

國立政治大學資訊科學系

Department of Computer Science

National Chengchi University

碩士論文

Master's Thesis

基於異質性資訊網路表示法學習之電子商務

推薦系統

E-commerce Recommendation Systems Based

on Heterogeneous Information Network

Embedding

研究生：張伯新

指導教授：蔡銘峰

中華民國一百零七年七月

July 2018



基於異質性資訊網路表示法學習之電子商務推薦系統  
E-commerce Recommendation Systems Based on  
Heterogeneous Information Network Embedding

研究生：張伯新      Student：Bo-Sin Zhang  
指導教授：蔡銘峰      Advisor：Ming-Feng Tsai

國立政治大學  
資訊科學系  
碩士論文

A Thesis  
submitted to Department of Computer Science  
National Chengchi University  
in partial fulfillment of the Requirements  
for the degree of  
Master  
in  
Computer Science

中華民國一百零七年七月  
July 2018

# 致謝

兩年過去，看著自己的論文完成，心中也感到踏實。想當初也算是誤打誤撞進入政大研究所，從網路前端及繼續進修，我選擇了後者。現在看起來，除了對於專業領域的更加認識，也或多或少的確信了自己的定位。當然，這一路要感謝不少師長及親友。

首先要先感謝我的指導老師-蔡銘峰老師。幸虧有老師大一的計算機程式設計課程，就開始奠定我後來的程式基礎。到後來的大三大四的 Exchange World 專案，老師也一路陪伴著我們。研究所的這段求學過程中，提供了我不少機會發揮。多虧老師引進，讓我有機會能在 Appier 有難得長達一年半的產學實習，讓我學到不少實務上的經驗。在碩論題目上，老師也給我很多空間自由發揮，做我自己有興趣的題目，指導上自由也不失嚴謹。同時也跟上不少業界新訊，對我這種偏 Engineer 取向的研究生受益良多。跟著老師學習，的確很幸福。

此外，也有許多老師也對我幫助甚多。王釗茹老師帶領 Sean、Leo、和我完成研討會論文。讓只有參與實驗的我，能夠有機會去見見世面，看看天花板有多高。這著實刺激了我自己的競爭心，我也開始比較去鑽研許多背後的原理。另外也要感謝劉昭麟老師，雖然現在不是 MIG 實驗室的一份子，大學部的時候專題讓我接觸了自然語言處理，也作為我現在的研究的基礎。當然，其他政大資科老師們的諄諄教誨，也是我這一路學習的過程更加順利。

我也感謝實驗室的夥伴們。不管是超 carry 的志明，402 守護者俊翔和文毓，碩一的昇芳、裕勝、聖祐、中研院的 RA 大家庭。當然還有我的家人以及好朋友們，有你們力量的結合，才能讓我在研究過程中，也增添不少青春歡樂。碩士生的旅程將告一段落，也將在人生面臨更多的挑戰。希望之後能繼續踏穩腳步，發揮自身長才。LIFE WILL CHANGE！

張伯新

國立政治大學資訊科學系  
July 2018

## 中文摘要

近年來由於龐大的資料量，電子商務的商品推薦變成一項具有挑戰性的工作。因此，我們使用了異質性資訊網路表示法學習（Heterogeneous Information Network Embedding），能夠將網路上不同類型的節點及之間的關係投影到低維度向量空間，以進行電子商務相關的產品推薦工作。本論文提出了一個基於異質性資訊網路表示法學習的電子商務推薦系統，能夠更有效地整合額外資訊（Meta Information）。首先，我們萃取標題的詞彙並與商品鏈結，再加入使用者過去的歷史紀錄轉換成異質性資訊網路。從這個網路中，我們可以使用各式各樣的網路表示法學習方法訓練，並在同一個向量空間中學習使用者及商品的表示式。除此之外，我們更將學習到的表示法當做特徵值，結合矩陣分解（Matrix Factorization）及排序學習（Learning to Rank）的做法，來達到有效地推薦商品的目的。在實際電商 Amazon 的資料集中，此論文的方法能使推薦的效果有所提升。另外，此方法也能夠有效改善電子商務推薦系統所重視的覆蓋率（Coverage）表現。

## **Abstract**

In recent years, E-commerce product recommendation has been a challenging task due to its data sparsity and volume. Heterogeneous information network embedding encodes the node information into low-dimensions vector space from different types of nodes and their corresponding relations. In this paper, we propose an E-commerce product recommendation method based on the heterogeneous network embedding. First, we incorporate words from product title as the attributes of the item. Then, we transform words, and user behavior into heterogeneous network for E-commerce. For this network, we use various network embedding methods to learn both user and item representations in the same latent space. Moreover, we integrate the learned embedding as the features into Matrix Factorization and Learning to Rank. The experiment results show that we improve the recommendation quality on Amazon dataset. Also, we demonstrate our model can perform better in terms of coverage, the focus of E-commerce recommendation systems.

# 目錄

致謝	1
中文摘要	2
Abstract	3
第一章 緒論	1
第二章 相關文獻探討	4
2.1 推薦系統	4
2.2 網路表示法學習	5
第三章 研究方法	7
3.1 問題定義	7
3.2 異質性資訊網路表示法學習	8
3.2.1 建圖策略	8
3.2.2 Deepwalk	9
3.2.3 LINE	10
3.2.4 HPE	11
3.2.5 metapath2vec	12
3.3 推薦模型	13
第四章 實驗結果與討論	15
4.1 資料集	15
4.2 實驗設定	16
4.3 評估標準	18
4.4 實驗結果	18
4.4.1 準確率及召回率表現	18
4.4.2 覆蓋率表現	20
4.4.3 建圖策略比較	22
4.4.4 案例分析-網路表示法學習	22
4.4.5 案例分析-推薦系統	25
第五章 結論	27

# 圖目錄

圖 1.1	此論文的推薦系統架構圖 . . . . .	2
圖 3.1	電子商務的異質性資訊網路 . . . . .	8
圖 3.2	LINE的相似度關係 . . . . .	10
圖 3.3	Meta-path 的序列生成例子 . . . . .	12
圖 4.1	網路表示法的視覺化呈現 . . . . .	24



# 表目錄

表 2.1	使用者－商品的評分數據範例 . . . . .	4
表 4.1	Amazon 資料集數據統計 . . . . .	15
表 4.2	top-10 推薦結果 (Movies, Home, Electronics) . . . . .	19
表 4.3	top-10 推薦結果 (Cell Phones, Clothing, CDs) . . . . .	19
表 4.4	覆蓋率評估 . . . . .	20
表 4.5	有無文字建圖方式的推薦結果比較 (Movies, Home, Electronics) . . . . .	21
表 4.6	有無文字建圖方式的推薦結果比較 (Cell Phones, Clothing, CDs) . . . . .	21
表 4.7	網路表示法學習的案例分折 . . . . .	23
表 4.8	推薦系統的案例分折 . . . . .	25

# 第一章

## 緒論

在網際網路時代，人們的購買行為已經開始轉變。漸漸地習慣透過網路上琳琅滿目的物品資訊、方便的服務平台來購買商品。隨著電子商務平台的蓬勃發展，商品資料及使用者紀錄被大量地產生。如何將使用者及商品有效地連結，變成一項重要的研究議題。因此，推薦系統（Recommendation System）在目前電子商務平台裡扮演著重要的角色。透過演算法或機器學習模型，幫助使用者找到或發現有興趣的商品。推薦系統通常分為三種，協同過濾（Collaborative Filtering）[1]、基於內容推薦（Content-Based Recommendation）[2]與混合式推薦（Hybrid Recommendation）[3]。協同過濾藉由用戶的歷史紀錄，像是評分、點擊、購買等行為，再利用興趣類似的用戶群的喜好來推薦。基於內容推薦利用物品的屬性，像是標題、類別、品牌等離散特徵推薦類似的物品。混合式推薦則是融合上述兩項方法，以達到更好的推薦效果。而隨著網路服務的快速成長，更多的物品資料及紀錄能被推薦系統所利用。雖然這些資料隱含著許多豐富的資訊，但如何同時將龐大的資料及之間的關係儲存，並考慮計算複雜度的成本，是一個相當具有挑戰性的任務。

由於實作上的彈性及優異的效果，表示法學習（Embedding）[4]在人工智慧的許多問題中被廣泛地使用。表示法學習的技術主要是將一個物件的屬性或鄰近資訊壓縮成一個向量。例如像在自然語言處理領域中，word2vec [5]透過類神經網路學習每個字詞的向量表示式。由於方便應用及具有延展性，已經在許多機器學習任務裡廣泛地應用。而在電子商務裡，如何學習每個使用者、商品的表示法應用在推薦上，仍是目前實務上的一大挑戰。

社群網路分析中，網路表示法學習（Network Embedding）是目前新的研究主題方向。我們可以把用戶及連接關係建成一張圖。基於這張圖，我們可以運用類似於詞向量的想法，將網路節點投影到低維向量空間中。目前有許多基於word2vec的網路表示式學習框架，像是DeepWalk [6]、LINE [7]、node2vec [8]等方法，都被廣泛應用於資料探勘、資訊檢索的工作上。但是，大部份的網路表示法學習框架，往往是應用在同質性網路，而無法有效地將不同類型的節點投影到

