**（一） 計畫之背景及目的**

1. **計畫背景**

近年來，雲端運算技術成為了現在應用程式供應商提供服務的媒介。許多企業集成運算資源，將應用程式布署在各自的資料中心，虛擬化技術可以彈性分配資源，並以網際網路為使用者提供服務。資源的集成在管理上大幅度的減少應用的維護成本。資料中心結合了多台的伺服器運算，每當伺服器崩潰或停機時總會造成困擾。應用服務主導著部份主機的資源分配，服務品質與硬體資源使用量息息相關。因此，若能使用硬體資源監測工具即時蒐集資料，分析出故障的關鍵點並建立在線預測模型，實施有效的預測性維護。

本計畫與台灣恩智浦半導體股份有限公司(NXP)合作進行兩年期的改善方案。首先該公司使用Zabbix Server監控資料中心資源使用率，雖然Zabbix Server能在使用率高於70%時發出警告訊息，但此方案缺乏預判性，且無法得知實際的異常原因，因此計畫第一年將分析Zabbix Server監控之資源使用率資料，針對異常情境進行故障檢測與分類，並使用深度學習技術建立在線預測模型提早預測故障；其次該公司的資料中心屬於多機環境，大量資料流會造成預測延遲，且在線預測模型需要頻繁更新以維持預測準確度。所以第二年計畫將使用TensorflowOnSpark部屬在線預測模型，實現大量資料下的分布式預測，並建立AutoML將模型訓練流程簡化，方便管理人員能快速更新在線預測模型。

本計畫「伺服器異常預測及分布式伺服器異常預測系統」(如圖1所示)規劃以兩年期程解決恩智浦公司在生產過程所遭遇的問題，逐年分別提出(1)基於深度學習檢測晶片輪廓分析與智動化系統、(2)機器手臂的運行速度優化與智慧化健康管理系統。

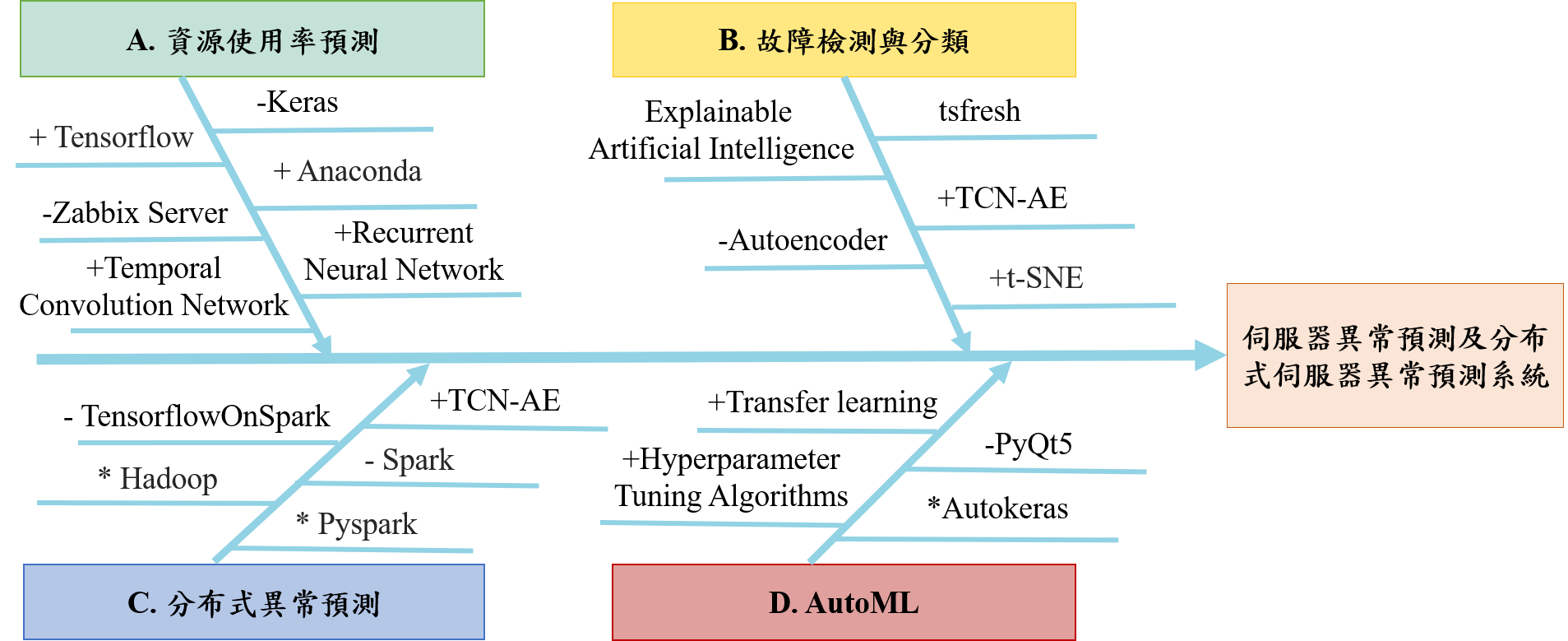


圖1.本案之技術架構

**第一年期程**

恩智浦公司的雲端伺服器遭遇兩項問題需要解決(1)目前的雲端服務上僅有Zabbix Server作為監控工具，Zabbix Server會在資源使用率超過70%時發出警報。此機制無法準確檢測出異常，誤報率高使得管理人員時常忽略此警告訊息，導致系統發生停機時無法即時反應。(2)異常的檢測方法缺乏預判性，容易在故障發生當下無法即時修復。因此，在第一年期程中恩智浦公司希望能改善雲端伺服器的異常檢測機制。

本計畫的目的是針對上述兩項雲端伺服器的問題，我們提出了兩階段異常預測的解決方案。第一階段將會蒐集Zabbix Server監測的CPU使用率、Memory使用率、Disk I/O等資料作為樣本，並使用深度學習方法預測時間序列。當系統持續運行時，第一階段模型會提前預測系統可能的使用率。第二階段會將歷史資料與預測資料合併，並針對該區間的長時間資料作完整的異常分析。

目前預計使用Keras作為主要的深度學習框架，將Zabbix Server監測的各項資料作適當的前處理與特徵選擇，並以滾動的時間窗口產生訓練資料。

**第二年期程**

恩智浦公司的雲端伺服器建置在台灣、中國、泰國等地區，分散在不同的區域。考慮到多個資料中心的大量資料流，單機的硬體條件無法應付即時預測任務，容易拖慢預測時間。另外，雲端伺服器資源使用量會因為使用者需求變動，導致在線預測模型準確度降低。因此第二年期程將以分布式架構部屬第一年的異常預測系統，並簡化更新在線預測模型的訓練流程，減少管理成本。

本計畫為應付多台主機的大量資料流輸入，將使用大規模深度學習技術將預測資料分批傳送至計算節點，再將預測結果回傳至主節點做故障分析，實現分布式預測。另外，系統會隨時監測在線預測模型的準確度，當使用者需求變動導致準確度降低，系統會要求管理者更新模型。然而，重新訓練一個模型的流程是非常複雜的，如何簡化也是另一項難題。我們使用Google Cloud Platform(GCP)、Microsoft Azure等雲端平台的AutoML概念，應用不同策略建立多項預測模型，從中挑選最佳模型部屬，提升在線模型的適應性。

目前預計使用TensorflowOnSpark框架在叢集環境上部屬在線預測模型。系統會讀取Zabbix Server儲存在HDFS的使用率資料，並將資料發配至計算節點進行Pipeline流程預測。AutoML部分預計將原模型準確度作為Baseline，應用兩項策略訓練新模型(1)使用Transfer Learning技術轉移原有模型知識至新模型，再以新資料微調。(2)使用新資料重新訓練一個模型。AutoML功能將以PyQt5整合，使用少量按鈕元件簡化訓練流程，減少管理人員的複雜操作。

1. **市場現況及需求分析**

**第一年期程**

在網際網路發達的時代，透過網路為媒介就可享有多樣的服務。雲端運算透過將運算資源集成，舉凡各家企業、學校、政府等機關都常使用這項技術。虛擬化技術打破了實體結構間的不可分割障礙，使得使用者能更適當的應用運算資源。雲端運算與虛擬化相輔相成，只需要網路和行動裝置方便使用。然而，越大的計算規模，就需要更多的主機連接達到叢集運算的效果，要如何確保所有主機穩定運行是一項具有挑戰性的任務。

現今伺服器維護策略多為即時性策略，僅監測資源使用率超過閥值時發出警示通知。但此策略無法反映真實的異常情境。本計畫開發之伺服器異常預測系統可用於一般網頁伺服器和多主機之集群。

**第二年期程**

現今社群網站及終端感測器發達，短時間產生的巨量資料常導致大量的工作量負載。以Twitter為例，該社群一天必須處理生成的7TB資料，大規模資料量

1. **計畫目的**

**第一年期程**

本計畫伺服器異常預測可以達成以下的目標:

* 1. 使用Tensorflow、Keras訓練資源使用率預測模型
* 擷取Zabbix Server監測資料作為訓練樣本。
* 使用不同深度學習模型準確預測資源使用率。
* 多變量預測增加預測準確度。
* 多步預測未來使用率趨勢，增加管理者的反應時間。  
  1. 結合長時間使用率資料，分析可能發生之故障情境
* 使用Pandas讀取資料表分析。
* 使用tsfresh套件產生多尺度特徵，分析長時間資料。
* 使用Matplotlib視覺化觀察使用量趨勢與相關度分析。  
  1. 伺服器資源使用率的異常檢測與分類
* 使用t-SNE對資料進行降維分析。
* 使用Autoencoder對檢測異常使用率資料。
* 找出故障類別與原因，並通報管理人員修正。  
  1. 預測異常並提前通報
* 使用電子郵件通報超過閥值之異常使用量。
* 使用電子郵件通報異常類別。

