

# Team 47

r11944039 鍾元皓, r11921102 曾峻麒, r11944051 黃彥豪

## *Abstract*

這次期末專題題目為 Hahow 推薦課程系統，其中分為預測客戶接下來想購買的課程和有興趣的主題，所以主要會專注在應用適合推薦相關資訊的模型，這類型常見的模型，有 DSSM、YoutubeDNN、YoutubeDSSM、FacebookEBR、MIND 等等，而預測課程和主題所適用的 model 也有所不同。

預測課程方面，在實驗多種模型後，選擇使 DSSM 這款模型，較符合我們的資料集特性，而使用傳統的 DSSM 準確性表現太優異，則我們做了一些部分的修改。

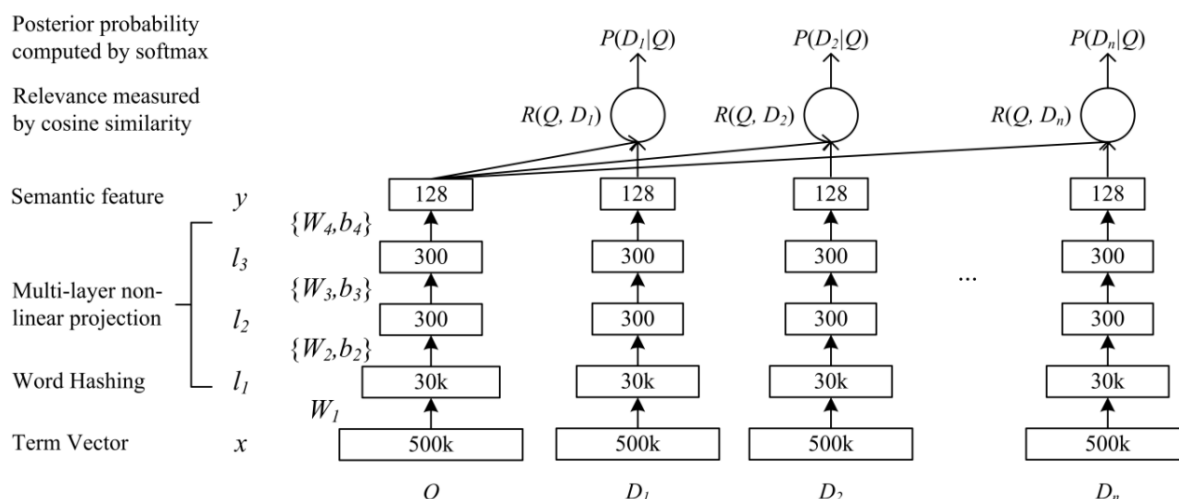
預測主題方面，使用 DSSM 效果沒有達到預期的效果，所以我們使用自己造的 model，根據每位使用者預測出對於不同課程的喜好機率。

## *Introduction*

在試過眾多模型中，效果最好的是 DSSM (Deep Structured Semantic Model)，而 DSSM 是由微軟於 CIKM 在 2013 年提出的模型，該模型主要用來解決 NLP 領域中，語義相似度問題。主要的原理是透過 deep learning 將我們要搜尋 query 和匹配的 document 映射到共同維度的空間，透過計算最大化 query 和 document 語義向量之間的 cosine similarity，從而訓練得到隱含語義模型，也就是 query 特徵的 embedding 和 document 特徵的 embedding，所以藉由此方

式就可以獲取語句的語義向量，預測兩句話的語義相似度。

再來有個特色就是這模型的訓練速度非常快，造就這模型其實在實際應用中非常常見，是因為他在大量的數據進行初步的預測速度可以達到很快，雖然效果沒有到非常好，但對於一般場景而言，效果也夠用了。



**Figure 1:** Illustration of the DSSM. It uses a DNN to map high-dimensional sparse text features into low-dimensional dense features in a semantic space. The first hidden layer, with 30k units, accomplishes word hashing. The word-hashed features are then projected through multiple layers of non-linear projections. The final layer's neural activities in this DNN form the feature in the semantic space.