同一個空間。

異質性網路（Heterogeneous Information Network）是一種包含了多種型態的節點及鏈結的有向圖。在實務應用中，多數系統都具有多種型態的節點及關係交互作用。像是社群媒體就有像是使用者、文章、標籤、地點等多種類型節點，以及他們的複雜交互關係。像是使用者可以對一篇文章按讚、一篇文章有著多個標籤等關係。而推薦系統也有使用者、商品、評論、品牌等類型節點可以利用。透過異質性網路建模是相當自然的一件事情。且這些數據的多樣性可以為目前現有的模型增添更多豐富的資訊。

然而在電子商務中，使用者的數量及商品的數量都是相當龐大的。實務上，一位使用者往往只會與一個到數個的商品有互動紀錄，而造成資料的稀疏性（Sparsity）。傳統的推薦模型如矩陣分解（Matrix Factorization），效果往往會受到稀疏性的影響而降低。而許多現有的模型中，往往是根據均方根誤差（Root Mean Square Error）做評分。這樣只往往考慮到推薦系統的準確性而忽略了實際應用情境。在電子商務的推薦情境中，我們所關注的是使用者有興趣購買的商品，使用者也只會看前一兩頁的推薦商品去下單。所以如何讓使用者「看到」喜歡的商品的機會增加，變成是在電子商務推薦系統中重要的課題。

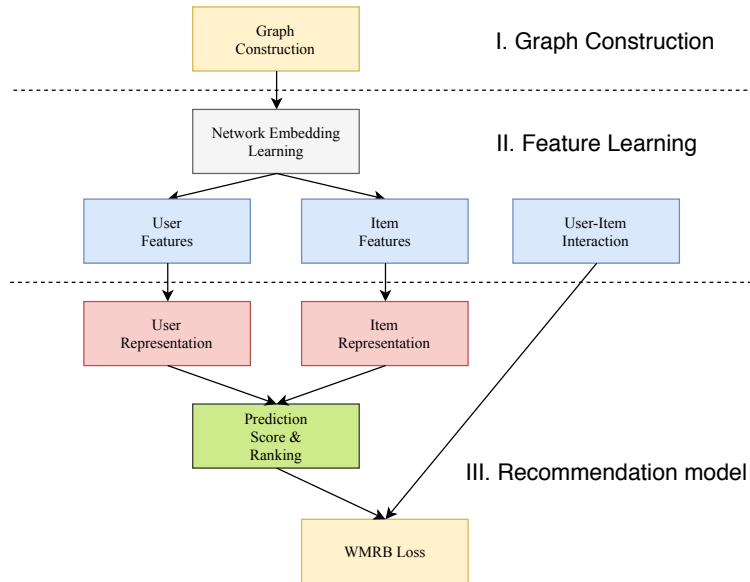


圖 1.1: 此論文的推薦系統架構圖。主要分成三大階段，第一階段是將資料轉換成異質性網路。第二階段則是學習異質性網路表示法。第三階段再透過第二階段學出的使用者特徵值及商品特徵值，透過 Learning to Rank 的方式訓練矩陣分解模型，進而達到推薦商品的目的。

此論文嘗試針對電子商務推薦系統制定一個泛用的建圖策略，將使用者、商

品、文字結合成異質性網路中。再透過網路表示法學習，將節點投影到同一個向量空間。只不過學習出來的表示式，並沒有辦法被直接應用到推薦系統上。所以我們更嘗試著將學習到的表示法結合矩陣分解（Matrix Factorization）以及排序學習（Learning to Rank）作為我們推薦的模型。透過監督式的學習，我們可以將使用者評分、點擊、購買的紀錄結合網路表示式，以達到混合式的 top-N 推薦（如圖1.1）。在電子商務平台 Amazon 這個真實世界的資料集中，此方法皆能使推薦的效果有所提升。另外，此方法在覆蓋率（Coverage）的表現也明顯優於其他方法，帶動整體商品的流通。我們認為這樣有助於電商平台的營業額提升。此研究主要的貢獻整理如下：

- 我們提出了一個針對商品推薦的建圖方法，再透過網路表示法學習訓練使用者及商品的表示式，作為推薦系統的初始特徵值。
- 我們提出了一個結合異質性網路表示法的推薦方法，融合了個人化推薦及基於內容推薦的優點，以達到改善現行推薦系統的效果。

之後的論文章節安排如下：第二章介紹相關的文獻研究。第三章提出此論文的研究方法及模型。第四章敘述我們所使用的資料集、實驗設定、以及實驗結果。最後第五章是我們對此研究的結論。

## 第二章

### 相關文獻探討

#### 2.1 推薦系統

許多方法結合使用者對於商品的評分，來增進推薦系統的準確性。基於矩陣分解（Matrix Factorization）的協同過濾系統（Collaborative Filtering）[9] 是目前業界最普遍使用的技術。目標是透過興趣相似的群體的喜好，來推薦使用者感興趣的資訊或商品。而在電子商務中，可以根據使用者的行為來推薦他可能喜歡的商品。像是利用瀏覽、點擊、購買、評論的行為，來量化每個使用者對於不同資訊的喜好強弱。矩陣分解是將一個矩陣分解成兩個或多個矩陣的乘積。假設我們有  $m$  個使用者、 $n$  個商品，在推薦系統中，我們常會把使用者對商品的評分數據紀錄成下列形式：

	$i_1$	$i_2$	$i_3$	$i_4$
$u_1$	2		4	
$u_2$	5	3		
$u_3$	1	5		5
$u_4$		5		4

表 2.1: 使用者 — 商品的評分數據範例

我們利用上述的資料，建構使用者 — 商品矩陣  $X_{m \times n}$ 。假設分解成兩個矩陣  $P_{m \times k}$  和  $Q_{k \times n}$ ，我們要使矩陣  $P_{m \times k}$  和  $Q_{k \times n}$  的乘積能夠接近原本的矩陣  $X_{m \times n}$ 。

$$X_{m \times n} \approx P_{m \times k} \times Q_{k \times n} = \hat{X}_{m \times n} \quad (2.1)$$

矩陣分解能分解出使用者和商品的潛在因素（Latent Factor）。並透過學習出來的潛在因素，去預測使用者對於沒有評分的商品的喜好程度。雖然說矩陣分解

的技術在很多應用獲得好的效果，但是當使用者只對極少的物品評分時，往往會遇到資料過於稀疏的挑戰。而在真實世界中，這是一個經常會發生的問題。

近年來，網際網路的快速發展，平台上有許多使用者、物品的額外資訊（Meta Information），研究者發現這些額外資訊對於個人化推薦有極大的幫助。因此開始有研究開始結合這些資訊，來改進推薦系統的準確性。有一部份的研究利用使用者對於商品的評論，來掌握使用者對於商品的偏好及習慣。像是 Ling 等人與 McAuley 等人利用主題模型（Topic Modeling）的技術，找出評論文字與分数的關係並加以利用 [10, 11]。爲了利用文字語義資訊，Zhang 等人則是結合 word2vec 和傳統矩陣分解的作法，以提升個人化推薦的效果 [12]。

除了文字資訊之外，許多研究加入圖像資料來學習使用者的喜好。像是 McAuley 等人使用了圖像特徵值來加強協同過濾的準確性 [13]。He 等人則是針對 top-N 的推薦，建立了基於圖像的 Bayesian Personalized Ranking（BPR）模型 [14]，另外他們也針對了衣物穿搭的圖像特徵以及歷史紀錄學習使用者品味，以加強在商品風格的推薦 [15]。

近幾年的推薦問題研究，也有不少使用深度學習框架。像是 He 等人提出通用的神經網路模型，來代替內積對使用者－商品的互動建模 [16]。Zheng 等人等人利用卷積神經網路（Convolutional Neural Network），從使用者、商品的評論中學習使用者、商品的表示法，再結合矩陣分解模型完成評分預測的任務 [17]。Zhang 等人則是將文字、圖像、分数的資訊串接，經由多層神經網路學習使用者及物品的表示式以便後面的 top-N 的 pair-wise 學習排序推薦 [18]。Shi 等人則是使用了異質性資訊網路表示法當作特徵值，來加強評分預測任務的準確性 [19]。雖然這些方法利用額外資訊加入推薦系統，但許多方法往往是針對評分預測去做優化。而在電子商務的推薦情境中，使用者是否購買商品與使用者評分的高低並沒有直接的關係。而在有商品資訊的情況之下有一個通用的推薦策略，能帶給電商平台更多的收益，這也是此論文框架所想要達到的目的。

## 2.2 網路表示法學習

在自然語言處理、文字探勘中，詞向量（Word Embedding）已經成爲一種常用的文本特徵值且得到廣泛的應用。在另一方面，網路表示法學習（Network Embedding）希望透過網路結構的鏈結，得到網路節點的向量表示法，作爲後續任務的特徵值。Perozzi 等人學習了 word2vec 的精神，將網路結構透過隨機走動的方法，轉換成類似自然語言句子的節點序列。再通過 skip-gram 訓練得到每個網路節點的向量表示法 [6]。Grover 等人定義了另外一種基於深度優先搜尋（Depth-first search）及廣度優先搜尋（Breadth-first search）的走動策略，以實現更有彈性的框架彈性 [8]。此外，Tang 等人則是提出了明確的目標函式去優化網