表1. 經營要點分析

|  |  |
| --- | --- |
| 營運要點 | 內容 |
| 目標客戶 | * 提供雲端服務與應用之企業。 * 使用網頁伺服器之政府、學校一般機構。 |
| 行銷模式 | * 藉由開辦講座廣邀廠商聽講宣傳經營理念。 * 架設網站提供智慧化解決方案說明書以及播放成功客戶個案運作影片。 * 定期發表期刊論文於學術界提升知名度。 |
| 經營模式 | * 委派專員理解客戶需求，架設資源監測軟體蒐集資料。 * 協助部屬專用系統，並持續提供更新版本升級。 * 累積不同經驗與客戶回饋持續研發更完善的系統。 * 開發更細緻的視覺化互動視窗，提升用戶體驗。 |
| 獲利模式 | * 與客戶簽訂合約模式獲取利潤。 * 透過蒐集多個客戶資料訓練更泛用的預測模型，減少不同客戶之間的轉移成本。 |

**第二年期程**

本計畫分布式伺服器異常預測系統可以達成以下的目標：

1. 使用虛擬機軟體建立Hadoop分散式資料庫與Spark叢集運算環境模擬情境

* 使用VMWare Workstation Player 15建立虛擬環境。
* 在虛擬環境上建立Hadoop分散式資料庫。
* 建立Spark叢集運算環境與Pyspark套件。

1. 使用TensorflowOnSpark分布式架構減少大量資料處理時間

* 將TensorflowOnSpark建立在Hadoop與Spark之混合集群。
* 建立分布式模型預測Pipeline。
* 讀取即時資源使用率串流資料發配至計算節點。
* 使用TensorflowOnSpark佈署在線預測模型，使用Spark-submit建立模型副本傳送至各運算節點，執行分布式預測。
* 運算節點將預測結果回傳主要節點執行故障分析。
* 偵測異常並發出警報。

1. 擬定在線預測模型更新策略

* 對在線預測模型進行持續性的誤差評估，評估是否更新模型。
* 嘗試不同策略訓練模型，例如:遷移式學習。

1. AutoML自動訓練新模型，減少管理人員維護成本

* 建立一個含有目標函數的機率模型，選擇最佳的超參數去評估正確的目標函數。
* 加入不同超參數調整演算法，自動調整超參數以生成最佳模型。
* 選擇一個AutoML框架為核心，修改程式和本系統的專有功能。
* 使用PyQt5整合功能，管理員點擊按鈕元件即可自動更新模型。

表3. 經營要點分析

|  |  |
| --- | --- |
| 營運要點 | 內容 |
| 目標客戶 | * 使用大量集群提供雲端服務之企業。 |
| 行銷模式 | * 藉由開辦講座廣邀廠商聽講宣傳經營理念。 * 架設網站提供智慧化解決方案說明書以及播放成功客戶個案運作影片。 * 定期發表期刊論文於學術界提升知名度。 |
| 經營模式 | * 實際考察客戶情境並為客戶量身打造智慧化的解決方案。 * 打造資料中心為客戶提供後端分析資料支持。 * 投入資金研發更多伺服器管理的解決方案。 * 研發自有品牌伺服器監控工具，並提供預測性服務等智慧化解決方案。 * 研發跨領域的預測性維護技術。 |
| 獲利模式 | * 與客戶簽訂合約模式獲取利潤。 * 透過蒐集多個客戶資料訓練更泛用的預測模型，減少不同客戶之間的轉移成本。 |

1. **重要參考文獻**

**（二） 執行優勢**

**第一年期程**

生產設備的維護策略一直是各企業的關注重點，設備如何能夠穩定提供生產力，且最大的減少維護成本是主要的目標。近年來，深度學習技術的快速發展，預測性維護議題也再次受到重視，並得到更多的改良可能。相較於工業生產機台需要以感測器接收資料，雲端伺服器可以使用程式擷取大量資料，不必安裝感測元件。而現在的雲端伺服器多以條件式維護策略為主，使用伺服器資源監視工具監測資源，並在數值達到預設閥值時啟動警示。此方案雖然能夠立即性的得知系統正處於高負載狀況，卻無法在事前做適當的防護，造成服務的不穩定。因此，本計畫使用循環式神經網路(RNN)提前預測可能的使用率，再以自編碼器偵測異常的使用量，將先進的預測性維護概念應用在雲端伺服器的維護上，提升應用服務的穩定度。

表4.設備維護方法比較

|  |  |  |
| --- | --- | --- |
| 比較項目 | 做法 | 特點 |
| 損壞式維護 | 當設備發生損壞停擺才會維護。 | 最低層級的保養策略，適用於低成本設備，且損設備損壞無立即性危險性與影響。 |
| 預防式維護 | 根據使用經驗得知此零組件或系統的一般壽命，在此壽命週期內進行維護，以避免元件不預期失效造成的系統停擺。 | 此策略為大部分傳統設備的維護方式。但固定周期的維護容易造成冗餘的成本耗費。 |
| 條件式維護 | 在設備上安裝感測器，並預先設定閥值。當感測器偵測到元件達到預設閥值時啟動警示。 | 此方案可以立即得知當下的異常，但缺乏預判性。 |
| 預測性維護 | 使用歷史資料以及演算法模型預測元件故障的機率，以及剩餘的元件壽命，並在元件可能失效前提出警示 | 最高級的維護策略。預測性維護可以最大程度的減少維修成本，提升產能。 |

而近年來也企業也開始研發預測型的維護工具，例如分配最適當的虛擬資源、最佳化的電力分配等功能，也能達到同樣的目的。本計畫針對所有異常情形進行全面性的預測與維護，達到伺服器的穩定運行，保障雲端最基本的要求。本專案SWOT分析如下:

表5. SWOT分析

|  |  |
| --- | --- |
| 強　　勢　(Ｓ) | 威　　脅　(Ｔ) |
| * 開源軟體的使用上極為便利，可依需求修改原始碼。 * 開源軟體無任何軟體授權費用。 * 多步預測提前預測流量。 * 解釋性的預測結果使用者信任系統的決策。 | * 若預測失準將有鉅額損失。 * 客製化營運需要依客戶需求改變系統。 * 依使用需求會產生不同資料集，不同客戶需要建立不同預測模型。 |
| 弱　　勢　(Ｗ) | 機　　會　(Ｏ) |
| * 預測模型會依使用需求變動而失準。 * 大型資料中心主機數量大，如何同時兼顧所有伺服器將會是一項難題。 | * 小規模的伺服器維護需求眾多。 * 目前市場皆為資源使用率監視器，無預測性維護相關產品。 |

面對此SWOT分析，因應對策如下：

* 爭取政府研發經費補助。
* 培養資料分析與時間序列預測之研發人才，深植資通訊技術能量。
* 聘請伺服器建置專長顧問，以不同角度理解客戶需求，加以改善問題。
* 尋求人工智慧中時間序列預測專長之廠商或大學進行產學合作暨技術加值。
* 持續開發泛用性模型減少客製化成本。

**第二年期程**

近年來工業4.0興起一波工業革命，工業生產邁向智慧製造，越來越多生產設備不再僅限於固定動作。其中智慧化機械手臂技術愈趨成熟，許多工廠也將機械手臂投入生產線，讓生產更加智慧化。隨著更多的智慧化設備投入精細的產品生產，會需要更加精確的生產動作，任何細微的誤差皆會造成產品的良率下降，故障與停機更會造成更大的生產維護成本。

本計畫透過感測動作感測器與接收監控參數之巨量資料，運用機器學習進行大數據分析，建立深度學習模型並部屬至計算節點推論出最佳保養策略與即時健康狀態，獲得資訊與知識反饋至企業和工廠的終端加以最佳化調控。減少保養時過多資源浪費與降低意外停機之成本，維持最佳化的效率穩定生產，實踐既定目標以達成使命。本專案SWOT分析如下：

表5. SWOT分析

|  |  |
| --- | --- |
| 強　　勢　(Ｓ) | 威　　脅　(Ｔ) |
| * 開源軟體的使用上極為便利，可依需求修改原始碼。 * 開源軟體無任何軟體授權費用。 * 分布式預測緩解巨量資料的壓力。 * 預測性維護提升雲端服務品質與穩定性。 * AutoML降低訓練模型難度。 | * 若預測失準將有鉅額損失。 * 客製化營運需要依客戶需求改變系統。 * 依使用需求會產生不同資料集，不同客戶需要建立不同預測模型。 |
| 弱　　勢　(Ｗ) | 機　　會　(Ｏ) |
| * 預測模型會依使用需求變動而失準。 * 大型資料中心主機數量大，如何同時兼顧所有伺服器將會是一項難題。 * AutoML工具需針對客戶配置工作站，無法在直接部屬在個人電腦。 | * 大型企業多使用大規模運算伺服器，若使用效果良好可增加曝光度，為產品宣傳。 |

面對此SWOT分析，因應對策如下：

* 爭取政府研發經費補助。
* 累積大型伺服器維護經驗持續開發新功能。
* 培養資料分析與時間序列預測之研發人才，深植資通訊技術能量。
* 聘請伺服器建置專長顧問，以不同角度理解客戶需求，加以改善問題。
* 尋求人工智慧中時間序列預測專長之廠商或大學進行產學合作暨技術加值。
* 持續開發泛用性模型減少客製化成本。
* 針對AutoML軟體提供套裝GPU工作站選購，或是建立雲端服務提供AutoML訓練環境。

**（三） 研究方法與進行步驟**

**第一年期程**

1. **資料前處理**

使用Pandas讀取資源使用率資料，並轉換為Dataframe。針對缺失值我們將以0值補償，視為系統無法監測到資料。資料集為每三分鐘蒐集一筆CPU、Memory使用率資料，並以百分率作為數值單位。本計畫對象為台灣恩智浦半導體股份公司之Oplus應用服務為主，Oplus由四台伺服器(代號1V、2V、3V、4V)提供資源執行。使用Matplotlib資料視覺化套件觀察趨勢，如圖2所示。本計畫將資料使用不同的採樣頻率處理，如圖3、圖4所示。

