

國立台北大學資訊工程學系專題報告

懶人の救星:個人化套裝推薦系統

專題組員:章劉軒瑋, 顏佑勳, 唐誌遠, 楊璉侑

專題編號: PRJ-NTPUCSIE-111-010

執行期間:111年9月 至112年6月

1. 摘要

為了給予更多服裝搭配的靈感，本專題實現了個人化套裝推薦系統，旨在給予使用者喜歡的服裝搭配，在穿搭時做出更明智的選擇。我們提出了一種 rule incorporated semi-supervised learning 的推薦方法，採用了半監督式學習和深度學習的技術，建立模型來學習套裝的特徵和使用者評分之間的關聯性，將預測評分較高的服裝搭配推薦給使用者。最後的實驗結果也顯示出，此系統推薦的服裝搭配會越來越貼近使用者真正的喜好，推薦出使用者沒想過但喜歡的服裝搭配。

2. 簡介

在選購衣褲或穿搭時，儘管服飾店擁有眾多款式的衣服和褲子供我們選擇，但在服裝搭配上卻沒有足夠的範本，而花費較多的時間。在服裝搭配時，自己喜歡的衣服搭到喜歡的褲子上不一定會好看，但某些衣服單看的時候沒有特別喜歡，和某些褲子搭在一起卻又會有不錯的效果，因此我們想要的是推薦整套的服裝搭配，讓

我們了解怎樣的搭配我們會喜歡，而不僅是推薦衣服或褲子供我們做選擇。

表一、與市面上推薦系統比較

	推薦單件	推薦搭配	推薦量大	個人化
lativ [1]	✓		✓	
XZ(Closet) [2]		✓		
WEAR時尚搭配 [3]		✓	✓	
個人化套裝推薦系統		✓	✓	✓

比較其他市面上服裝推薦系統（表一），目前大型服飾店的網路商城（如 lativ）以單件的衣服或褲子做推薦，而只看衣褲會很難想像他實際搭出來的效果。XZ（closet）和 WEAR 時尚搭配是以推薦搭配為主，但前者是以天氣來推薦搭配且一次只有推薦六套服裝搭配，顯然推薦的數量非常不夠。後者是像社群軟體一樣，讓其他的用戶推薦出他們喜歡的穿搭供其他用戶參考，問題是推薦出來的衣服是別人喜歡的不一定會貼近我們的喜好，用了老半天也許也找不到和自己頻率對到的服裝搭配。我們想做的服裝搭配推薦系統與其它者不同能達到服裝搭配數量多、服裝搭配推薦、個人化這幾個好處。

我們希望一個能夠根據使用者的喜好學習推薦大量服裝搭配的系統，讓像我們這樣的資工宅男能夠接觸到更多的服裝搭配，以獲得自己的穿衣靈感！而根據每次的評分，使用者可以更清楚自己喜歡的搭配類型，進而在穿搭和購買衣服方面提高效率。同時，希望這個系統能最終推薦出我從未想過但可能會喜歡的服裝搭配給使用者。

因上述種種動機，本專題建置了 rule incorporated semi-supervised learning 服裝搭配推薦系統，來實現我們的目標。

3. 專題進行方式

在初期思考如何進行推薦時，一般最簡單的方式是使用 rule-base 來推薦使用者喜歡的服裝搭配。然而，rule-base 的限制是其規則是粗略的，衣服的特徵和整體搭配出來的感覺可能無法用訂好的規則來詮釋。為了更貼近使用者的喜好，我們想設計卷積神經網路來擷取圖片更細微的特徵，並搭配 supervised learning 來預測使用者的喜好程度並推薦服裝搭配。

而在設計使用時，我們預想到單純使用 supervised learning 來進行預測為了獲取足夠的訓練資料，必須讓使用者進行許多的手動標注，為此我們決定在推薦系統中導入 semi-supervised learning，以此來降低使用者手動進行的標注還能維持訓練資料的數量。

隨後發現在初期冷啟動時可能造成使用者對於推薦準確度的表現不滿意，因而在推薦初期加入了 rule-base 所推薦的服裝搭配，使初期也能推薦出使用者可能較為感興趣的服裝搭配。

綜上所述，我們實作了一個 rule incorporated semi-supervised learning 服裝搭配推薦系統（Sec.3.3），其中採用了多種技術包括 rule-base 服裝推薦系統（Sec.3.1）和 semi-supervised learning（Sec.3.3）等技術，以提高推薦的準確度和符合使用者的喜好。這使得在初期和後續階段，系統能夠更好地推薦出合適的服裝搭配給使用者。