路表示法學習問題 [7]。另外，也有不少工作結合節點的資訊訓練網路表示法。像是 Yang 等人結合了 TF-IDF 文字特徵值矩陣，以學到含有更多文字資訊的表示式 [20]。Tu 等人則是結合了節點 label 資訊及支持向量機器的損失函式，使表示法學習帶有具鑑別分類的特徵 [21]。

但是，絕大部分的網路表示法學習方法，往往是針對同質性網路。而在異質性網路中，由於稀疏性及雜訊都比同質性網路高上許多，若直接套用上述等方法訓練模型，往往無法達到很好的效果。因此開始有針對異質性網路表示法學習的研究出現。Chang 等人利用多層嵌入函式的架構，學習異質性網路中資料複雜的交互關係 [22]。Dong 等人則是利用元路徑（Meta-path）的方式去決定每個節點的 context，再使用異質性 skip-gram 模型訓練每個節點的表示式 [23]。此論文將會利用 [23] 的方法訓練異質性網路表示法。在第三章中，我們會再詳細地介紹如何定義及實現。

# 第三章

## 研究方法

在此章節，我們將會詳細地介紹我們的方法。在 3.1 小節我們會先定義我們所要解決的問題。3.2 小節介紹如何學習我們如何建構異質性網路及訓練網路表示法。最後 3.3 小節我們則是講述如何結合網路表示法，實作混合式推薦模型。

### 3.1 問題定義

我們將系統化地闡述我們想要解決的問題。我們表示使用者集合為  $U$ 、表示商品集合為  $I$ 。在訓練資料中，紀錄著使用者  $u$  與商品  $i$  有正向互動（可能是瀏覽、點擊、購買等行為），每個使用者會有一至數對互動關係  $(u, i) \in U \times I$ 。在這邊我們定義  $S$  為我們觀察到的所有使用者－商品集合。而我們的框架想要優化的是下列抽象的損失函式（Loss Function）：

$$\mathcal{L} = \sum_{(u, i_u) \in S} g(u, i_u, \bar{i}_u) \quad (3.1)$$

其中  $g(\cdot)$  代表的是我們定義的排序損失函式（Ranking Loss Function）。 $(u, i_u)$  是正向的使用者－商品互動（使用者  $u$  購買了商品  $i_u$ ）。 $(u, \bar{i}_u)$  為負向的使用者－商品互動（使用者  $u$  沒有購買  $\bar{i}_u$ ）。我們希望訓練模型，使得正向的使用者－商品互動的預測分數大於負向的使用者－商品互動的預測分數。而這邊需要注意的是，在訓練模型的時候，我們只在意排序是否正確，而預測的分數將只會是相對的大小。



## 3.2 異質性資訊網路表示法學習

首先，我們先定義異質性資訊網路。異質性資訊網路（Heterogeneous Information Network）是一種具有多種節點型態的特殊資訊網路。一個異質性網路可以表示成  $G = \{V, E, T\}$ ，節點集合  $V$ 、連結集合  $E$ ，以及對應的型態集合  $T$ 。 $T$  代表了節點的型態集合，並且  $|T| > 2$ 。

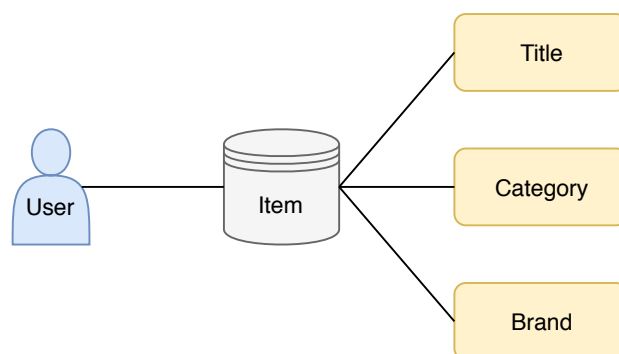


圖 3.1: 電子商務的異質性資訊網路

舉例來說，電子商務的購物行為可以表示成圖 3.1 的網路。使用者（user）、商品（item）、標題（title）、類別（category）、品牌（brand）為節點。而連結代表的意義是購買（user-item），抑或是商品屬性（item-title, item-category, item-brand）。在此小節中，我們的目標是讓每個商品節點  $v$  學習一個好的低維度表示法  $e_v$ 。

### 3.2.1 建圖策略

處理機器學習的問題的過程中，除了模型之外，訓練資料的品質好壞也會直接影響我們預測任務的結果。訓練網路表示法學習的時候，我們的輸入就是一張網路。而現實世界的資料往往不是以網路的方式儲存，我們需要進行透過演算法將原本的資料轉換成網路。在此小節我們提供了一個泛用的策略，將使用者－商品紀錄以及商品標題、商品後設資訊等資料轉換成異質性資訊網路。

首先，我們的輸入是使用者的歷史紀錄，並將每個使用者－商品互動關係建邊。再來是將商品的標題斷詞，並經過一些前處理過濾辭彙。這邊可以依據情境做客製化調整，我們建議將純數字、停用詞去除。前處理之後，再將每個商

品－辭彙關係建邊。另外，如果有其他可以利用的商品屬性，我們也可以利用類似於商品標題的方式建邊。像是品牌、發行商、作者等，我們都認為是可以利用的額外資訊。完整的演算法框架我們列於 Algorithm 1。

---

**Algorithm 1** 建圖演算法

---

**Input:** The users set  $U$ ; The items set  $I$ ; The user-item history pairs  $\mathcal{S}$ ; The set of item titles  $\mathcal{T}$ ; The set of item useful attributes  $M$  (brand, publisher, etc.);

**Output:** The edgelist of heterogeneous information network, denoted by  $\mathbb{E}$ .

```

1:
2: for each positive pair  $(u, i_u) \in \mathcal{S}$  do
3:   append  $(u, i)$  into  $\mathbb{E}$ 
4: end for
5: for each item  $i$  in  $I$  do
6:   words  $\mathcal{W} = \text{tokenize}(\mathcal{T}_i)$ 
7:   preprocessing  $(\mathcal{W})$ 
8:   for each word  $w$  in  $\mathcal{W}$  do
9:     append  $(i, w)$  into  $\mathbb{E}$ 
10:  end for
11:  for each attribute  $m$  in  $M_i$  do
12:    append  $(i, m)$  into  $\mathbb{E}$ 
13:  end for
14: end for
15: return  $\mathbb{E}$ 

```

---

在電子商務的資料中，商品會有像是標題、標籤、品牌等有用的離散特徵。透過我們的建圖策略，對於後續的網路表示法學習來說，除了能學到使用者與商品之間的鏈結關係之外，購買紀錄較少的商品也能透過文字的鄰居共享，拉近與相似商品的距離。我們認為能夠使推薦模型推薦新上架或較為冷門的商品的可能性上升，提高整體推薦系統的品質。

生成  $\mathbb{E}$  之後，我們等同得到異質性網路  $G$ 。再來，我們挑選了四種網路表示法學習方法訓練使用者及商品的表示法。分別為 Deepwalk、LINE、HPE、以及 metapath2vec。學習到的表示式，我們將會當作 3.3 小節中推薦模型的初始特徵值。

### 3.2.2 Deepwalk

Deepwalk [6] 學習了 word2vec 的精神，將網路結構透過隨機走動的方法，轉換成類似自然語言句子的節點序列。我們再通過 skip-gram 訓練得到每個網路節點的向量表示法。我們給定網路  $G$ ，我們將會針對生成的節點序列，優化以下機率分佈：

$$\arg \max_{\theta} \sum_{v \in V} \sum_{c \in N(v)} \log p(c|v; \theta) \quad (3.2)$$

且定義條件機率  $p(c|v; \theta)$ ：

$$p(c|v; \theta) = \frac{y^{O_c \cdot O_v}}{\sum_{u \in V} y^{O_u \cdot O_v}} \quad (3.3)$$

其中  $N(v)$  是節點  $v \in V$  的鄰居節點。 $c$  則是代表節點中  $v$  的 context。 $O_v$  為  $v$  的向量表示法， $O_c$  是 context 的向量表示法。雖然想法不難，但學習出的表示法在分類、連結預測等問題有不錯的效果。也因為實作快速，也在實務的工作上有許多應用。

### 3.2.3 LINE

LINE [7] 則是提出了明確的目標函式，優化網路表示法學習問題。一種為一階相似度（First-order Proximity），就是當兩點直接連接，或連接的權重大的時候，則兩個點愈相似。另一種為二階相似度（Second-order Proximity），當兩點有很多共同鄰居的時候，則他們的相似度很高。我們以圖 3.2 來做解釋：