1. **產生訓練資料**

以Oplus在2021/2/8~2021/3/10的CPU、Memory使用率資料為例，我們將2月資料作為訓練集，並以3月資料做為測試集。使用RobustScaler對資料進行歸一化，再以長度為20的滑動窗口產生訓練資料用於預測未來的20筆資料，如圖5所示。

1. **訓練環境的建置**

本計畫以Windows 10為主，以Anaconda為Python執行環境。安裝Tensorflow-GPU、Cuda、Cudnn、Tensorboard、Jupyter-Lab等必要套件，並使用Tensorflow內建之Keras為深度學習模型設計套件。

1. **預測資源使用率**

本計畫使用RNN模型的改良版Long Short-Term Memory(LSTM)為基本模型架構，修改為雙向時序結構以及更深層的Deep Bidirectional LSTM(Deep BLSTM)作為主要的預測模型，模型架構如圖6所示。此模型考慮了資源使用率過去與未來的時間依賴性，加強了模型的預測能力。我們使用三層Bidirectional LSTM Layers、50 Epoch的超參數訓練，訓練期間的損失如圖7所示。預測成果如圖8所示。另外，我們也針對了重採樣的使用率資料做預測，預測成果如圖9所示。

目前，使用循環式模型已經能夠預測出資料的部分趨勢。觀察擬合圖發現模型對於高峰值的準確度較差，由訓練損失折線圖成果得知目前模型在學習上已經出現瓶頸，經過多次實驗後損失仍然無法低於0.4 MAE。由此可知，循環式神經網路無法適用於本次情境。因此，我們考慮使用Temporal Convolution Network(TCN)作為我們的預測模型。相對於循環式神經網路，TCN對時間序列做因果卷積(Causal Convolutions)，能夠保留較多的時間訊息。

1. **伺服器異常檢測與分類**

以Oplus在2021/5/4~2021/6/3的資料為例，該資料因為出現異常導致無法接收資料，其包含缺失值，如圖10所示。對於多個伺服器資料的異常檢測，我們使用無監督式的深度學習模型Autoencoder來檢測異常。Autoencoder使用編碼與解碼的架構，使得它能夠還原出原始資料。在異常檢測裡通常會包含大量的正常樣本以及極少的異常樣本，因此我們使用一維卷積層建立Encoder和Decoder，並以正常資料訓練，將模型用於檢測異常資料。

在異常資料裡，第127筆資料的CPU與Memory皆有出現高峰，如圖11所示。我們擷取出2V\_Memory\_usage欄位前300筆資料作為樣本，使用Autoencoder來還原資料。原本為0值的資料經過模型還原後，約在第127筆資料準確還原了高峰值，如圖12所示。我們將以此為基礎，對預測的使用率執行異常檢測。若真實資料與預測資料誤差超過閥值，系統將以此作為異常判定標準之一。另外，我們希望能夠對異常分類，預計使用t-SNE降維分析資料的分布，並將分類結果也包含在警示訊息內。

1. **發送警報**

當系統執行在線預測時，第一步由接收端擷取即時串流做前處理，對使用率資料做歸一化和拆分處理。第二部會將資料輸入模型預測資源使用率。第三步將歷史資料與預測資料連接後進行第二階段的異常偵測。第三步將會計算真實值與預測值之間的Mean absolute percentage error(MAPE)，判斷是否通知管理人員異常。若預測結果為異常則會以電子郵件通知管理人員處理，系統流程圖如圖13所示。

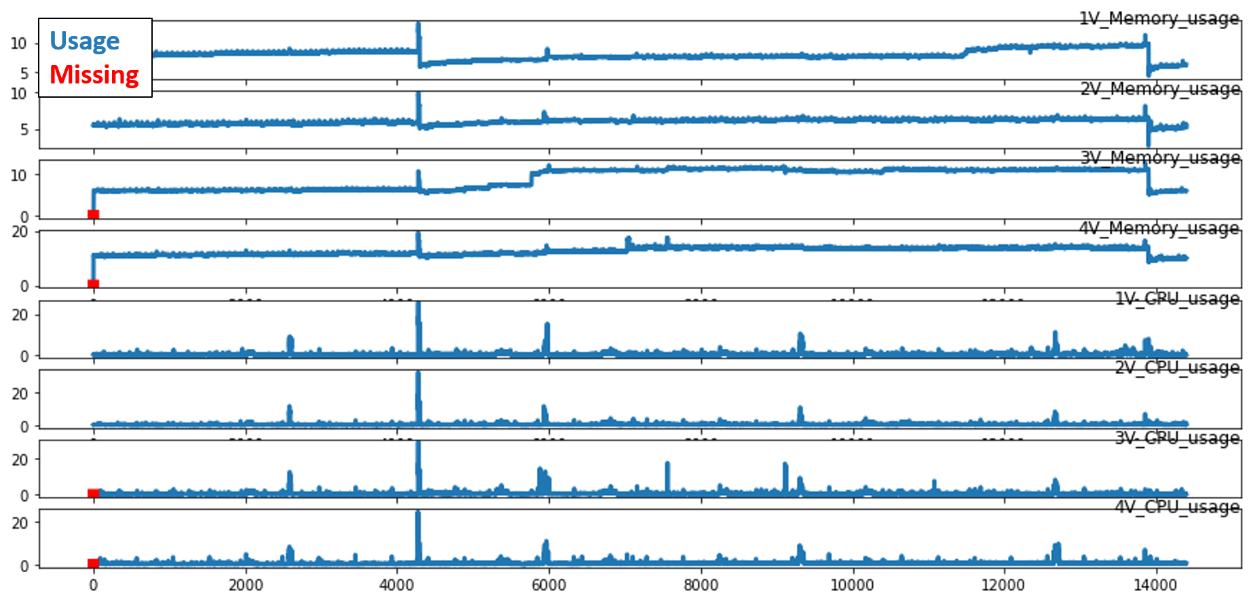
****

圖2. Oplus四台伺服器之資源使用率

|  |  |
| --- | --- |
| C:\Users\user\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\C61177AB.tmp  圖3. Oplus原始資料趨勢 | C:\Users\user\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\4C4DE227.tmp  圖4. Oplus資料以每小時重新採樣趨勢 |

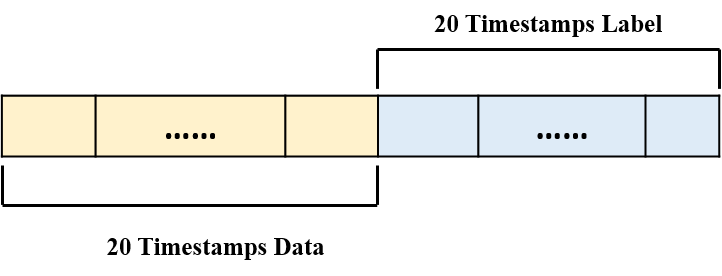


圖5. 訓練資料和標籤

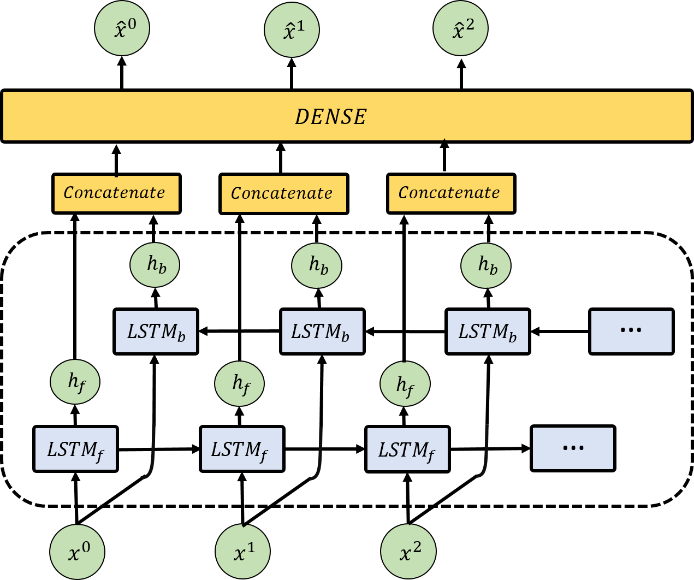


圖6. Deep Bidirectional LSTM

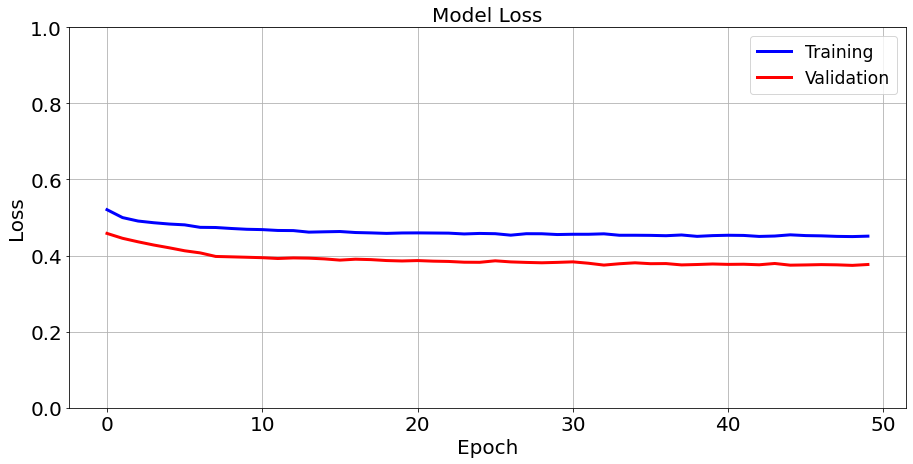


圖7. 訓練過程損失折線圖

|  |  |
| --- | --- |
| 圖8. 預測值與真實值之擬合圖 | C:\Users\user\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\5813C7C1.tmp  圖9. 重採樣資料之擬合圖 |

