi-model ensemble method for cancer prediction. *Computer Methods and Programs in Biomedicine*, 153(1):1–9.
- [Yang and Shami, 2020] Yang, L. and Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, 415(1):295–316.
- [Yang et al., 2021] Yang, Y., Wei, L., Hu, Y., Wu, Y., Hu, L., and Nie, S. (2021). Classification of Parkinson’s disease based on multi-modal features and stacking ensemble learning. *Journal of Neuroscience Methods*, 350:109019–109029.
- [Yi and Bui, 2020] Yi, H. and Bui, K.-H. N. (2020). An automated hyperparameter search-based deep learning model for highway traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, 22(9):5486–5495.
- [Yi and Khac-Hoai, 2020] Yi, H. and Khac-Hoai, N. B. (2020). An automated hyperparameter search-based deep learning model for highway traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, 22(9):5486–5495.
- [Young et al., 2015] Young, S. R., Rose, D. C., Karnowski, T. P., Lim, S.-H., and Patton, R. M. (2015). Optimizing deep learning hyper-parameters through an evolutionary algorithm. In *Proceedings of Workshop on Machine Learning in High-performance Computing Environments*, pages 1–5.

- [Zhang et al., 2021] Zhang, S., Chen, Y., Zhang, W., and Feng, R. (2021). A novel ensemble deep learning model with dynamic error correction and multi-objective ensemble pruning for time series forecasting. *Information Sciences*, 544(1):427–445.
- [Zheng et al., 2017] Zheng, Z., Yang, Y., Niu, X., Dai, H.-N., and Zhou, Y. (2017). Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids. *IEEE Transactions on Industrial Informatics*, 14(4):1606–1615.