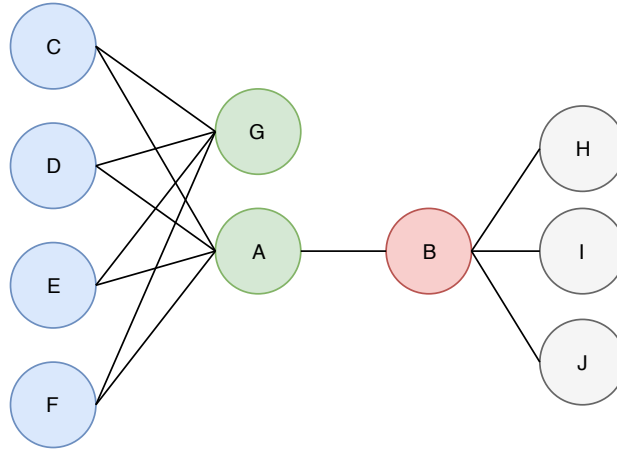


圖 3.2: LINE 的相似度關係

就一階相似度而言，A 點和 B 點是相似的（A 和 B 直接相連）。而就二階相似度來說，A 點則會跟 G 點比較相似（共享了 C、D、E、F 點鄰居）。

在此論文中，我們使用 LINE 的二階相似度。每個節點扮演了兩個角色：節點本身以及其他節點的 context 針對每個有向邊  $(v_i, v_c)$ ，我們定義  $v_i$ ，他所擁有的 context 節點  $v_c$  的條件機率為：

$$p(v_j|v_i) = \frac{\exp(O_{v_c} \cdot O_{v_i})}{\sum_{v_k \in V} \exp(O_{v_k} \cdot O_{v_i})} \quad (3.4)$$

並優化以下目標函式：

$$O_{LINE} = - \sum_{(i,j) \in P} w_{i,j} \log p(v_j|v_i) \quad (3.5)$$

其中  $P$  代表我們抽樣到的邊集合， $O_{v_i}$  為  $v_i$  的向量表示法， $O_{v_c}$  是 context 節點  $v_c$  的 context 向量表示法。 $w_{i,j}$  則是代表邊的權重。

### 3.2.4 HPE

HPE [24] 則是類似於上述兩項工作，但對於網路表示法學習有做些許的變化。不像 Deepwalk 照著均勻分布隨機走動，為了學習使用者的偏好，HPE 透過有權重的隨機走動生成訓練對。拿圖 3.2 當例子，在抽樣 A 點的正向對時，假設我們抽到 B 點當作 A 點的 context，那我們也會同時抽樣 B 點的正向對（像是 A、H、I、J 點）當作 A 點的 context。代表著抽樣的時候除了直接連接的節點之外，也能透過有權重的隨機走動，接收間接連接的節點 context。HPE 根據異質性網路抽樣更新節點表示式  $O_{v_i}$ ，並定義條件機率：

$$P(v_j|O_{v_i}) = \begin{cases} 1 & \text{if } v_j \in \text{Context}(v_i) \\ 0 & \end{cases} \quad (3.6)$$

類似於 LINE，HPE 優化以下目標函式：

$$O_{HPE} = - \sum_{(i,j) \in P} w_{i,j} \log p(v_j|O_{v_i}) + \lambda \sum |O_{v_i}|^2 \quad (3.7)$$

其中  $P$  代表我們抽樣到的邊集合， $w_{i,j}$  則是代表邊的權重。另外，有加入正規化參數，以避免過適（Overfitting）的問題發生。

### 3.2.5 metapath2vec

另外，我們也挑選 metapath2vec [23] 的方法來學習異質性網路表示法。由於每個節點有其節點類別，因此我們可以依據不同的元路徑（Meta-path）去限制隨機走動的序列。給定一個異質性網路，我們定義元路徑  $\rho$  為下式：

$$\rho = A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \dots A_t \xrightarrow{R_t} A_{t+1} \dots \xrightarrow{R_{\ell-1}} A_\ell \quad (3.8)$$

$\ell$  為元路徑的長度， $A = A_1, A_2, \dots, A_\ell$  為此元路徑所走動會經過的型態， $R = R_1, R_2, \dots, R_{\ell-1}$  代表在  $A_1$  到  $A_\ell$  節點型態之間的關係。給定元路徑  $\rho$  及異質性網路  $G$ ，我們根據下式機率分佈產生走動序列：

$$P(v_{k+1}|v_k, \rho) = \begin{cases} \frac{1}{|N(A_{t+1}, v_k)|} & (v_{k+1}, v_k) \in E, \phi(v_{k+1}) = A_{t+1} \\ 0 & (v_{k+1}, v_k) \in E, \phi(v_{k+1}) \neq A_{t+1} \\ 0 & (v_{k+1}, v_k) \notin E \end{cases} \quad (3.9)$$

其中  $v_k$  是在序列中第  $k$  個的節點， $A_t$  為  $v_k$  的節點型態， $N(A_{t+1}, v)$  代表的是  $v$  型態為  $A_{t+1}$  的鄰居個數。元路徑走動會根據事前定義好的路徑  $\rho$  重複產生序列，直到達到我們指定的長度參數，才停止走動。

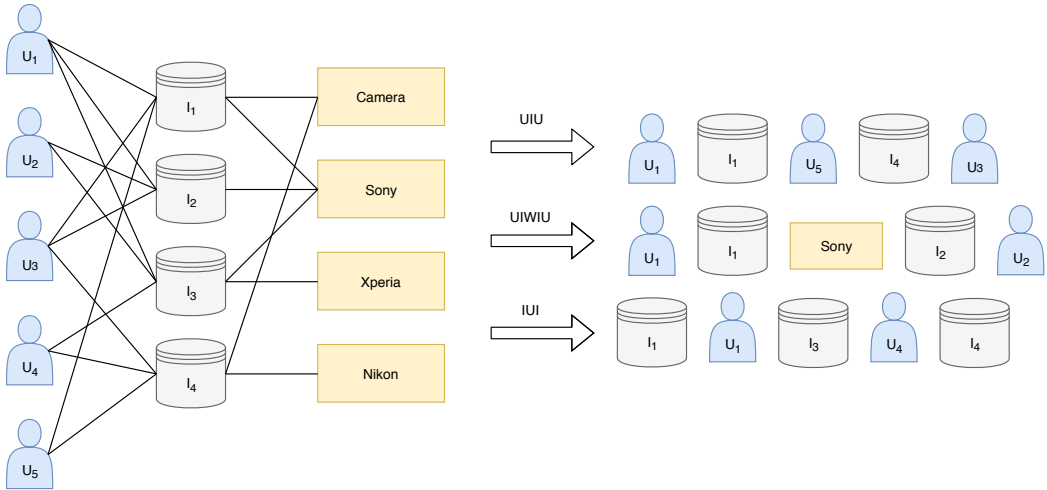


圖 3.3: Meta-path 的序列生成例子

我們拿圖 3.3 當作例子。給定 UIU 的元路徑，從  $U_1$  節點出發，我們可以透過節點走動產生路徑：(1)  $U_1 \rightarrow I_1 \rightarrow U_5 \rightarrow I_4 \rightarrow U_3 \dots$ ，(2)  $U_1 \rightarrow I_3 \rightarrow U_4 \rightarrow I_4 \rightarrow U_5$ 。而當我們給定 UIWIU，同樣從  $U_1$  節點出發，則可以產生路徑：(3)  $U_1 \rightarrow I_1 \rightarrow W_{Sony} \rightarrow I_2 \rightarrow U_2 \dots$ ，(4)  $U_1 \rightarrow I_3 \rightarrow W_{Sony} \rightarrow I_2 \rightarrow U_3 \dots$ 。

從上面的例子可以看出，不同的元路徑所代表的物理意義不同，例如像是 UIU 所代表的意義是連結購買相同商品的使用者，而 UIWIU 則是連結購買有相同字詞商品的使用者。

產生序列文本之後，我們透過異質性 skip-gram 模型，學習有效的網路表示法  $O \in \mathbb{R}^{|V| \times d}$ ,  $d \ll |V|$ 。給定異質性網路  $G = (V, E, T)$ ， $|T_V| > 1$ ，以及節點  $v$ ，去優化以下機率分佈：

$$\arg \max_{\theta} \sum_{v \in V} \sum_{t \in T_V} \sum_{c_t \in N_t(v)} \log p(c_t|v; \theta) \quad (3.10)$$

且定義條件機率  $p(c_t|v; \theta)$ ：

$$p(c_t|v; \theta) = \frac{y^{O_{c_t} \cdot O_v}}{\sum_{u_t \in V_t} y^{O_{u_t} \cdot O_v}} \quad (3.11)$$

$N_t(v)$  是節點  $v$  中型態為  $t$  的鄰居節點。 $c_t$  則是代表節點中  $v$  型態為  $t$  的 context。 $O_v$  為  $O$  的第  $v$  列向量，也代表著節點  $v$  的表示式  $f_v$ 。一般的網路表示法學習，如 DeepWalk、node2vec，所優化的多項機率分佈（Multinomial Probability Distribution）是全部的節點去建構機率分佈。而在異質性 skip-gram 中，每個型態  $t$  的多項機率分佈是由所有型態為  $t$  的節點所建構。這樣可以確保在負向取樣（negative sampling）的時候，我們所抽取的節點為同樣型態的，並專注在相同型態節點的資訊。