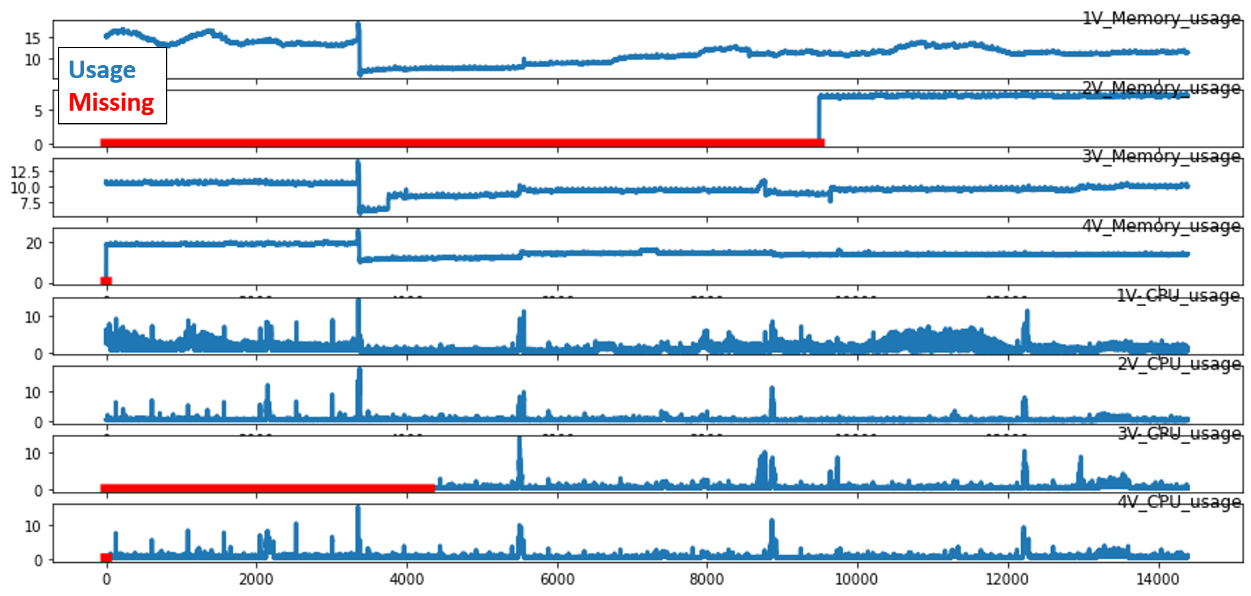


圖10. Oplus異常資料

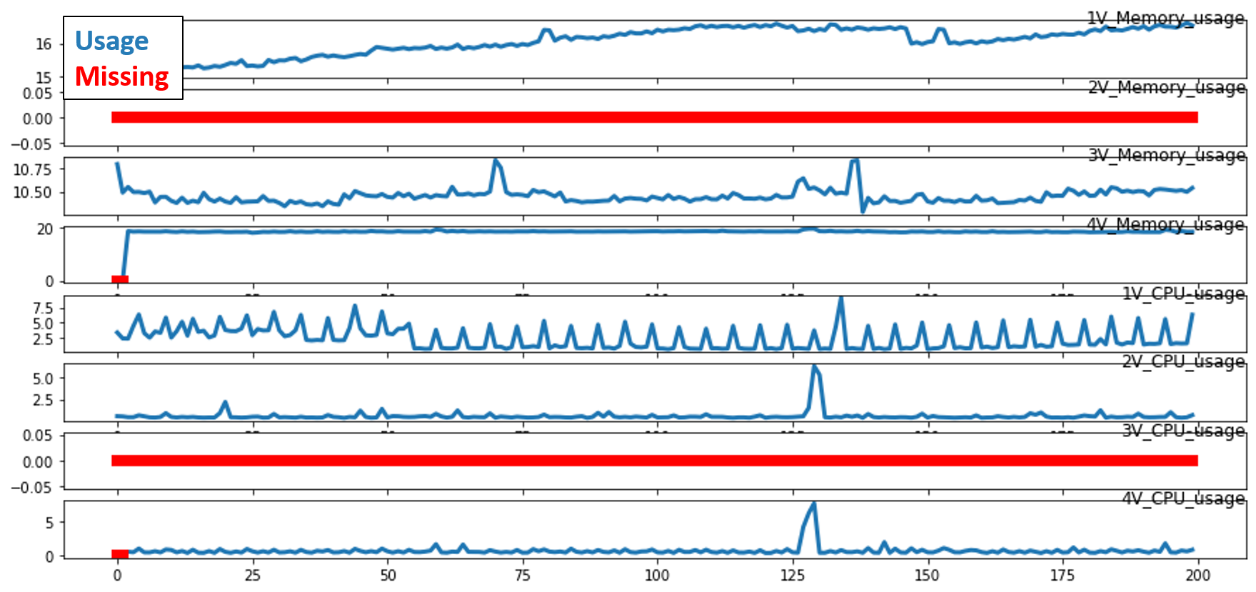


圖11. Oplus異常資料第0~200筆資料

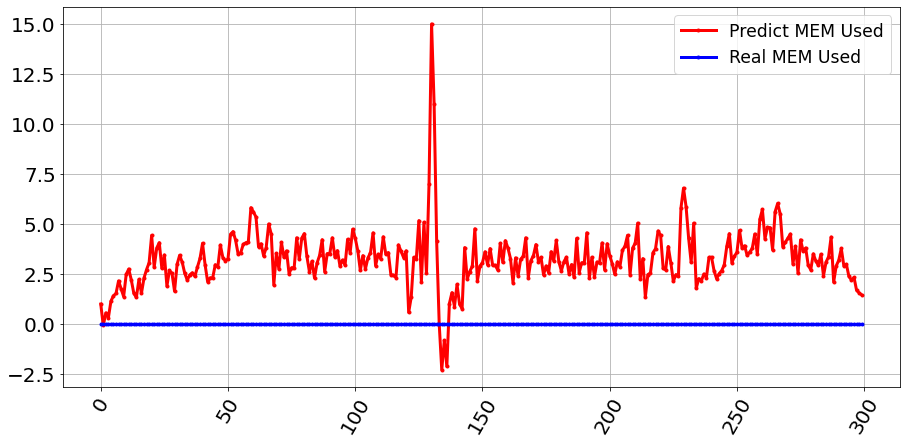


圖12. 還原異常資料

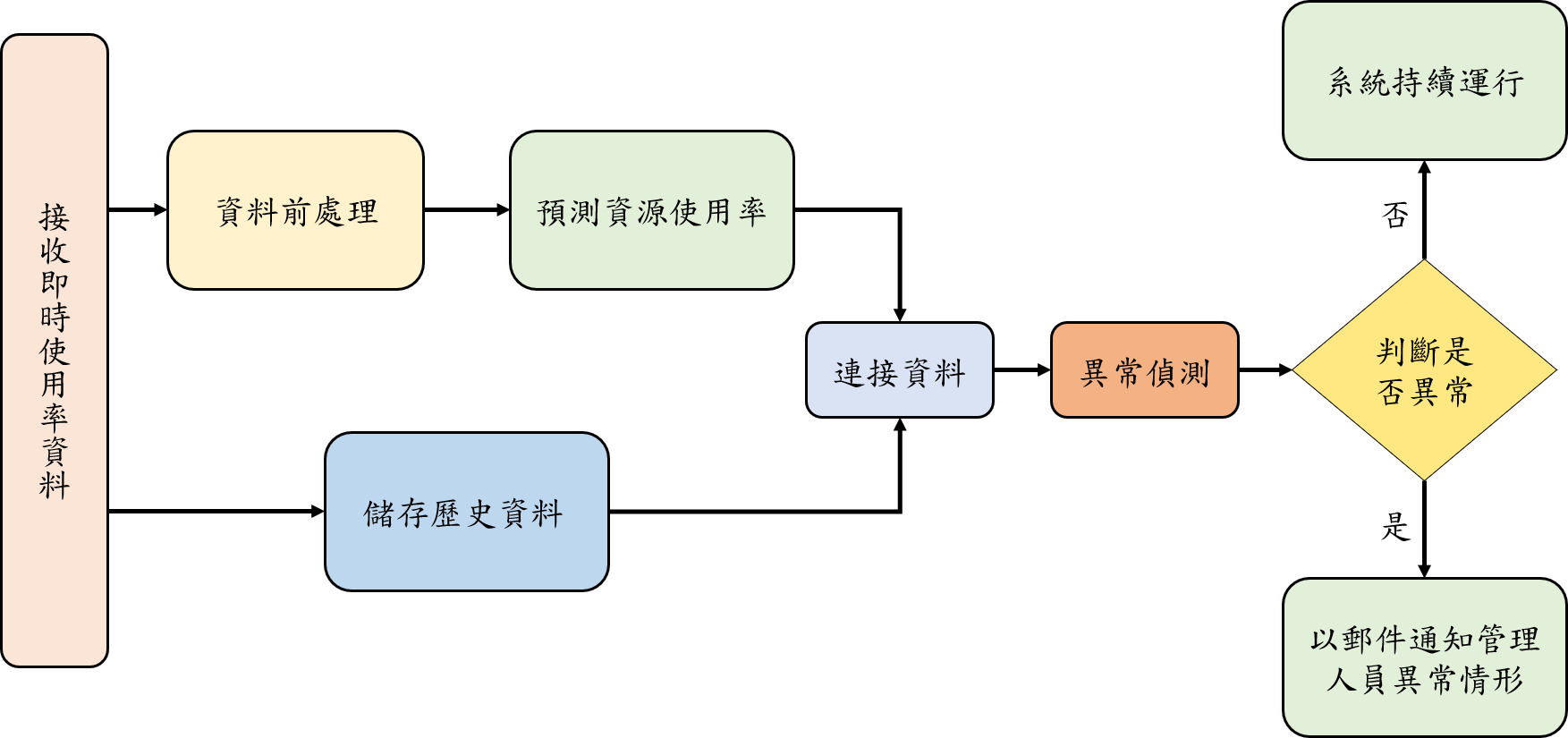


圖13. 異常預測系統流程圖

**第二年期程**

1. **建立模擬叢集環境**

本計畫結合Hadoop和Spark的建立叢集運算系統。Hadoop是一個免費開源的分散式儲存框架，該框架使用平行擴充的方式為大量資料儲存提供一個解決方案。Hadoop Distributed File System (HDFS)為Hadoop的分散式檔案系統，系統會將檔案拆分成數等分的Block分配給DataNode，並將存放位置清單將放置在NameNode。Spark是一個開源的叢集運算框架，提供了彈性分散式資料集Resilient Distributed Datasets（RDDs）、Spark SQL、Spark Streaming、MLlib等功能，也兼容Hadoop及Yarn。