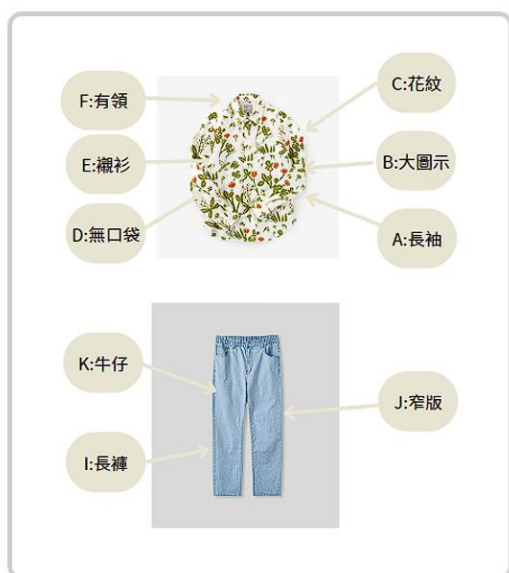
3.1 Rule-base 服裝推薦系統

我們首先從各大服飾品牌收集了衣服和褲子的樣本，並進行人工標註。這些標籤分為九個大類，總共有36個標籤（如圖一、圖二）。然而，在進行人工標籤時，由於衣服和褲子的數量太多，無法全部標註，因此我們選擇使用已經標註好的部分衣褲作為訓練資料，並根據不同的大類訓練了九個 CNN 分類器，這些分類器能夠自動進行標籤上的預測。

Checkbox Group
☐ 短袖 ☒ 長袖
☒ 大圖案 ☐ 小圖案 ☒ 無圖案
☒ 卡通 ☒ 花紋 ☒ 風景 ☒ 格紋 ☐ 純字 ☒ 素色
☒ 單口袋 ☐ 無口袋 ☐ 腰部口袋 ☒ 雙口袋
☒ polo衫 ☐ 衛T ☐ 風衣 ☒ 襯衫 ☒ T-shirt
☒ 圓領 ☒ V領

圖一、使用者衣服標籤喜好表

圖二、使用者褲子標籤喜好表



圖三、服裝搭配標籤

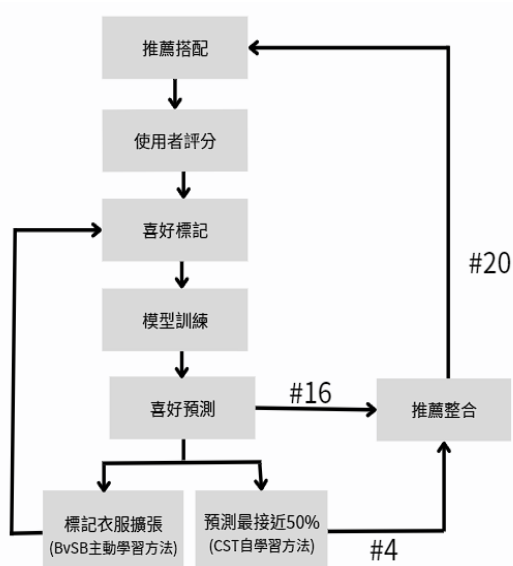
在完成圖片的標籤工作後，我們設計了一個標籤喜好的表單（如圖一和圖二），讓使用者選擇他們喜歡的標籤。系統讀取表單後，根據符合表單的程度來對服裝搭配評分。例如，圖三中，衣服部分符合長袖、大圖案、花紋、襯衫和有領這五個標籤，所以衣服部分得到了五分。而褲子部分符合長褲和牛仔褲這兩個標籤，所以褲子部分得到了兩分。該服裝搭配的總分為衣服部分的分數乘以褲子部分的分數，即十分。當沒有比十分更高的服裝搭配時，這個服裝搭配將被推薦出來。

3.2 Semi-supervised Learning

關於 semi-supervised Learning 的部分所使用的架構是參考《基于主动

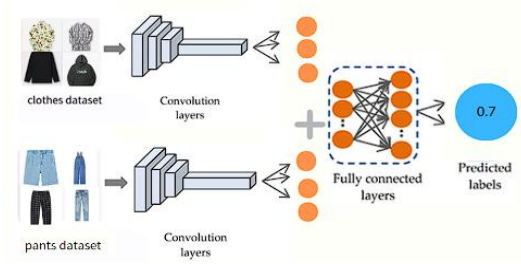
学习和半监督学习的多类图像分类》[4] 這篇的模型來進行，其中主要分為 BvSB（Best vs second-best）的主動學習方法用來挑選預測中不易判斷喜歡與不喜歡的組合（圖四預測最接近50%）給使用者進行手動的標註以及 CST（Constrained self-training）的自學習方法來將預測中最確定結果的組合進行自動標註（圖四標記衣服擴張），用以減少使用者進行標註的數量。