### 3.3 推薦模型

透過 3.2 所學習出的表示法，我們有使用者特徵值及物品特徵值，其集合各別記為  $F_U$ 、 $F_I$ 。每個使用者  $u$  都有其特徵值  $f_u \in F_U$ ，每個物品  $i$  也有其特徵值  $f_i \in F_I$ 。集合  $S$  為我們觀察到的使用者－商品正向互動矩陣。 $W_1 \in \mathbb{R}^{d \times k}$  及  $W_2 \in \mathbb{R}^{d \times k}$  各別為連結使用者特徵值、商品特徵值，所學習到的權重。其中  $d$  為我們設定的網路表示法維度， $k$  是我們在矩陣分解中可以設定的隱含表示法維度參數。我們利用線性組合的方式，得到使用者的隱含表示式（Latent Representation）：

$$r_u = f_u W_1 \quad (3.12)$$

及商品的隱含表示式為：

$$r_i = f_i W_2 \quad (3.13)$$

在模型中，使用者  $u$  對於商品  $i$  的預測值  $\hat{p}_{i_u}$ ，我們定義為使用者表示法及商品表示法的內積值。 $b_u$  和  $b_i$  是在預測時我們所加上的正規化參數（Regularization Bias）：

$$\hat{p}_{i_u} = r_u \cdot r_i + b_u + b_i \quad (3.14)$$

最後我們定義損失函式  $g(\cdot)$ 。 $i_u$  代表與使用者有互動關係的商品。對每個使用者我們將會隨機抽樣數個商品  $i' \in I$ 。在這邊我們使用 WMRB [25] 的作法，並使用梯度下降法（Gradient Descent）做  $g(\cdot)$  最小化優化：

$$g(\hat{p}, u, i) = \log\left(\frac{|I|}{|Z|} \sum_{i' \in Z} |1 - (\hat{p}_{i_u} - \hat{p}_{i'_u})| + 1.0\right) \quad (3.15)$$

由於在實務上，我們沒辦法在訓練的過程中一一計算每個負向對的分數，因此我們會使用負向採樣（Negative Sampling）策略來優化我們的訓練過程。 $Z$  是我們在整個商品集合  $I$  中抽樣出來的商品集合。 $\hat{p}_{i_u}$  代表的是正向對的分數， $\hat{p}_{i'_u}$  則是代表了我們抽樣對的分數。(3.15) 式計算了整體商品集合的 score violation。這代表了我們希望  $(\hat{p}_{i_u} - \hat{p}_{i'_u})$  的值越接近 1 越好，如此才能使 (3.15) 式最小化。(3.15) 式可以在許多深度學習框架中，使用 SGD（Stochastic Gradient Descent）的技術實現。最後，我們可以利用學習出的模型，預測使用者對於每個商品的分數。我們再取前  $k$  項分數排序最高的商品，產生個人化的商品推薦清單。

## 第四章

### 實驗結果與討論

在此章節，我們將提供且分析我們框架的表現。在 4.1 小節我們將敘述我們使用的資料集及相關統計數據。4.2 小節則是給定實驗所使用的參數及我們所比較的方法。4.3 小節介紹我們所使用的評估指標算法。最後，4.4 小節將會比較和討論我們方法及其他方法的表現。

#### 4.1 資料集

我們使用 Amazon 評論資料集<sup>1</sup>來實驗。此資料集收集了 Amazon 網站從 1996 年五月至 2014 年六月約 1.42 億則評論。其中包含了使用者對於商品的評論分數、文字，以及商品的後設資料（像是商品描述、分類、價格、品牌、圖片等有用資訊）。我們挑選了其中六個商品類別，Movies、Home、Electronics、Cell Phones、CDs、以及 Clothing 的資料集實驗。另外，我們選擇的是標準的 5-core 資料，這代表著在資料中，已經過濾掉出現低於 5 次的使用者及物品。表 4.1 為資料集的統計數據。

資料集	Movies	Home	Electronics	Cell Phones	CDs	Clothing
$ U $	123960	66519	192403	27879	75258	39387
$ I $	50052	28237	63001	10429	64443	23033
$ S $	1697533	551682	1689188	194439	1097592	278677
$d(\cdot)$	13.69	8.29	8.78	6.97	14.58	7.06
稀疏度	0.0273%	0.0293%	0.0139%	0.0669%	0.0226%	0.0307%

表 4.1: Amazon 資料集數據統計

其中  $|S|$  代表所有使用者－商品評論數量， $d(\cdot)$  為使用者評論商品數目的平均。稀疏度我們定義為  $\frac{|S|}{|U| \times |I|}$ ，及使用者－商品相鄰矩陣中非 0 元素所佔的比

<sup>1</sup><http://jmcauley.ucsd.edu/data/amazon/>



例。

我們的實驗希望盡可能的模擬真實世界的情況。所以在每個資料集中，我們把每個使用者的評論依時間戳記（Timestamp）排序。並把前 80% 的資料當作訓練資料，後 20% 的資料當作測試資料。目的是想要透過使用者過往的歷史紀錄，去預測使用者可能有興趣的商品。

## 4.2 實驗設定

我們挑選了幾項其他 state-of-the-art 方法與我們的方法<sup>2</sup>比較：

- **Matrix Factorization**：透過分解評分矩陣成使用者因子矩陣及商品因子矩陣，進而預測使用者對於未評分商品的喜好。為目前應用最廣泛的推薦模型 [9]。
- **BPR-MF**：Bayesian Personalized Ranking 是目前在 top-N 推薦上被廣為使用的方法，這邊我們使用 BPR 損失函式訓練 Matrix Factorization [26]。
- **WARP-MF**：Weighted Approximate-Rank Pairwise 最早被用在影像注釋（Image Annotation）的工作 [27]。他建立了一個有效取樣的作法，在大數據量級的隱含回饋（Implicit Feedback）資料集上常被使用。這邊我們使用 WARP 損失函式訓練 Matrix Factorization。
- **BoW-MF**：詞袋模型（Bag-of-Words model）是在自然語言處理領域中被廣泛使用的表達模型。我們將標題辭彙出現的頻率當作訓練模型的特徵，建立一個稀疏的向量代表每個商品。這邊的詞袋向量我們會當作推薦模型中商品的初始特徵值。
- **JRL**：Joint Representation Learning 提出了結合不同層表示法學習的方法，結合評論文字、圖像特徵、評分資訊，訓練使用者及商品的表示法。再利用這些表示法訓練 pair-wise Learning to Rank 模型，以達到推薦的目的 [18]。我們使用作者提供的程式碼實驗<sup>3</sup>。

以下網路表示法學習方法將會結合我們的建圖方式訓練，所學習到的特徵值當作後續推薦模型的初始值。

- **Deepwalk\***：為 Deepwalk 加上我們所提出的建圖策略之作法。Deepwalk 利用了隨機走動生成序列，並利用網路結構相似於文字的特性訓練網路表示法。我們使用 Deepwalk 所學習到的特徵值當作後續推薦模型的初始值 [6]。

---

<sup>2</sup>此論程式碼可於 github 上取得 [https://github.com/LonSilent/thesis\\_codes](https://github.com/LonSilent/thesis_codes)

<sup>3</sup><https://github.com/evison/JRL>

- **LINE\***：為 LINE 加上我們所提出的建圖策略之作法。我們使用二階相似度的 LINE 訓練網路表示法，節點之間共享的鄰居節點越多則越相似。同時也作為特徵值當作後續推薦模型的初始值 [7]。
- **HPE\***：為 HPE 加上我們所提出的建圖策略之作法。Heterogeneous Preference Embedding 可以將使用者的異質性的資訊結合，讓使用者在進行檢索時能產生對應的推薦清單 [24]。
- **metapath2vec\***：為 metapath2vec 加上我們所提出的建圖策略之作法。metapath2vec 提供了一個針對異質性網路的隨機走動策略，並透過異質性 skip-gram 學習表示法。

參考圖 3.1，實驗時我們設定三種節點型態：使用者（U）、商品（I）、字詞（W）。字詞節點可能為商品標題所含辭彙及商品品牌（如果有的話）。再根據評論訓練資料及商品後設資料建立網路。建立完網路之後，我們使用 proNet [28] 套件學習 Deepwalk、LINE、及 HPE 的網路表示法。訓練 Deepwalk 時，我們給定窗口大小 window size 為 5，並過濾掉 min count 小於 5 的節點，每個節點作為起點的次數 walk times 為 10，隨機走動的長度 walk length 為 50。HPE 及 LINE 我們設定 sample times 為 2 億次。

而在 metapath2vec 的實驗，我們設定元路徑  $\rho$  為 {UIU, UIWIU, IUI} 生成訓練文本。metapath2vec 也相同於 Deepwalk，每個節點作為起點的次數 walk times 為 10，隨機走動的長度 walk length 為 50。得到文本之後，我們再使用 metapath2vec 作者提供的程式碼訓練網路表示法<sup>4</sup>。其他參數我們則是使用套件預設值。最後，以上四種方法所訓練的 embedding 大小我們都設定為 100。

再來是給定推薦模型的參數。我們在 Electronics 的資料集設定使用者 batch 大小為 50000，以免記憶體用盡（Out of Memory）的問題。其餘 5 個資料集則是設定 batch 大小為原本使用者的大小。初始的 learning rate 給定 0.1，並在訓練的過程中動態減少。對於每個使用者，負向取樣 negative sampling 的次數我們設定為 20。在 MF、BPR-MF、WARP-MF 及我們的方法中，我們設定 latent factor 的長度  $K$  為 150，模型訓練更新的次數 train epoch 為 150。