本計畫將建立一個Hadoop和Spark模擬環境。我們使用VMWare Workstation Player 15新建兩台虛擬機，安裝Ubuntu18.04作業系統。將虛擬機資源配置為4核心處理器與8GB 記憶體，如圖14所示。叢集之Master節點及Slave1節點作為Worker，每個節點在啟動時分配4GB 記憶體及4個CPU核心，並透過瀏覽器搜尋Master IP觀察叢集啟動狀態，如圖15所示。由於恩智浦公司的Zabbix Server監控資料將會不斷匯入HDFS，因此我們將資源使用率資料集儲存至HDFS，並使用PySpark讀取資料。目前，模擬環境已經架設完成，做為初步的實驗環境使用。

1. **建立大規模深度學習環境**

本計畫使用TensorflowOnSpark (TFoS)大規模深度學習框架。TFoS可將原本的Tensorflow程式能應用在已架設之Hadoop和Spark叢集上，支援model parallelism、data parallelism、inference功能。InputMode.TENSORFLOW可以使用Tensorflow提供的API直接提取HDFS資料。InputMode.SPARK保留Spark RDD資料型式，使用Pipeline優化模型的預測流程。我們使用TensorflowOnSpark InputMode.SPARK提取HDFS的Zabbix Server監測資料，再由Master分配資料到各個節點平行化預測，如圖20之(1)、(2)部分所示。

1. **分布式異常預測**

首先Master會先讀取HDFS之Zabbix Server監測資料，經由PySpark轉換為Dataframe形式，如圖16所示。轉換後資料會分批傳至計算節點，如圖20之(3)部分所示。我們使用TensorflowOnSpark.pipeline TFModel Transformer設計Pipeline的預測流程，指定使用率欄位轉換為模型的輸入層，再將模型的輸出層轉換為預測結果的Dataframe輸出，如圖17所示。程式編輯好後再以spark-submit執行，指令上需指定Master IP、CPU核心數量、Worker的CPU核心數量、叢集數量，如圖18所示。

我們以2個主機共8個CPU核心做為Worker執行分布式預測，預測後會生成一個JSON檔案儲存至HDFS，檔案裡記錄著預測的使用率數值與對應的時間戳，如圖19所示。經計算後每個Worker的平均預測時間大約為0.1秒，我們相信此方法可以減少大量資料量下的預測時間。目前，系統將資料傳送至各個節點分布式預測，再將結果傳回主節點匯總，如圖20之(4)部分所示。我們預計會加入第一年期程之兩階段預測流程，當主節點匯總了預測資料後，執行Autoencoder異常檢測，當預測值與誤差值之MAPE超過閥值，發送警告郵件通知管理人員，如圖20之(5)部分所示。