首先先介紹傳統 DSSM 的基本架構，Figure 1 為 DSSM 的架構，從圖可以看出，該架構相較現今眾多語意模型相較簡單，是一個由幾層 DNN 組成的架構，第一層 Term Vector 為輸入的 embedding 向量，第二層 Word Hashing 為為了解決 Term Vector 太大問題，對 bag of word 做降維，第三層 Multi-layer nonlinear projection 表示 Deep Learning Hidden Layer，第四層 Semantic feature 為最終的 embedding 向量，第五層 Relevance measured 為 Query 和 Document 之間的 cosine similarity，這個過程可以看作每一個 query 和 document 之間相似分數，最後一層為 softmax。

## Related Work

關於推薦系統的模型，除了 DSSM 這個模型，我們也試過其他模型，像是 YoutubeDNN, YoutubeSBC, MIND，從表現結果來看，DSSM 最為凸出，所以我們就打算把這次的焦點放在 DSSM 上。

## Approach

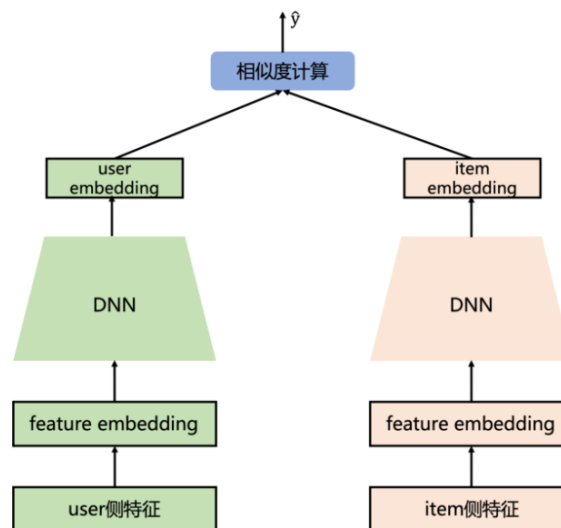


Figure 2

而在這次的 Final Project 中，我們把主要的輸入設定為 User 和 Course，而 Figure 2 就是我們這次的架構。對於每個塔分別是一個 DNN 結構，由兩側把特徵輸入，利用 User 和 Course 的特徵進行訓練，最終得到 User Side 的 Embedding 和 Course Side 的 Embedding，然後計算兩者之間的 Cosine Similarity，因此對於 User 和 Item 兩側最終得到的 Embedding 維度需要保持一致，也就是最後一層全連接 hidden unit 個數相同。

這種架構造就 DSSM 在 Training 時速度很快，是因為模型結構簡單，且兩側沒有特徵交叉，但這也帶來了問題，就是這種雙塔的結構無法考慮兩側特徵之間的交互訊息，在一定程度上犧牲掉模型的部分準確率。

對於準確率的問題，我們有使用一些方法提升，根據 *Sampling-Bias-Corrected Neural Modeling for Large Corpus Item Recommendations* 這篇論文，這篇論文提到了兩種做法分別為 Normalization 和 Temperature，Normalization 為對 user side DNN 和 item side DNN 的 input embedding 進行 L2 normalization，Temperature 為在 Normalization 之後，除以一個固定的參數，而這個過程在論文中命名為 Temperature。

再來還有使用一個方法，就是我們有對要被訓練的資料做 Negative Sampling，主要的用意為避免非熱門的課程缺少關注，因為對於這一類的相似度比較 model，樣本的選擇上會很大程度影響模型的結果。

```
class MLP(torch.nn.Module):
    def __init__(self, num_classes = 91):
        super(MLP, self).__init__()
        self.feature = torch.nn.Sequential(
            torch.nn.Linear(5, 16),
            torch.nn.BatchNorm1d(16),
            torch.nn.ReLU(inplace = True),
            torch.nn.Dropout(0.2),

            torch.nn.Linear(16, 32),
            torch.nn.BatchNorm1d(32),
            torch.nn.ReLU(inplace = True),
            torch.nn.Dropout(0.2),
        )
        self.classifier = torch.nn.Sequential(
            torch.nn.Linear(32, 16),
            torch.nn.BatchNorm1d(16),
            torch.nn.ReLU(inplace = True),
            torch.nn.Dropout(0.2),

            torch.nn.Linear(16, num_classes),
        )

    def forward(self, x):
        x = self.feature(x)
        x = self.classifier(x)
        x = F.log_softmax(x, dim=1)

        return x
```