BPR-MF 及 WARP-MF 我們使用 LightFM [29] 實作<sup>5</sup>。JRL 則是參考原作者參數設定，並只使用 text、rating 兩層 view 學習特徵。另外，我們也使用 proNet [28] 套件學習 JRL 的初始 rating 特徵值。

<sup>4</sup><https://ericdongyx.github.io/metapath2vec/m2v.html>

<sup>5</sup><https://github.com/lyst/lightfm>

## 4.3 評估標準

爲了衡量我們方法與其他方法在真實電子商務網站的表現，我們使用下列兩種評估標準來評估 top-N 推薦的表現。

- 精確率（Precision）：成功推薦的商品在模型推薦清單的百分率。這邊我們使用 Macro-average Precision。

$$Precision = \frac{\sum_{u \in U} \frac{|I_u^+|}{N}}{|U|} \quad (4.1)$$

- 召回率（Recall）：測試資料中使用者購買的商品被成功推薦的百分率。這邊我們也使用 Macro-average Recall。

$$Recall = \frac{\sum_{u \in U} \frac{|I_u^+|}{|I_u|}}{|U|} \quad (4.2)$$

- 覆蓋率（Coverage）：爲可推薦商品佔所有商品的比例。

$$Coverage = \frac{|I'|}{|I|} \quad (4.3)$$

其中  $U$  爲使用者集合， $I$  爲總商品集合。 $I_u$  爲使用者  $u$  過去購買的商品集合， $I_u^+$  爲對於使用者  $u$  的成功推薦商品集合。 $I'$  是代表著所有使用者的推薦清單中涵蓋的商品集合，也就是如果推薦系統沒有推薦過某商品，則該商品就不包含在  $I'$  中。

在測試資料中，我們對每個使用者生成  $N = 10$  的 top-N 商品推薦清單。再利用上述兩個評估標準評估各個方法的好壞。

## 4.4 實驗結果

### 4.4.1 準確率及召回率表現

表 4.2 及表 4.3 表列了各個方法的實驗結果，包含了 Precision@10 以及 Recall@10。首先 MF 的表現在許多分類上都表現不好。是因爲 MF 原本的目標是讓均方根誤差（Root Mean Square Error, RMSE）愈小愈好。而在電商的推薦上，更合理的場景是推薦商品的排序問題。所以在大部分的資料上，透過 BPR 或 WARP 的損失函式訓練 MF 會讓整體的準確率提升不少。再來，由

資料集	Movies		Home		Electronics	
評估	Precision	Recall	Precision	Recall	Precision	Recall
MF	0.165%	0.818%	0.077%	0.402%	0.207%	0.963%
BPR-MF	0.798%	3.842%	0.248%	1.343%	0.229%	1.301%
WARP-MF	0.749%	3.571%	0.199%	1.026%	0.251%	1.307%
BoW	0.135%	0.583%	0.155%	0.844%	0.095%	0.517%
JRL	0.964%	5.343%	0.296%	2.222%	0.407%	<b>3.145%</b>
Deepwalk*	0.996%	4.699%	0.318%	1.730%	0.246%	1.381%
LINE*	0.341%	1.381%	0.171%	0.859%	0.148%	0.685%
HPE*	<b>1.147%</b>	<b>5.405%</b>	<b>0.418%</b>	<b>2.313%</b>	<b>0.433%</b>	2.392%
metapath2vec*	0.941%	4.330%	0.375%	2.044%	0.373%	2.018%

表 4.2: top-10 推薦結果 (Movies, Home, Electronics)

資料集	Cell Phones		Clothing		CDs	
評估	Precision	Recall	Precision	Recall	Precision	Recall
MF	0.430%	2.606%	0.093%	0.559%	0.144%	0.577%
BPR-MF	0.663%	4.029%	0.361%	2.145%	1.036%	4.469%
WARP-MF	0.744%	4.512%	0.244%	1.417%	0.811%	3.151%
BoW	0.762%	4.575%	0.477%	2.905%	0.118%	0.429%
JRL	0.825%	<b>6.921%</b>	0.387%	<b>3.585%</b>	0.774%	3.701%
Deepwalk*	0.797%	4.853%	0.414%	2.483%	1.187%	5.113%
LINE*	0.682%	4.149%	0.186%	1.072%	0.244%	0.786%
HPE*	<b>1.039%</b>	6.494%	<b>0.509%</b>	3.099%	<b>1.316%</b>	<b>5.645%</b>
metapath2vec*	0.968%	5.986%	0.412%	2.482%	1.138%	4.780%

表 4.3: top-10 推薦結果 (Cell Phones, Clothing, CDs)

於 JRL 及我們的方法加入了額外資訊訓練，所以表現可以再更進一步提升。然後我們可以發現在大部分的分類中，我們的方法在 Precision 上的表現都明顯優於其他方法。在推薦系統上，這是我們樂見的結果。比起召回率，精確率高的模型更能符合真實世界使用者的期待。

另外，加入了文字資訊訓練網路表示法之後，跟原本單純用使用者－商品互動所建立的模型比起來，所學習到的特徵值也提升了後續推薦系統的準確性。這裡可以解釋不僅商品的後設資料，商品的評論文字也有助於提升推薦模型的表現。這邊值得注意的是 LINE 的表現並不好，我們認為這是因為我們的網路在建置的時候，並不是以二分圖（Bipartite graph）來設計，而我們所建的圖混雜著使用者、商品、文字，使得在學習二階相似度時雜訊過多，而導致 LINE 的學習效果不好。反而是基於隨機走動的網路表示法學習方法，皆有不錯的表現。

#### 4.4.2 覆蓋率表現

	Movies	Home	Electronics	Cell Phones	Clothing	CDs
MF	0.642%	12.430%	0.069%	35.608%	27.512%	29.037%
BPR-MF	60.062%	74.565%	63.059%	82.583%	81.715%	54.684%
WARP-MF	59.980%	61.557%	50.181%	84.016%	68.097%	59.620%
BoW	80.958%	35.156%	73.228%	86.117%	85.835%	46.836%
JRL	8.669%	1.879%	9.971%	14.445%	91.773%	12.023%
Deepwalk*	<b>97.896%</b>	76.692%	<b>93.571%</b>	<b>96.302%</b>	96.811%	70.802%
LINE*	0.233%	0.489%	0.171%	87.550%	18.359%	1.020%
HPE*	71.151%	<b>90.736%</b>	84.541%	95.934%	<b>96.985%</b>	67.591%
metapath2vec*	62.231%	70.339%	40.671%	91.364%	89.470%	<b>91.930%</b>

表 4.4: 覆蓋率評估

在表 4.4 中我們列出了每個模型在覆蓋率的表現。覆蓋率在電商行業中格外重要。不同於線下商店，電商網站擁有大量的商品，而其中活躍的商品可能只佔其中的 10%，而其餘的 90% 的商品流通性都不高。因此帶動整體商品的銷售能給電商更多的營收成長。

這邊可以發現使用此論文建圖策略的推薦模型，在每一類的覆蓋率表現皆有很好的表現。在四個網路表示法學習方法裡，皆有方法能夠達到超過 90% 的覆蓋率。與像是 JRL 的方法比較起來，我們的方法在學習商品與物品的表示法時，會在相同空間學習到使用者與商品之間的交互關係，以及商品與文字的連接。我們認為這樣是能夠表現更好的原因。

資料集	Movies		Home		Electronics	
評估	Precision	Recall	Precision	Recall	Precision	Recall
Deepwalk(rating)	0.911%	4.417%	0.238%	1.303%	0.225%	1.311%
Deepwalk(rating+words)	0.996%	4.699%	0.318%	1.730%	0.246%	1.381%
LINE(rating)	0.639%	2.916%	0.142%	0.720%	0.248%	1.340%
LINE(rating+words)	0.341%	1.381%	0.171%	0.859%	0.148%	0.685%
HPE(rating)	<b>1.209%</b>	<b>5.719%</b>	0.374%	2.010%	0.432%	2.384%
HPE(rating+words)	1.147%	5.405%	<b>0.418%</b>	<b>2.313%</b>	<b>0.433%</b>	<b>2.392%</b>
metapath2vec(rating)	0.885%	4.071%	0.257%	1.376%	0.357%	1.928%
metapath2vec(rating+words)	0.941%	4.330%	0.375%	2.044%	0.373%	2.018%

表 4.5: 有無文字建圖方式的推薦結果比較 (Movies, Home, Electronics)

資料集	Cell Phones		Clothing		CDs	
評估	Precision	Recall	Precision	Recall	Precision	Recall
Deepwalk(rating)	0.681%	4.287%	0.290%	1.739%	1.062%	4.547%
Deepwalk(rating+words)	0.797%	4.853%	0.414%	2.483%	1.187%	5.113%
LINE(rating)	0.485%	2.989%	0.174%	0.996%	0.840%	3.463%
LINE(rating+words)	0.682%	4.149%	0.186%	1.072%	0.244%	0.786%
HPE(rating)	0.880%	5.482%	0.379%	2.279%	1.306%	5.606%
HPE(rating+words)	<b>1.039%</b>	<b>6.494%</b>	<b>0.509%</b>	<b>3.099%</b>	<b>1.316%</b>	<b>5.645%</b>
metapath2vec(rating)	0.666%	4.107%	0.261%	1.568%	0.943%	3.967%
metapath2vec(rating+words)	0.968%	5.986%	0.412%	2.482%	1.138%	4.780%