1. **AutoML應用程式開發**

在線預測模型會因為使用者的需求改變，導致使用率趨勢變動

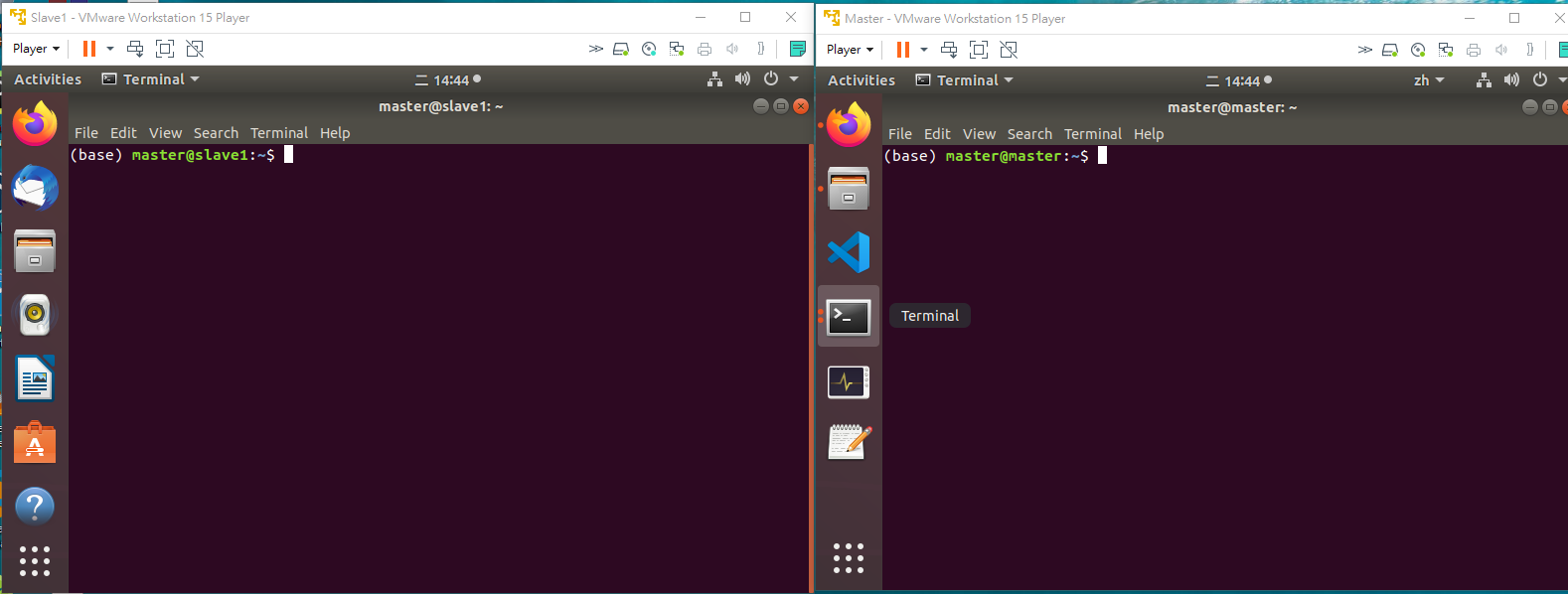


圖14. VMWare Workstation Player 15虛擬機

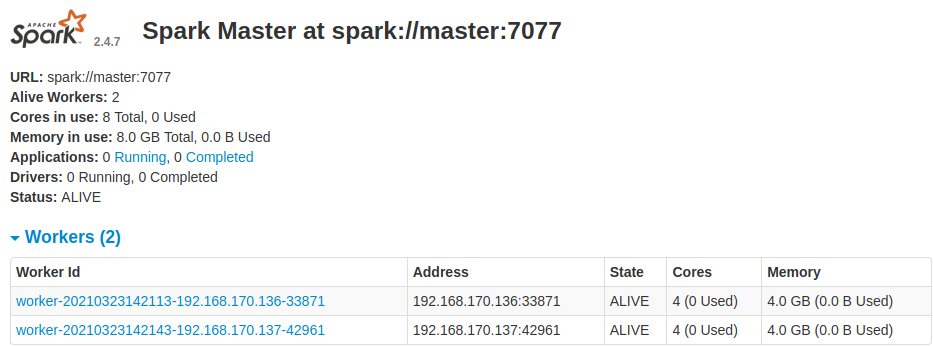


圖15. 使用瀏覽器觀察叢集啟動狀態

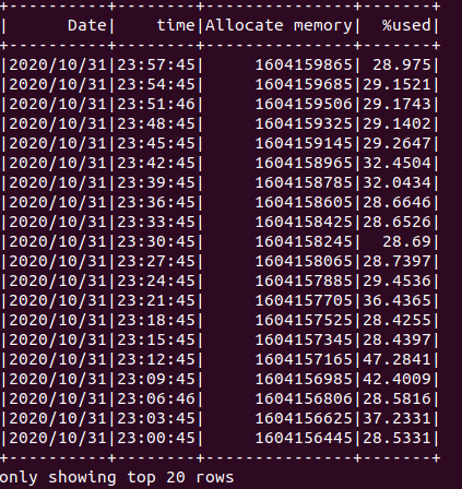


圖16. 使用率資料轉換為Dataframe型式

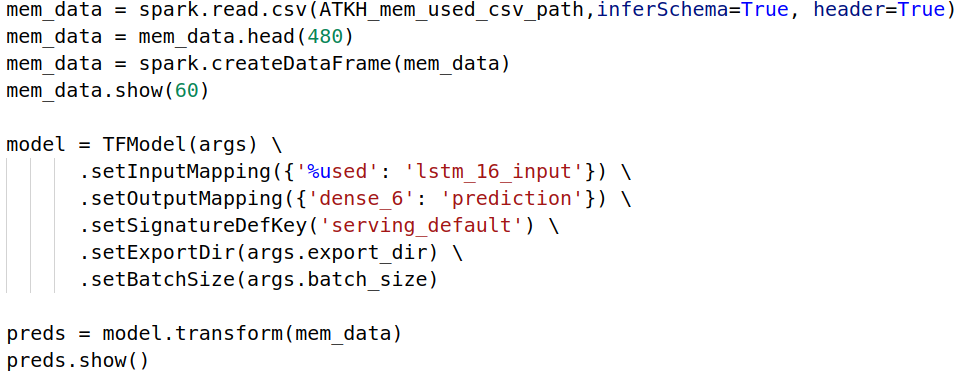


圖17. TFModel Transformer

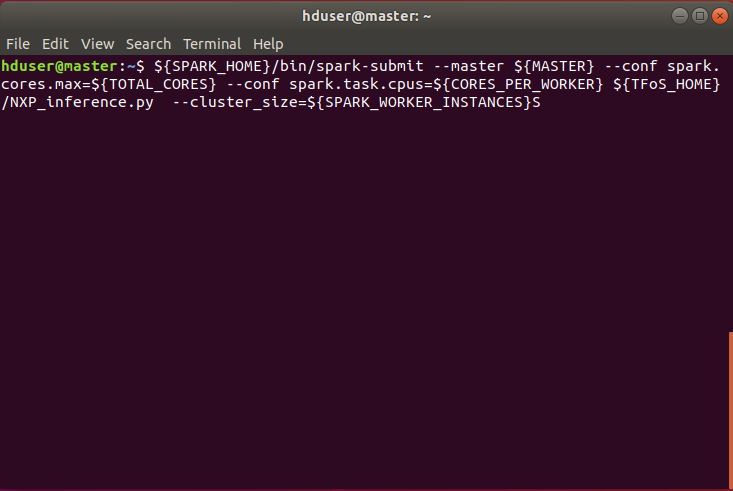


圖18. 執行指令圖

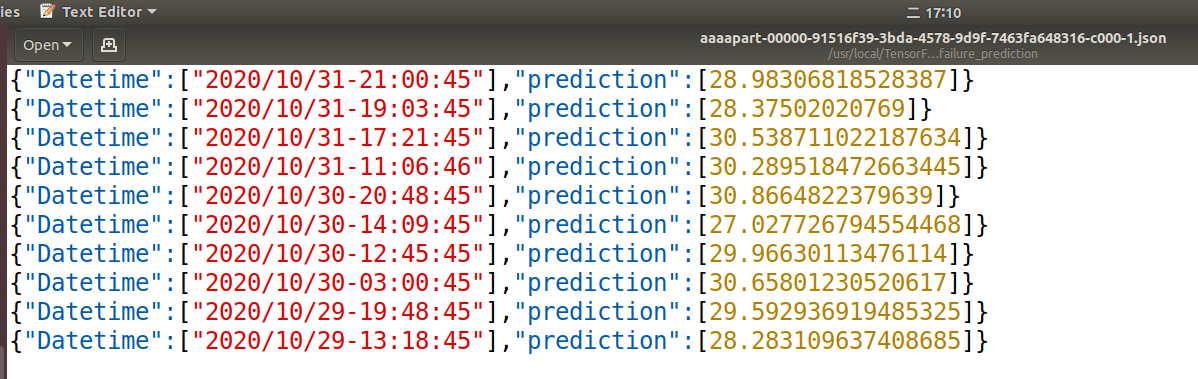


圖19. 預測數值JSON檔內容

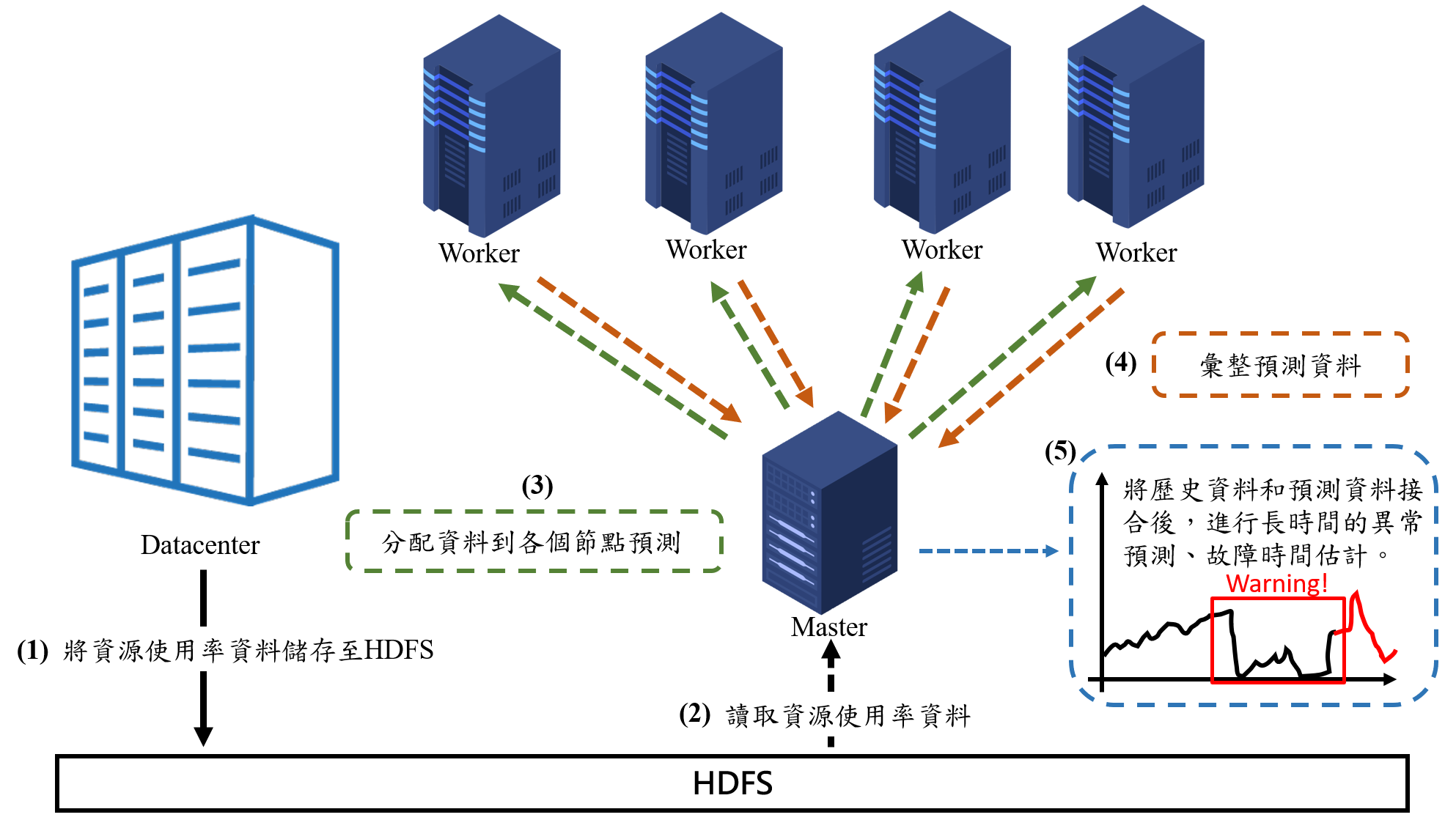
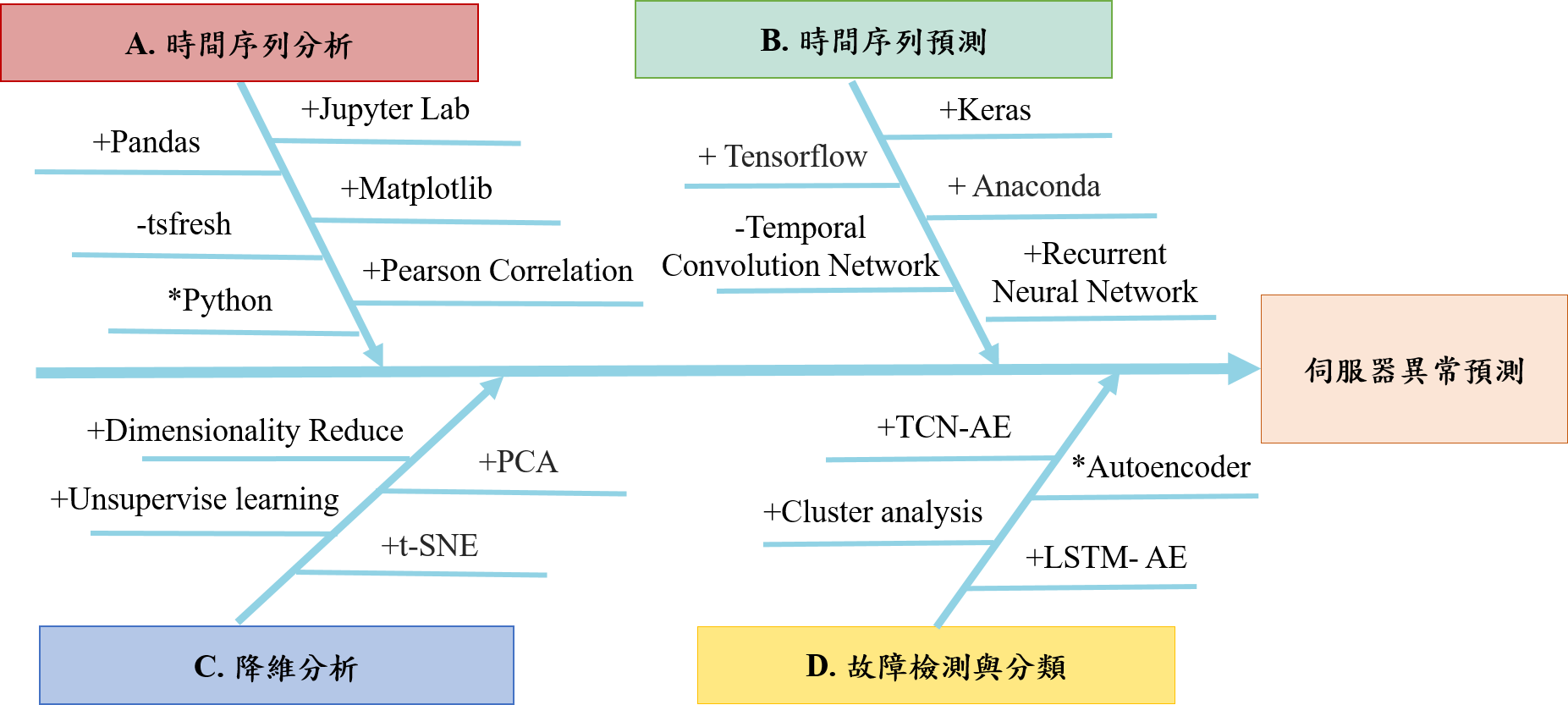