其流程（圖四）為每回合推薦20件服裝搭配，包含16件針對使用者喜好的服裝推薦和4件最適合做為模型自學習的服裝搭配，供使用者作評分，評完分後做喜好標記更新，作為訓練資料丟入模型做訓練。訓練完畢後，再使用模型將未標記的服裝搭配做喜好程度的預測，將預測喜好程度前16高的服裝搭配和喜好程度最接近50%的4件服裝搭配（BvSB）整合起來做為下一輪的服裝搭配推薦，同時將最有可能預測正確的搭配（CST），做自學習的標記衣服擴張。接下來會介紹較為重要的模型訓練、標記衣服擴張、預測最接近50%三個區塊。



圖四、semi-supervised learning 流程圖

3.2.1 模型訓練



圖五、預訓練的 CNN 提取服裝搭配的特徵放入 DNN 做訓練

3.2.1.1 圖片特徵擷取

由於數據量有限，無法訓練出具有較好特徵提取能力的卷積神經網絡。因此，我們選擇使用預訓練的 VGG16 模型[5]來處理圖片。該模型在大型數據集上進行了訓練，具備良好的特徵提取能力。我們將服裝圖片分為衣服和褲子兩個部分，分別將它們輸入預訓練模型中（圖五 Convolution layers），並提取第13層（卷積層的末尾）的特徵值作為衣服或褲子的特徵，以供後續的訓練使用。

3.2.1.2 訓練深度神經網絡（DNN）

我們以完整的服裝搭配作為單位，將衣服和褲子的特徵結合起來，生成該服裝搭配的新特徵表示。在獲得服裝搭配的特徵後，構建了一個未經訓練的深度神經網絡（DNN）模型（圖五 Fully connected layers），該模型包含三個全連接層，並使用 RMSE（Root Mean Squared Error）作為我們的 loss function，其中 $f()$ 為我們的模型， x^n 為測試資料的特徵， $f(x^n)$ 為預測結果， $\hat{y}^n$ 為測試資料的實際喜好程度。

$$RMSE = \sqrt{\frac{1}{N} \sum_{n=1}^N (f(x^n) - \hat{y}^n)^2}$$

我們將提取到的服裝搭配特徵作為輸入，通過 DNN 模型進行訓練，以預測使用者對該套服裝推薦的喜好程度。

3.2.2 BvSB 主動學習方法

由於在我們的專題中預測出的結果會是介於喜歡與不喜歡之間的值，因此我們所選擇分類結果最不確定的樣本就會是機率最接近50%的樣本，為了增加樣本的多樣性並提供更多的資訊來訓練，因此會選擇離已標註樣本最遠也就是相似度最低的樣本點為優先選擇。

所有未標註樣本 X_i^U 都會介於喜歡(1)及不喜歡(0)之前，因此每個未標註樣本 X_i 都會在模型預測中得到一個介於0跟1之間的預測值 $p(X_i^U)$ ，而樣本 X_i^U 與已標註樣本的相似度寫為 $\text{sim}(X_i^U, X_j^L)$ ，並於相似度前乘上一個係數 α 來控制相似度的影響程度， $\text{score}(X_i^U)$ 則為未標註點 X_i^U 經過公式計算後所得到的分數值。

$$\text{score}(X_i^U) = |p(X_i^U) - 0.5| + \alpha * \max_{X_j^L} \text{sim}(X_i^U, X_j^L)$$

並選出 K_{BvSB} 個 $\text{score}(X_i^U)$ 最低的未標註樣本 X_i^U 給使用者進行手動標註。

3.2.3 CST 自學習方法

CST 自學習方法的目標為喜好標記的擴張，以此降低使用者手動標註的樣本數量，因此採用自學習的方法，來將在確保預測正確性同時可以提供給預測面足夠信息的未標註樣本加入至訓練樣本之中。

為了確保預測正確性，以 y_{best} 表示該未標註樣本點最有可能之喜好預測結果。

$$y_{Best} = \begin{cases} 1, & \text{if } p(X_i^U) > 0.5 \\ 0, & \text{otherwise} \end{cases}$$