表 4.6: 有無文字建圖方式的推薦結果比較 (Cell Phones, Clothing, CDs)

### 4.4.3 建圖策略比較

對網路表示法學習來說，不同的網路就代表不同的訓練資料，將會大大影響我們的預測任務結果。透過第三章的 Algorithm 1，我們將文字加入異質性網路。在此單元中，我們將比較有無文字的網路對於推薦系統效能的差異。在無文字的圖中，我們只保留使用者-商品的鏈結，而沒有執行 Algorithm 1 中第 5 行到第 14 行的步驟。網路建置好之後，我們按照一樣的程序：訓練網路表示法，再將表示法當作推薦模型的初始特徵值。

表 4.5 和 4.6 列出了我們實驗的結果。在 Deepwalk 和 metapath2vec 的結果可以看出，加入文字都使推薦的效果有所提升。而在 LINE 中，在比較大的資料集中，加入文字反而或多或少減低推薦的成果。這如同我們之前所提的，加入過多異質性的節點當作 context，由於沒有分享共同的鄰居，反而混亂表示法的學習過程。

這邊值得注意的是，HPE 在 Movies、Electronics、CDs 中，有無文字並沒有使效果明顯的提升，在 Movies 類中更是比沒有加文字的效果還要好。我們認為是因為資料的特性所造成。像是 CDs、Movies 等資料，商品標題的文字沒有那麼重要，反而像是導演、歌手、曲風、發行商等後設資料影響較大，而在 Amazon 資料集中沒有提供這些資訊。可能透過相同的方式連接這些屬性，才會使額外資訊的效果彰顯出來。這也告訴我們仍須根據不同的領域去客製化不同的推薦模型，以達到最好的效果。

### 4.4.4 案例分析-網路表示法學習

除了好的推薦系統以外，我們也有另外一個主要目標，就是希望透過網路表示法學習，解決電子商務中嚴重的稀疏性問題。所學到的表示法除了當作推薦系統的初始值，也能夠在許多實務工作中發揮作用。像是訓練出的使用者表示法可以用在使用者分析（User Profiling）的情境中。就商品的角度而言，我們希望的方向有兩個：

- 常被一起購買的商品越相似越好。例如我們希望一個電腦的節點表示法，與充電器、螢幕、滑鼠、鍵盤等商品相似。換句話說，我們希望能在商品的表示式學到協同過濾的特徵。
- 擁有類似標題辭彙的商品越相似越好。舉例來說，我們希望 **Sony Xperia XZ2** 與 **Sony Xperia XZ2 Case**、**Sony Xperia XZ2 Compact** 等商品相似。加入文字資訊之後，我們希望弭平像是尺寸、型號等「雜訊」，以達到更好的推薦效果。

方法	相似商品
Deepwalk	HP Chromebook 14 (Ocean Turquoise) HP 15.6" Laptop 4GB 320GB   2000-2d19WM 12-Cell Laptop Battery for HP PAVILION DV7 New Laptop Battery for HP Pavilion dm4 HP Pavilion DM1-4310nr 11.6" Laptop (T-Mobile 4G)
LINE	DIGITTRADE LS104-15 Sakura Designer Notebook Sleeve 15,4" Laptop Hanns.G HG281DPB 28" Widescreen LCD Monitor amFilm Premium HD Clear (Invisible) Screen Protectors for Apple iPad Samsung Galaxy Desktop Dock (EDD-D1B1BEGXAR) Furman M-8X2 Merit Series 8 Outlet Power Conditioner and Surge Protector
HPE	HP Chromebook 14 (Ocean Turquoise) HP 15.6" Laptop 4GB 320GB   2000-2d19WM BESTEK hp laptop charger universal adapter Acer Aspire V5-122P-0637 11.6-Inch Touchscreen Laptop (Chill Silver) Toshiba CB35-A3120 13.3-Inch Chromebook
metapath2vec	HP Chromebook 14 (Ocean Turquoise) ASUS Wireless Router (RT-N10P) Anker Golden AC Adapter + Power Supply Cord for Laptop HP Pavilion HP 15.6" Laptop 4GB 320GB   2000-2d19WM HP Chromebook 11 (White/Blue)

表 4.7: 網路表示法學習的案例分析。在不同的網路表示法學習方法下，我們對於 **HP 14-q070nr 14-Inch Chromebook (free T-Mobile 4G)** 這項商品的表示法，與其餘商品表示法進行餘弦相似度檢索。此表所列出的是每個方法中最高前五相似的商品。



另外，也可以將商品表示法直接利用在 Next-items 推薦中，利用使用者的搜尋問句檢索對應的物品。這樣的情境常常在音樂、影片推薦中常常使用。

我們隨機選擇了 **HP 14-q070nr 14-Inch Chromebook (free T-Mobile 4G)** 這項商品，並於表 4.7 列出相似商品。DeepWalk、HPE、metapath2vec 不約而同的將 **HP Chromebook 14** 擺在第一位。這是符合我們預期想要的結果，成功地將分享相同標題辭彙的商品在向量空間中盡可能的靠近。另外，也可以發現有部分商品符合我們所要的協同過濾特性，像是 **12-Cell Laptop Battery for HP PAVILION DV7** 及 **BESTEK hp laptop charger universal adapter** 都某種程度抓到我們想要的結果。

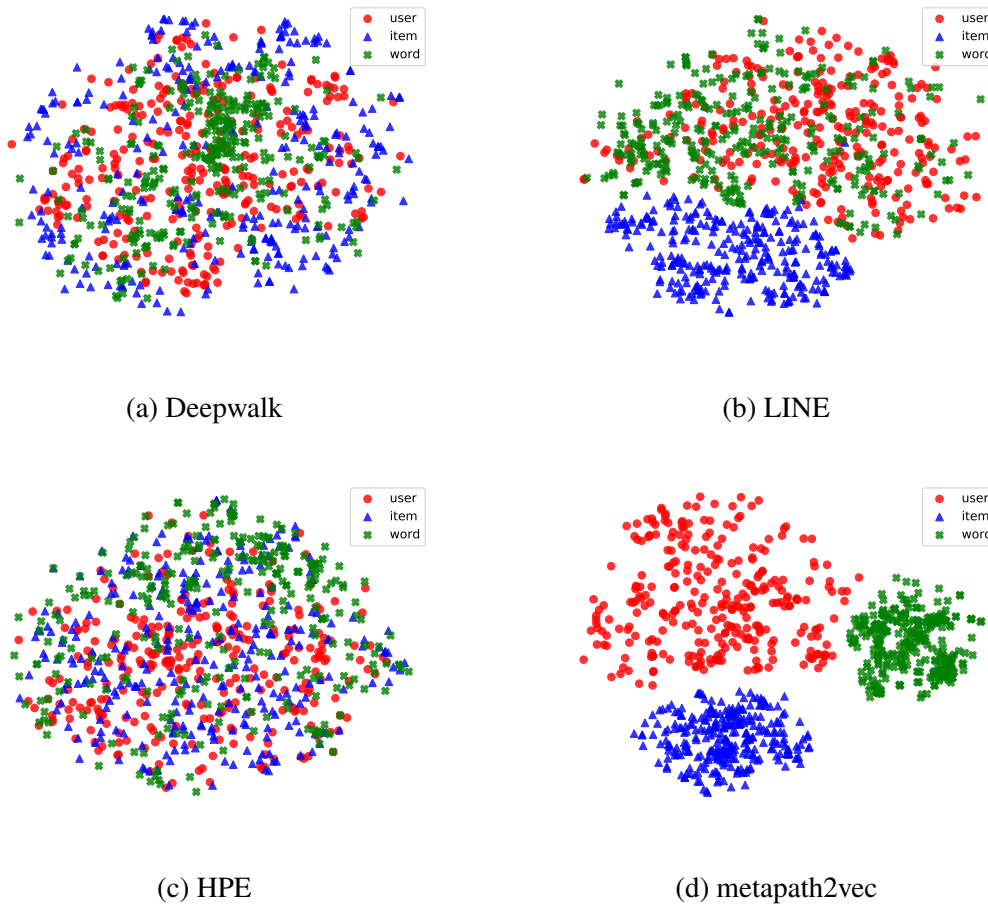


圖 4.1: 網路表示法的視覺化呈現

我們也透過 t-SNE [30] 將網路表示法視覺化呈現，來比較不同網路表示法學習方法的差異。在圖 4.1 中，我們可以看出 Deepwalk 及 HPE 的表示法分佈是混合在一起的，這是因為每個節點型態都會分享到各種型態的 context。而 LINE 則是商品一群，使用者及文字一群。會有這樣的現象則是因使用者及文字都是以商品當作 context，商品則會有使用者及文字的 context。metapath2vec 則是每個型態節點各一群。也證明了透過異質性 skip-gram，可以在享有多種型態 context 節點的情況下，把同型態的節點聚集在一起，並把不同型態的節點分開。