圖20. 分布式異常預測流程圖

1. **產學計畫之需求性**

**第一年期程**

如圖27所示，本計畫晶片瑕疵檢測分析的技術關聯涵蓋了時間序列分析、時間序列預測、降維分析、故障檢測與分類共四個部份。



註：請加註標記符號說明。

『＊』表示我國已有之服務、技術或產品（並註明企業或機構名稱）

『＋』表示我國正在發展之服務、技術或產品（並註明企業或機構名稱）

『－』表示我國尚未發展之服務、技術或產品

圖27. 伺服器異常預測技術關聯圖

**本產學計畫之需求性說明如下：**

**A. 時間序列分析**

計畫中會先針對Zabbix Server監測之資源使用率進行時間序列分析。以Python為主要程式語言，Python提供了許多對資料分析的套件，

**B.** **時間序列預測**

計畫中所使用的深度學習框架為Tensorflow，建立在Anaconda開發環境上。

**C. 降維分析**

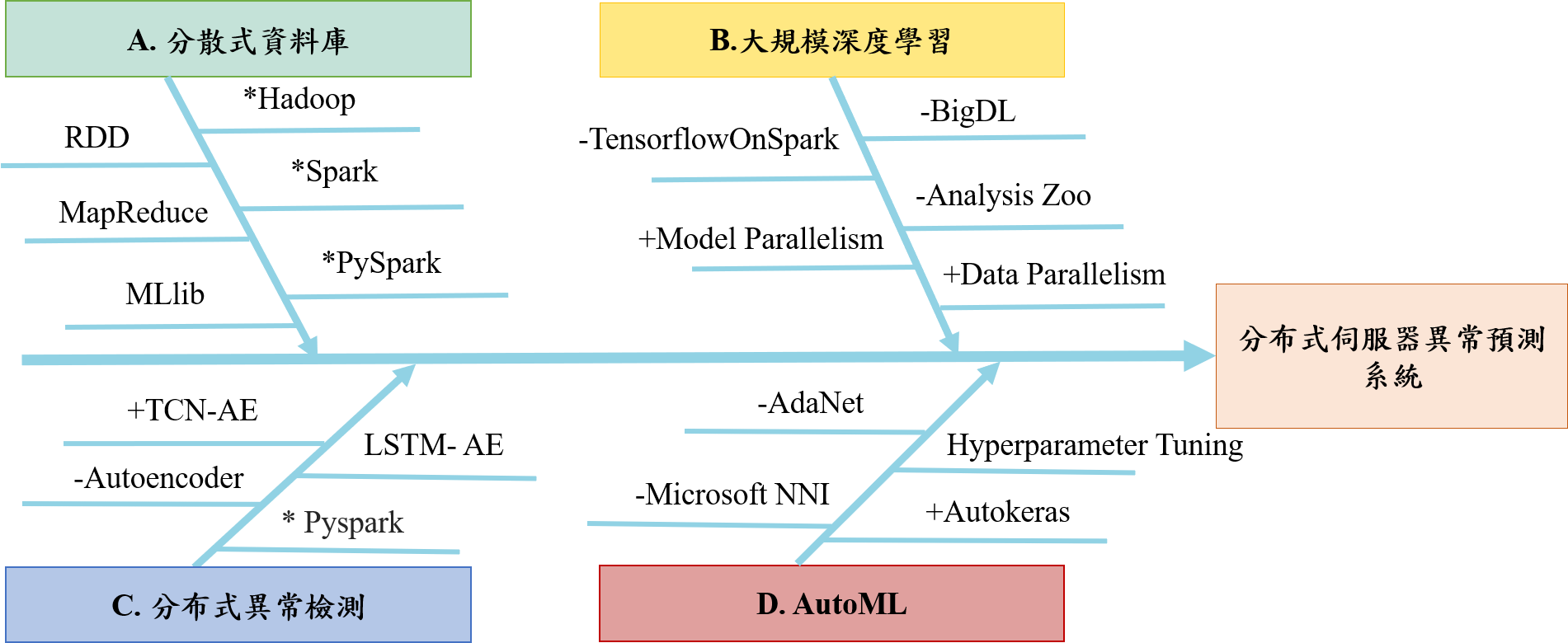
計畫中預計會使用PLC進行機台的智動化控制，因為以人為停機仍然不是一個好的解決方案，所以當接收到來自CCTV回傳的訊號後，異常偵測系統即會開始進行檢測，而只要異常偵測系統檢測到有缺陷的晶片時，會根據目前晶片的狀態進行資料分析，只要回傳的內容具有defect類別時，則保存內容並製作一個警示系統，該系統能夠即時告知產線人員目前產線上的晶片有缺陷的情況，並且將訊息回傳給機台，透過PLC智動化控制機台使機台能夠即時停止運作，達到智慧化與智動化的晶片監控系統。

**D. 故障檢測與分類**

計畫中以NXP的i.MX8M開發板為中心，主要使用eIQ開源軟體套件，eIQ中包含計畫中需要使用的推理引擎，並且在基於Linux的Yocto開發環境進行使用，而eIQ所支援的神經網路框架為TensorFlow Lite，嵌入式平台大多數會選擇使用TensorFlow Lite是因為嵌入式平台的效能有限性，如果利用如一般主流框架的模型如TensorFlow、Keras、Pytorch等進行物件偵測時，容易因硬體效能導致檢測品質低落，如FPS(Frame Per Second)幀數過低造成延遲的問題，於是將目前TensorFlow的Keras模型轉換成輕量級的TensorFlow Lite模型，並評估檢測品質是否良好，如果FPS仍無法達到理想值，則會考慮轉換成ONNX模型再進行評估，比較兩者優化後的模型好壞。而i.MX8M平台所使用的處理器ARM Cortex-A53有支援OpenCV函式庫，所以在使用上會更為方便，尤其是攝影機的即時影像捕捉會需要OpenCV的功能，對計畫的進行將有很大的助益。

**第二年期程**

如圖28所示，本計畫的技術關聯含蓋了分散式資料庫、大規模深度學習、健康診斷模型以及圖形化使用者介面共四個部份。



註：請加註標記符號說明。

『＊』表示我國已有之服務、技術或產品（並註明企業或機構名稱）

『＋』表示我國正在發展之服務、技術或產品（並註明企業或機構名稱）

『－』表示我國尚未發展之服務、技術或產品

圖28. 分布式伺服器異常預測系統技術關聯圖

**本產學計畫之需求性說明如下：**

1. **結合Hadoop與Spark的大數據環境**

計畫中以NXP的FRDM-K22F-AGM01開發板為中心，分別由FRDM-K22F與FRDM-STBC-AGM01結合，透過FRDM-STBC-AGM01板上嵌入的FXAS21002(陀螺儀)和FXOS8700CQ(加速度計與磁力計)感測器融合算法精確感測機械手臂動作姿態。該感測器開發是以MCUXpresso開發，MCUXpresso為NXP公司為其產品開發的基於Eclipse之MCU開發環境，提供編譯、除錯、腳位設定以及應用開發套件SDK，主要以C語言編寫。利用嵌入式的動作感測，感測更為高階級的機械手臂的TCP方向精度、手臂的軌跡、力和扭矩、關節位置，使用UART(Universal Asynchronous Receiver/Transmitter)將資料由感測平台傳出至後端接收並加以分析。

1. **大規模深度學習與分布式異常預測**

計畫中使用機器學習的降維、聚類等資料探勘技術分析機械手臂的監控資料，在降維部分，由於監控資料是來自各個感測器的資訊，每個時間點會有多個欄位的資料輸出，若未經處理在分析時除了造成龐大的運算量，而且在高維的空間裡特徵分布是非常稀疏的，不容易從資料點中得知資訊。在Scikit-learn裡包含了需多實現機器學習的工具，其中PCA能將資料點從高維的空間降維到二維的空間觀察，透過映射到低維度的空間，且保留原資料的重要特徵，在資料的視覺化上較容易判斷感興趣的資料分布。在聚類部分，本計畫是從一筆龐大資料中找尋可能故障特徵與衰退趨勢，沒有透過預先的正確類別標記得知故障訊號的訊號，需要透過無監督式的SOM聚類分析，分離出故障訊號後再加以分析建模。開發過程中需要用到Pandas將訊號整理成DataFrame資料格式方便對欄位執行複雜運算，並使用Matplotlib將資料分析的成果視覺化。

1. **遷移式學習**

本計畫接收了機械手臂多方位資料，以機器學習的方式根據資料的特徵分析出精確的保養時間和最佳運作速度，尋找出保養和運作速度與產能之間的最佳平衡。將輸入的樣本資料透過高階的Keras函式編寫一個Sparse autoencoder深度學習模型無監督式的分類出故障特徵，並定義後端的Tensorflow運算流將資料運算交由GPU的平行運算，加速模型的訓練過程。將模型轉換成ONNX model部屬在終端的嵌入式開發平台，為設備即時監控與診斷健康狀態。

1. **AutoML**

本計畫透過感測、分析、診斷機械手臂達到故障預測與健康管理，功能包含故障預測、衰退檢測、健康診斷，將多樣的功能透過Qt 視窗應用程式框架提供的Python介面套件PyQt5整合多樣的功能建立一個Human-Machine Interface提供有用資訊給使用者參考。其中繪製數據資料使用PyQtGraph套件。