以 $k(y_{Best}|X_i^U)$ 表示該樣本點對於 y_{best} 的確定性為多少。

$$k(y_{Best}|X_i^U) = \begin{cases} p(X_i^U), & \text{if } y_{best} = 1 \\ 1 - p(X_i^U), & \text{otherwise} \end{cases}$$

並以 y_{nn} 表示與 X_i^U 最相似之已標註樣本點 X^* 其喜好預測結果

$$y_{nn} = L(X^*)$$

$$\text{where } X^* = \underset{X_j^L}{\operatorname{argmaxsim}}(X_i^U, X_j^L)$$

因此其約束條件為

$$(1) \quad k(y_{Best}|X_i^U) > \text{threshold}$$

而 $0 \leq \text{threshold} \leq 1$ 為設置的閾值，用以設定對其分類結果確定性的篩選

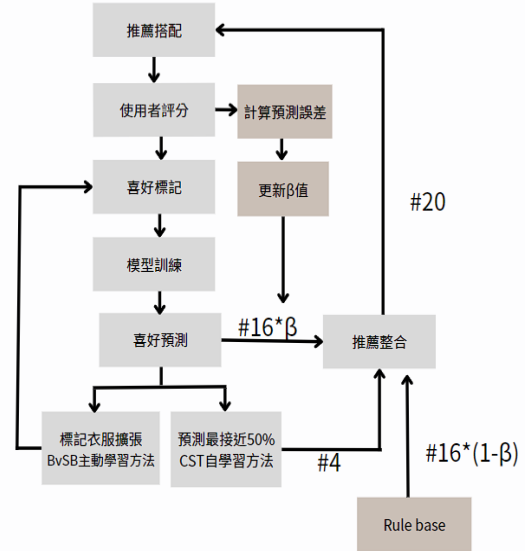
$$(2) \quad y_{Best} = y_{nn}$$

第二個條件用意確定與其最相似的已標註樣本與該未標註樣本的喜好預測結果一致。

將所有符合(1)(2)的 X_i^U ，放入子集 S 中，並從子集 S 中選出 $p(y_{Best}|X_i^U)$ 最小的 K_{CST} 個樣本加入到訓練樣本之中，來確保在預測正確性的情況下依舊可以提供一定的訊息量給預測面。

3.3 Rule Incorporated Semi-supervised Learning

承自 semi-supervised learning（圖四），rule incorporated semi-supervised learning 為了解決 semi-supervised learning 在前期準確度比較低的情況，在前期混入由 rule-base 推薦的組合來解決（圖六棕色區塊）， β 則是用來控制 semi-supervised learning 跟 rule-base 推薦的比例。



圖六、rule incorporated semi-supervised learning 流程圖

在第一輪推薦時由 semi-supervised learning 所推薦的只要一組，因此 β 的初始值為 $\frac{1}{0.8*N}$ ，而在後面輪次當中， β 將透過前一輪推薦的 expectation loss 來進行反比成長，因此使用的為二次式的線性曲線，其公式為：

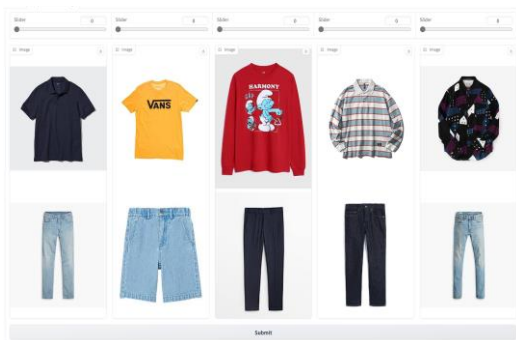
$$\beta_t = \beta_{t-1} * (1 + (el - 1)^2)$$

4. 實作結果

服裝推薦系統的介面由 gradio[6]實作，gradio 是一種將機器學習模型與他人分享的方式之一，創建一個可互動的 app，讓使用者在瀏覽器中使用。

此 app 使用方法非常簡單，只要先填寫表單（圖七）並提交，在移到服裝推薦的畫面（圖八）後系統會開始推薦衣服，每回合推薦20套服裝搭配，一次提供五套 服供使用者做評分，當每回的20套使用者評完分後，會進行模型的訓練再推出新的20套服裝搭配。