Figure 3

在預測 Course 方面，使用上面方法效果還不錯，但將上面模型用來預測 Topic，效果沒有預期的好，所以我們最後對於 Topic 使用我們自己做的簡易模型，Figure 3 為我們使用的結構，能在最後輸出每堂 Course 對於 User 的機率。

## Experiments

### 1. Benchmark

Table 1 分別是不同的模型的 baseline — DSSM、MLP、YoutubeSBC。

Model	DSSM	MLP	YoutubeSBC
Loss function	BCELoss	CrossEntropyLoss	BCELoss
Optimizer	Adam	SGD	Adam
Learning rate	1e-4	1e-1	1e-4
Epochs	5	50	10
Batch Size	1024	4096	1024
Topk	50	50	50

Table 1：依序為 DSSM、MLP、YoutubeSBC

### 2. 不同 Model 比較

在 Seen Course 任務中，model 使用 DSSM 與 YoutubeSBC 的 Accuracy 比較。

Model	DSSM	YoutubeSBC
Accuracy	0.13358	0.03277

在 Seen Topic 任務中，model 使用 MLP 與 DSSM 的 Accuracy 比較。

Model	MLP	DSSM
Accuracy	0.22027	0.13577

### 3. Epoch 與 Metrics (Loss、Accuracy) 關係圖

底下是 Seen Course (Figure 4)、Unseen Course (Figure 5)、Seen Topic (Figure 6)、Unseen Topic (Figure 7)。在不同 epoch 下的 Loss 與 Accuracy。Seen Course 任務與 Unseen Course 任務皆使用 DSSM，而 Seen Topic 任務與 Unseen Topic 任務皆使用 MLP 進行實驗。