#### 4.4.5 案例分析-推薦系統

	使用者歷史紀錄
	Garmin Portable Friction Mount - Frustration Free Packaging Garmin nüvi 255W 4.3-Inch Portable GPS Navigator (Discontinued by Manufacturer) Case Logic TBC-302 FFP Compact Camera Case (Black) Coby 1.44-Inch TFT LCD SNAPP Mini Camcorder/Camera CAM3001 (Black)
方法	推薦清單
Deepwalk	Transcend 8 GB Class 10 SDHC Flash Memory Card (TS8GSDHC10E) Case Logic DCB-302 Compact Camera Case (Gray) Canon PowerShot CMOS Digital Camera with 20x Image Stabilized Zoom 25mm Wide-Angle Lens and 1080p Full-HD Video (Black) Garmin nüvi 1350LMT 4.3-Inch Portable GPS Navigator with Lifetime Map & Traffic Updates (Discontinued by Manufacturer) Garmin nüvi Vehicle Power Cable
LINE	PNY GTX 750 Ti 2048MB Graphics Cards VCGGTX750T2XPB-OC Dell XPS 14.0-Inch Ultrabook Cutler Hammer CHSPT2ULTRA Single Pole UL1449 3RD Edition Type 2 SPD AmazonBasics High-Speed HDMI Cable - 15 Feet ( 4.6 Meters) Supports Ethernet SanDisk Ultra 64GB MicroSDXC Class 10 UHS Memory Card Speed Up To 30MB/s With Adapter, Frustration-Free Packaging
HPE	Garmin AC Charger and International Adapter Set Case Logic GPSP-2 Professional GPS Case for 4.3-Inch Flatscreen GPS Navigators AmazonBasics Hard Carrying Case for 5-Inch GPS - Black eForCity 4.3-Inch EVA Case for Garmin Nuvi 265WT 1300 1350 1370T GPS Garmin nüvi 1350LMT 4.3-Inch Portable GPS Navigator with Lifetime Map & Traffic Updates (Discontinued by Manufacturer)
metapath2vec	Garmin nüvi 50LM 5-Inch Portable GPS Navigator with Lifetime Maps (US) Garmin Portable Friction Dashboard Mount AmazonBasics Hard Carrying Case for 5-Inch GPS - Black Garmin nüvi 265W/265WT 4.3-Inch Widescreen Bluetooth Portable GPS Navigator with Traffic (Discontinued by Manufacturer) Garmin nüvi 260W 4.3-Inch Widescreen Portable GPS Navigator

表 4.8: 推薦系統的案例分析。我們隨機挑選了一位使用者的歷史紀錄，於此表格列出不同模型的推薦清單比較。

最後，我們來分析推薦的結果是否合乎預期。我們同樣隨機挑選一位使用者，並在表 4.8 列出根據我們推薦模型所建立的推薦清單。從這位使用者的紀錄來看，他購買了支架、GPS、相機包、以及相機。從直覺來看，是一位喜歡外出的使用者，推薦系統應該要推薦給他長途旅行相關的商品。在我們的三種方法中，

皆有一些不錯的推薦。像是 Deepwalk 所推薦的 **Transcend 8 GB Class 10 SDHC Flash Memory Card (TS8GSDHC10E)**（隨身碟）及 **Garmin nuvi Vehicle Power Cable**（車充）就令人驚艷。而其他方法也推薦出像是充電器、GPS 收納盒等推薦。加入了文字資訊之後，推薦清單中也許多含有像是 Garmin、GPS、case 等字的商品，這邊可見我們的方法成功融合協同過濾及基於內容推薦的優點，達到混合式推薦的目的。

## 第五章

### 結論

此論文中，我們提出了一個通用的電子商務推薦系統策略。首先，我們結合了使用者的歷史紀錄及商品額外資訊，訓練異質性網路表示法學習。透過我們的建圖策略以及各種的網路表示法學習方法，我們可以針對不同的情境生成適合的使用者、商品特徵值。由於學到的網路表示法無法直接應用到推薦系統上，我們更利用了基於 Learning to Rank 的矩陣分解方法，來完成 top-N 推薦的任務。與目前現有的推薦系統框架比較，我們的方法在準確率與覆蓋率的取捨上取得一個好的平衡點。而在實務面中，我們的方法較為直覺，且兼顧可用性及易用性，可快速應用到現實推薦系統中。

在未來的工作中，我們將會考慮其他的網路表示法學習方法，而能夠更好地將商品資訊結合到推薦系統。另外，如何有效地利用使用者對於商品的評論，也是我們所關注的課題之一。像是萃取文字序列的特徵，亦或是捕捉句子裡不同段落的語意，都是未來研究的方向。再來是結合不同的深度學習框架，尋找更多的可能性。像是圖像、音訊、影片資訊，都富含著有用的資訊。在第一階段的特徵值學習中，這些資訊都可以被善加利用，以完成各種客製化的推薦任務。

## 參考文獻

- [1] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. Grouplens: An open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, CSCW '94*, pages 175–186. ACM, 1994.
- [2] Michael J. Pazzani and Daniel Billsus. The adaptive web. chapter Content-based Recommendation Systems, pages 325–341. Springer-Verlag, 2007.
- [3] Robin Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, November 2002.
- [4] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.
- [5] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, NIPS'13*, pages 3111–3119. Curran Associates Inc., 2013.
- [6] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14*, pages 701–710. ACM, 2014.
- [7] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *WWW*. ACM, 2015.
- [8] Aditya Grover and Jure Leskovec. Node2vec: Scalable feature learning for networks. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16*, pages 855–864. ACM, 2016.
- [9] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8), 2009.

- [10] Guang Ling, Michael R Lyu, and Irwin King. Ratings meet reviews, a combined approach to recommend. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 105–112. ACM, 2014.
- [11] Julian McAuley and Jure Leskovec. Hidden factors and hidden topics: Understanding rating dimensions with review text. In *Proceedings of the 7th ACM Conference on Recommender Systems, RecSys '13*, pages 165–172. ACM, 2013.
- [12] Wei Zhang, Quan Yuan, Jiawei Han, and Jianyong Wang. Collaborative multi-level embedding learning from reviews for rating prediction. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI'16*, pages 2986–2992. AAAI Press, 2016.
- [13] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 43–52. ACM, 2015.
- [14] Ruining He and Julian McAuley. Vbpr: Visual bayesian personalized ranking from implicit feedback. 2016.
- [15] Ruining He and Julian McAuley. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *proceedings of the 25th international conference on world wide web*, pages 507–517. International World Wide Web Conferences Steering Committee, 2016.
- [16] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web, WWW '17*, pages 173–182. International World Wide Web Conferences Steering Committee, 2017.
- [17] Lei Zheng, Vahid Noroozi, and Philip S. Yu. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM '17*, pages 425–434. ACM, 2017.
- [18] Yongfeng Zhang, Qingyao Ai, Xu Chen, and W Bruce Croft. Joint representation learning for top-n recommendation with heterogeneous information sources. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1449–1458. ACM, 2017.
- [19] Chuan Shi, Binbin Hu, Wayne Xin Zhao, and Philip S Yu. Heterogeneous information network embedding for recommendation. *arXiv preprint arXiv:1711.10730*, 2017.

- [20] Cheng Yang, Zhiyuan Liu, Deli Zhao, Maosong Sun, and Edward Y. Chang. Network representation learning with rich text information. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 2111–2117. AAAI Press, 2015.
- [21] Cunchao Tu, Weicheng Zhang, Zhiyuan Liu, and Maosong Sun. Max-margin deep-walk: Discriminative learning of network representation. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, pages 3889–3895. AAAI Press, 2016.
- [22] Shiyu Chang, Wei Han, Jiliang Tang, Guo-Jun Qi, Charu C. Aggarwal, and Thomas S. Huang. Heterogeneous network embedding via deep architectures. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pages 119–128. ACM, 2015.
- [23] Yuxiao Dong, Nitesh V. Chawla, and Ananthram Swami. Metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, pages 135–144. ACM, 2017.
- [24] Chih-Ming Chen, Ming-Feng Tsai, Yu-Ching Lin, and Yi-Hsuan Yang. Query-based music recommendations via preference embedding. In *Proceedings of the 10th ACM Conference on Recommender Systems*, RecSys '16, pages 79–82. ACM, 2016.
- [25] Kuan Liu and Prem Natarajan. Wmrb: Learning to rank in a scalable batch training approach. *arXiv preprint arXiv:1711.04015*, 2017.
- [26] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, pages 452–461. AUAI Press, 2009.
- [27] Jason Weston, Samy Bengio, and Nicolas Usunier. Wsabie: Scaling up to large vocabulary image annotation. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Three*, IJCAI'11, pages 2764–2770. AAAI Press, 2011.
- [28] Chih-Ming Chen, Yi-Hsuan Yang, Yian Chen, and Ming-Feng Tsai. Vertex-context sampling for weighted network embedding. *arXiv preprint arXiv:1711.00227*, 2017.
- [29] Maciej Kula. Metadata embeddings for user and item cold-start recommendations. In Toine Bogers and Marijn Koolen, editors, *Proceedings of the 2nd Workshop on New Trends on Content-Based Recommender Systems co-located with 9th ACM*

*Conference on Recommender Systems (RecSys 2015), Vienna, Austria, September 16-20, 2015.*, volume 1448 of *CEUR Workshop Proceedings*, pages 14–21. CEUR-WS.org, 2015.

- [30] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.