圖七、表單畫面



圖八、推薦衣服畫面

5. 主要成果與評估

5.1實驗設計

一共收集樣版衣數量為1823件，樣版褲數量為1301件，邀請了20位的受測者參與實驗。

diversity（推薦種類多樣性）：

當服裝搭配的9個標籤完全相同時，他們被視為同一類，**diversity** 是用來計算推薦出來共有幾種不同類的服裝搭配，以此看出推薦種類的多樣性。

hit rate（喜好比例）：

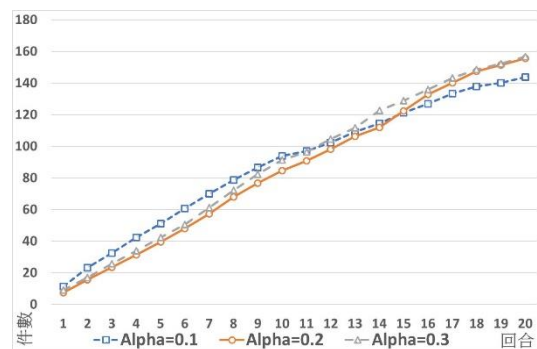
計算每回合中推薦服裝搭配中使用者喜歡的比例，每回合有20件衣服，若服裝搭配喜好程度高於0.5就將其視為喜歡。

5.2實驗結果

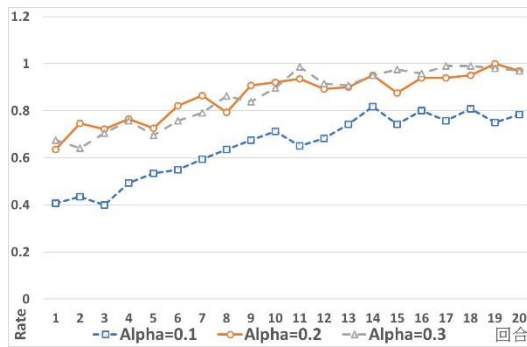
5.2.1 使用不同的 α 值得推薦

在 semi-supervised learning 的 BvSB 中有一個參數 α 用來控制已標註樣本與未標註樣本相似度的影響程度，我們想做實驗來比較不同的 α 會對 diversity（圖九）和 hit rate（圖十）有怎樣的影響，來選定最終的 α 值，其中圖九的 x 軸為回合數，y 軸為搭配的件數，圖十的 x 軸為回合數，y 軸為 hit rate。

在 diversity 中可以看到（圖九）當 α 越大時推薦的種類就會越多，當 $\alpha=0.1$ 時可以看到多樣性明顯的比0.2及0.3要小，而0.2和0.3之間雖然0.3在初期成長的較快，但在後期也相差不多，然而在 hit rate 中（圖十） $\alpha=0.2$ 比 $\alpha=0.3$ 早達到所設定的門檻0.8（因為每輪僅有8成是推薦確定喜歡的套裝，因此門檻為0.8），在這次的實驗中由於多樣性的差距沒有特別明顯，而 hit rate 在 $\alpha=0.2$ 時更快的達到我們所設定的門檻，因此 α 值的選擇為0.2。



圖九、依據 α 值不同進行的 diversity 比較

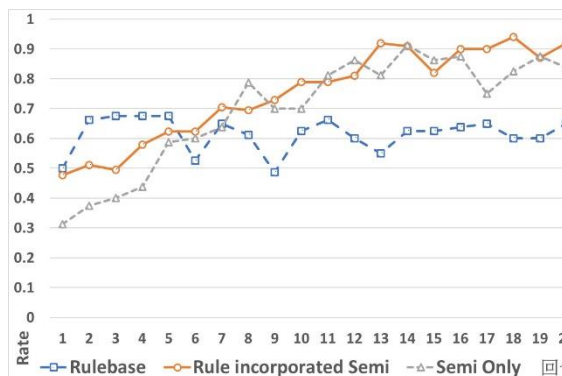


圖十、依據 α 值不同進行的 hit rate 比較

5.2.2 根據推薦系統的不同

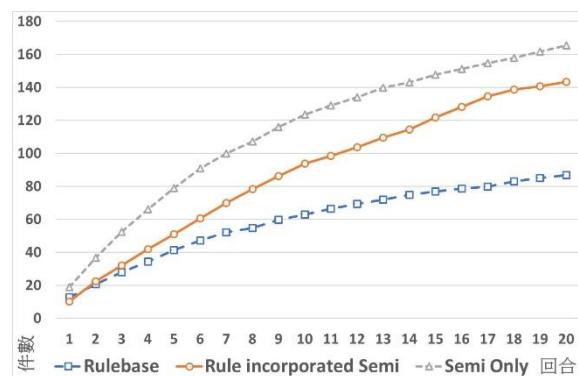
為了解三種推薦系統中哪一種推薦系統有較好的表現，因此比較它們的 hit rate (圖十一)、diversity (圖十二) 和推薦且喜歡的 diversity (圖十三) 的表現，圖十一的 x 軸為回合數，y 軸為 hit rate，圖十二、圖十三的 x 軸為回合數，y 軸為搭配的件數。