**（四） 預期完成之工作項目及成果**

1. **預期達成之工作項目與具體成果**

**第一年期程**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 項目 | 成果目標 | 功能說明 | 作法 | 驗證方式 |
| 深度學習與影像辨識偵測瑕疵殘留物 | 以PC為主要環境進行影像辨識，達到高準確度的辨識，並且能夠即時地將晶片異常瑕疵的狀況排除。 | 以物件偵測的方式對需要辨識的物件進行框選。 | 首先取得本計畫中所提供的資料樣本，之後將資料樣本針對需要進行檢測的物件進行標註，每種類別的標註數量盡量需維持平衡，最後利用分類完成的資料樣本進行YOLOv3模型的訓練，最後將CCTV的影像訊號作為測試模型的好壞。 | 精準測試:利用評估模型準確度的方式，如果準確度不如預期則需要再進行重新訓練，若準確度達到預期水平則以驗證集的資料樣本進行測試，每一種類別檢測正確的機率皆需達到七、八成才合乎預期水平。 |
| 嵌入式平台部署 | 主要會使用i.MX8M嵌入式平台來進行檢測的功能，透過嵌入式平台可同時讀入多個CCTV訊號源同時進行檢測晶片是否有缺陷。 | 利用嵌入式平台做到檢測的功能而不需使用PC進行，降低硬體設備成本。 | 一開始會先將i.MX8M平台原生Android系統重灌成Linux系統，隨後進行所需套件的下載，也將訓練好的模型進行轉換後得到較輕量的模型部署到平台內，平台部署完成後則把CCTV的訊號源接上平台，對產線內實際的晶片運送情形進行檢測。 | 單元測試:進行檢測時FPS(Frame Per Second)會希望越高越好，因為當FPS數值較低，CCTV即時串流的訊號畫面將會不流暢，造成延遲的問題，雖然檢測準確性仍具備但即時性卻嚴重受到影響。而經過由CCTV訊號輸出影片的測試得到影片的FPS約為7，表示當檢測時的FPS小於7，則不具即時檢測的效果，對於本計畫晶片檢測的即時性訴求即無法達標。所以將FPS設定門檻，訓練好的模型加上嵌入式平台所進行的檢測，其FPS需大於7才能夠達到使用的條件。 |
| 智動化檢測生產機台 | 當檢測的類別為defect時，則將訊息傳回機台使機台停止運作。 | 以PLC的方式去控制機台的動作，主要是需要實作出使機台停下的動作。 | 以檢測出的類別物件做為依據，只要檢測出類別有defect的出現即表示晶片槽目前有異常狀況，將機台立即停止運作，如果檢測出的類別是ocuupy、empty則不屬於異常狀況，不需執行智動化控制。 | 壓力測試:如果是以測試影片做為測試可能不夠確實，必須直接在產線上實作進行測試，在各種不同的狀況下才能夠反映最真實的情況。 |

**第二年期程**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 項目 | 成果目標 | 功能說明 | 作法 | 驗證方式 |
| 嵌入式動作感測 | 將動作感測器FRDM-K22F-AGM01安裝至機械手臂的各部位感測手臂TCP的資料。 | 使用FRDM-K22F-AGM01開發板，藉由板上的動作感測元件感測機械手臂之加速度、位置、姿態共九軸的動作資料。 | 將感測板安裝在機械手臂的前端、移動平台、樣本上感測，使用MCUXpresso SDK開發感測元件的動作感測功能，使用輪巡模式(Pooling mode)輪流尋訪FXOS8700CQ感測的三軸加速度數值、三軸磁力數值和FXAS21002C感測的三軸陀螺儀數值資料，透過開發板上的I2C bus整合至Debug Console上顯示，確定有資料輸出。 | 精準測試:感測板安裝機械手臂後，分別運作同一動作10次，觀察資料分布是否一致，並計算誤差值是否在合理範圍。  整合測試: 先將機械手臂各部位分別完成測試，確定都能正常運作後，再進一步整體系統隨機式情境模擬測試10次，檢驗是否全部通過，若不是則再修正再進行下回合10次隨機式情境模擬測試，直至全部通過為止。同時測試感測板在運作長時間下是否維持一定的準確度，評估感測器運作時間與準確度之關係。 |
| 資料傳輸與儲存 | 感測資料透過有線傳輸技術將感測資料傳輸至儲存設備，提供後續分析資料樣本。 | 擷取機械手臂上的感測資料，資料傳輸與儲存將分為以下兩個部分處理:   1. 感測板傳至運算節點:由於感測板本身無儲存空間，需將感測資料從感測板傳送至運算節點。 2. 運算節點至儲存空間:若以有線方式則需要將資料儲存在運算節點的儲存空間內，用於分析資料階段使用。   在測試階段需要儲存資料並提取出，用於分析資料；在監測階段需要不間斷收資料以預估出保養日期。 | 有線傳輸:將加速度計、磁力計、陀螺儀感測資料以逗點間隔九項資料並整合，透過UART傳輸至額外建置的運算節點，在測試期間須將資料以CSV檔案儲存，並提供給後端分析資料。 | 單元測試:確保傳送資料正常，我們可透過觀察CSV時間欄位對比目前時間的數據資料增加來檢查資料是否都準確儲存至磁碟空間。 |
| 動作資料分析 | 從接收資料中預估出準確的保養日期、壽命，同時也能找出能運作最久的運行速度。 | 將動作資料透過降維、分群，找出離群點的異常資料特徵。將異常程度數值化用於後續評估出準確的保養時程以及從資料預估出運作最短停機時間與最高產能之最佳速度。 | 透過SOM、PCA等資料分析技術找出離群點的異常資料，將異常程度數值話並評估出準確的保養時程。 | 精準測試:輸入長期運行之資料輸出數值化之設備衰退程度，比對真實情況之停機時間，評估預測準確度。 |
| 建立健康診斷模型 | 接收多樣的數據資料並建立一個深度學習模型，用於運作時監控設備故障的可能發生位置，提供直覺性的故障檢查。 | 即時接收機械手臂資料輸入健康診斷模型推論出手臂的故障關節。 | 使用Sparse Autoencoder無監督式深度學習模型，輸入監控資料並在壓縮後的模型表徵空間加上Regularization使其保持稀疏性，使其成為一個專門分類故障情形的專家。 | 精準測試:將系統處於長期監測狀態，並在停機時檢測故障關節之損耗度評估模型預測是否準確。  運算平台測試:測試不同的運算平台的運算速度、穩定程度、硬體成本、軟體相容性等條件，評估健康診斷模型的最佳運算平台。 |
| 圖形化介面 | 製作一個視覺化界面提供給產線人員參考機械手臂相關保養資訊，重點在於完整的資訊呈現與模型推論過程直覺化。 | 顯示設備上的監控參數值，並顯示明確的保養日期、壽命檢測數值、故障部位預測。 | 使用Qt GUI框架支援Python的PyQt5製作，建立一個視窗應用程式，使用PyQtGraph即時繪製監控參數走勢，資料分析的結果，內部包含了顯示保養日期、監控參數、故障部位預測顯示圖。 | 壓力測試:實際連接上數據傳輸線接收數據，觀察健康狀態顯示與設備保養日期與真實狀況之誤差，評估誤差的大小判斷模型的好壞。  使用測試:交由產線人員實際使用，提出顯示參數與互動介面的改動意見。 |

1. **對於學術研究、產業界、國家發展及其他應用方面預期之貢獻**

**第一年期程**

影像辨識結合深度學習的做法最重要的概念就是資料樣本，只要能夠取得資料即可針對資料提供的資訊給予相對應的回饋，完成需要進行的檢測動作，經過訓練最終得到一個辨識度最準確的模型，應用於本計畫的晶片瑕疵檢測分析。相信該計畫能夠為產業界帶來貢獻，如果在各工廠中有需要針對瑕疵品進行檢測時，則可以利用核心價值進行處理，對公司訴求能夠有效的解決。

**第二年期程**

因應工業4.0時代來臨，各工廠致力於研發智慧製造，產業界在設備保養上通常以被動保養和預防性保養為主，但如此的保養策略帶來的產能降低和過度保養都會提高生產成本，透過本計畫的動作感測與資料分析技術為工廠生產線提供更智慧化的設備保養與健康管理，解決傳統方法的缺陷，有助於提高產能和降低生產成本，讓產線的保養策略更智慧化，提升工廠產能。

「設備的故障預測與健康管理」保養策略適用於各種設備，依照設備的零件、功能、參數等的不同，制定專有的模型管理設備健康狀態，若能以最少量的模型足以管理整個廠區的生產設備，可以大幅度的降低保養成本和安裝成本，帶動生產效能增加。

1. **本產學計畫之技術或知識服務應用等範圍**

**第一年期程**

本計畫的技術著重在以深度學習與影像辨識進行晶片瑕疵的檢測分析，並以檢測的結果進行智動化控制，可為產線減少產品的損失數量，經由本計畫的建置，研發人員於深度學習、影像辨識、嵌入式平台、自動檢測及系統整合業相關理論的理解必能有所提升。

**第二年期程**

本計畫著重於動作感測、資料分析、設備保養與健康管理方面的發展。經由這次計畫的建置與設計，研發人員於系統規劃、程式撰寫方面，專業技巧與能力必能有所提升，尤其在資料分析部分，透過此次計畫的訓練有助於將資料分析知識轉移到不同設備資料，同時解決更多工廠面臨的問題。對於開放式軟體的系統使用，需要與具備基礎專業的學術人士來共同加速計畫流程，對於初學開放式軟體的計畫成員必定有所受益，也能提升計畫成員的整體能力。

1. **對於參與之工作人員，預期可獲之訓練**

**第一年期程**

(1) 熟知開源軟體開發與使用技術。(2) 獲得軟體工程相關概念與技術。(3) 智動化控制的相關概念與使用技術。(4) 學習影像辨識的相關概念與技術。(5) 理解閉路電視的概念與相關技術。(6) 學習通訊協定的相關概念。(7) 學習深度學習的概念與技術。

**第二年期程**

(1) 熟知開源軟體開發與使用技術。(2) 熟悉嵌入式系統資料傳輸介面。(3) 熟悉嵌入式系統開發流程。(4) 學習數據探勘流程。(5) 學習機器學習聚類分析的概念與技術。(6) 學習深度學習的概念與Autoencoder技術。(7) 學習生產設備保養策略分析之技術。(8) 熟悉視窗化介面開發與人機互動介面設計。