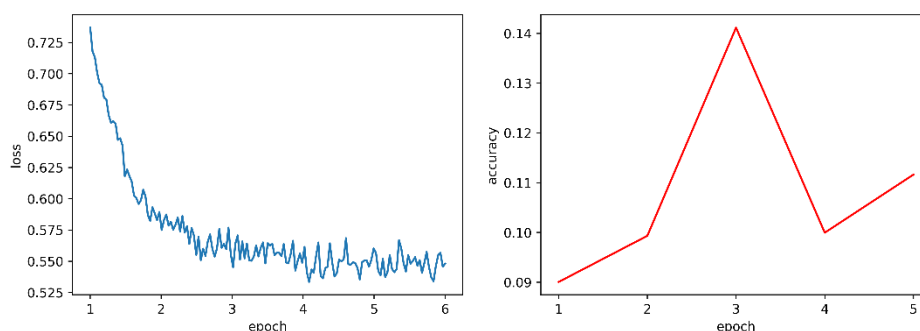


Figure 4 : Seen Course

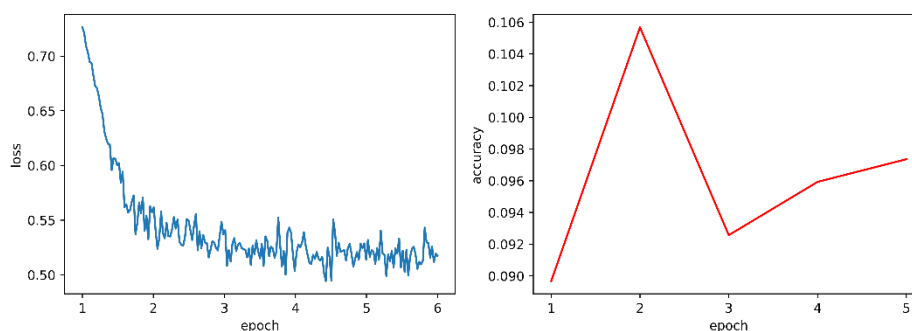


Figure 5 : Unseen Course

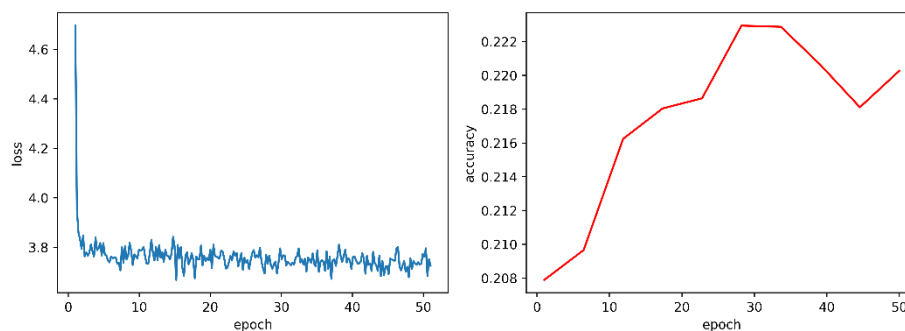


Figure 6 : Seen Topic

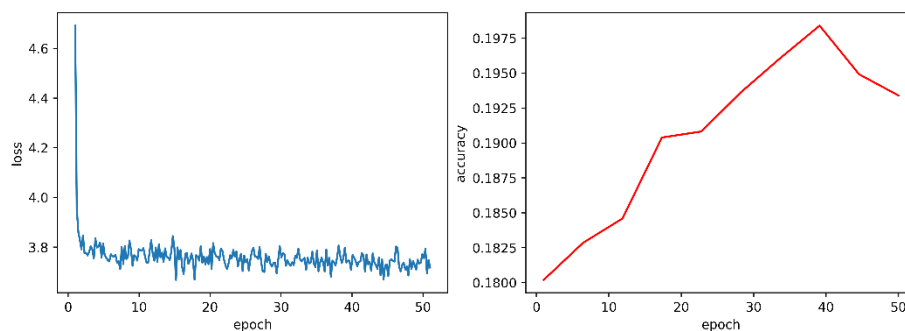


Figure 7 : Unseen Topic

#### 4. 不同 Learning Rate 比較

在 Seen Course 任務中，使用 DSSM 的 Learning rate 比較。

Learning Rate	1e-3	1e-4	1e-5
<b>Accuracy</b>	0.0731	0.13358	0.07568

在 Unseen Course 任務中，使用 DSSM 的 Learning rate 比較。

Learning Rate	1e-3	1e-4	1e-5
<b>Accuracy</b>	0.05175	0.09881	0.8435

#### 5. 不同推薦數量(topK)比較

在 Seen Course 任務中，使用 DSSM 的 TopK 比較。

Topk	10	50	300
<b>Accuracy</b>	0.09689	0.13358	0.03397

## Discussion

在 Course 任務中，我們發現兩點，第一，在 ml-1m 數據集中(我們有使用其他數據集測試，ml-1m 是一個電影觀看的數據集)， YoutubeSBC 的準確率為 0.1783 而 DSSM 為 0.1474，但在 Hahow 數據集中，使用 DSSM 的準確率更高，可以推論出在不同數據集中，表現較好的模型不一定較好。第二，當 Loss 收斂時，Epoch 3 的 Accuracy 來到最高，可以觀察到 Epoch 的次數選擇可以根據 Loss 的收斂時間挑選。

在 Topic 任務中，我們發現兩點，第一，分類模型 MLP 準確率為 0.22027，推薦模型 DSSM 則是 0.13577，準確率高出八成，可以推論 Topic 任務更適合使用分類模型而非推薦模型。第二，不同 Epoch 比較，可以看出由 Loss 的收斂選擇最合適的訓練次數。

## Conclusion

根據實驗，我們整理出最好的配置，如下表所示。

	Seen Course	Unseen Course	Seen Topic	Unseen Topic
<b>Model</b>	DSSM	DSSM	MLP	MLP
<b>Learning rate</b>	1e-4	1e-4	1e-1	1e-1
<b>Epochs</b>	3	2	25	35
<b>Batch Size</b>	1024	1024	4096	4096
<b>TopK</b>	50	50	50	50



## Kaggle :

- Seen User Course Prediction
  - 0.14474 (12/62)
- Unseen User Course Prediction
  - 0.10250 (18/61)
- Seen User Topic Prediction
  - 0.23485 (51/62)
- Unseen User Topic Prediction
  - 0.21987 (52/60)

## *Work Distribution*

Student ID	R11921102	R11944039	R11944051
<b>Contributions</b>	<ul style="list-style-type: none"><li>• Inference</li><li>• Tuning parameters</li><li>• Write report</li><li>• Oral Presentation</li></ul>	<ul style="list-style-type: none"><li>• Training</li><li>• Tuning parameters</li><li>• Write report</li><li>• Oral Presentation</li></ul>	<ul style="list-style-type: none"><li>• Data preprocess</li><li>• Tuning parameters</li><li>• Write Report</li></ul>