在 hit rate 中可以看到 (圖十一)，rule incorporated semi-supervised learning 和 semi-supervised learning 在第11回合後皆能超越0.8，但純 semi-supervised learning 前四回合的表現卻明顯低於 rule incorporated semi-supervised learning，因此 rule incorporated semi-supervised learning 在 hit rate 上明顯優於其他兩者。

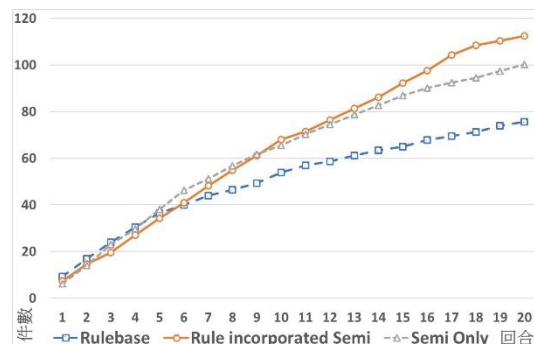


圖十一、依據推薦系統的不同進行的 hit rate 比較

而在 diversity 的部分 (圖十二) semi-supervised learning 雖然高於其他兩者，但在推薦且喜歡的 diversity 比較中 (圖十三) rule incorporated semi-supervised learning 卻又能贏過 semi-supervised learning，代表 semi 的 diversity 會高是源於初期偏隨機的推薦，此隨機的推薦離使用者喜好較遠，我們認為推薦且喜歡的多樣性比只看多樣性來的重要，因此這邊已 rule incorporated semi-supervised 作為最終推薦系統的選擇。



圖十二、依據推薦系統的不同進行的 diversity 比較



圖十三、依據推薦系統的不同進行推薦且喜歡的 diversity 比較

6. 結語與展望

目前，此項專案主要集中在推薦系統的開發，介面功能尚未完善。我們計劃進一步增加以下功能：儲存服裝搭配和根據使用者喜好固定上衣來推薦褲子。

這樣這個服裝搭配推薦系統將能夠成為一個完整且獨立運行的 App。

我們也希望能夠與服飾廠牌合作，這樣在收集衣物樣本時，我們可以直接向廠商索取照片，節省了自行搜尋樣本衣物所需的大量時間。同時，在推薦服裝搭配時，我們還希望能夠提供購買連結，讓使用者更快速地購買所喜歡的衣物。這樣一來，我們可以更好地與廠牌合作，提供更完整的服務給使用者。

7. 銘謝

感謝指導教授在開會時給予各種方面的看法，不管是在製作過程中的想法以及最終的簡報製作上，都修正了我們很多的錯誤，並給予我們許多解決問題的方案，讓整體作品更加完善，還有互相加油的組員們，我們都不會忘記這段雖然很辛苦但非常重要的專題時光。

8. 參考文獻

[1] lativ

<https://www.lativ.com.tw/MEN>

[2]XZ (closet)

https://play.google.com/store/apps/details?id=fashion.style.coordinate.app.xz&hl=zh_TW&gl=US

[3] WEAR: 時尚穿搭

<https://wear.tw/>

[4]陈荣, 曹永锋, 孙洪. (2011) 基于主动学习和半监督学习的多类图像分类. 自动化学报

<http://www.aas.net.cn/article/id/17514>

[5] Srikanth Tammina. (2019) . Transfer learning using vgg-16 with deep convolutional neural network for classifying

https://www.researchgate.net/profile/Srikant-h-Tammina/publication/337105858_Transfer_learning_using_VGG-16_with_Deep_Convolutional_Neural_Network_for_Classifying_Images/links/5dc94c3ca6fdcc57503e6ad9/Transfer-learning-using-VGG-16-with-Deep-Convolutional-Neural-Network-for-Classifying-Images.pdf?_sg%5B0%5D=started_experiment_milestone&origin=journalDetail&rtd=e30%3D

[6]gradio

<https://gradio.app/>