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

英文介系詞片語定位與英文介系詞推薦

Attachment of English Prepositional Phrases and Suggestions of English Prepositions

研 究 生：蔡家琦

指導教授：劉昭麟

中華民國一百一年七月

July 2012

英文介系詞片語定位與英文介系詞推薦

Attachment of English Prepositional Phrases and Suggestions of English Prepositions

研 究 生： 蔡家琦　　Student : Chia-Chi Tsai

指導教授： 劉昭麟　　Advisor : Chao-Lin Liu

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

中華民國一百一年七月

July 2012

英文介系詞片語定位與英文介系詞推薦

摘要

英文介系詞在句子裡所扮演的角色通常是用來使介系詞片語更精確地補述上下文，英文的母語使用者可以很直覺地使用。然而電腦不瞭解語義，因此不容易判斷介系詞修飾對象；非英文母語使用者則不容易直覺地使用正確的介系詞。所以本研究將專注於介系詞片語定位與介系詞推薦的議題。

在本研究將這二個介系詞議題抽象化為一個決策問題，並提出一個一般化的解決方法。這二個問題共通的部分在於動詞片語，一個簡單的動詞片語含有最重要的四個中心詞（headword）：動詞、名詞一、介系詞和名詞二。由這四個中心詞做為出發點，透過 WordNet 做階層式的選擇，在大量的案例中尋找語義上共通的部分，再利用機器學習的方法建構一般化的模型。此外，針對介系詞片語定的問題，我們挑選較具挑戰性介系詞做實驗。

藉由使用真實生活語料，我們的方法處理介系詞片語定位的問題，比同樣考慮四個中心詞的最大熵值法（Max Entropy）好；但與考慮上下文的 Stanford 剖析器差不多。而在介系詞推薦的問題裡，較難有全面比較的對象，但我們的方法精準度可達到 53.14%。

本研究發現，高層次的語義可以使分類器有不錯的分類效果，而透過階層式的選擇語義能使分類效果更佳。這顯示我們確實可以透過語義歸納一套準則，用於這二個介系詞的議題。相信成果在未來會對機器翻譯與文本校對的相關研究有所價值。

Attachment of English Prepositional Phrases and Suggestions of English Prepositions

Abstract

This thesis focuses on problems of attachment of prepositional phrases (PPs) and problems of prepositional suggestions. Determining the correct PP attachment is not easy for computers. Using correct prepositions is not easy for learners of English as a second language.

I transform the problems of PPs attachment and prepositional suggestion into an abstract model, and apply the same computational procedures to solve these two problems. The common model features four headwords, i.e., the verb, the first noun, the preposition, and the second noun in the prepositional phrases. My methods consider the semantic features of the headwords in WordNet to train classification models, and apply the learned models for tackling the attachment and suggestion problems. This exploration of PP attachment problems is special in that only those PPs that are almost equally possible to attach to the verb and the first noun were used in the study.

The proposed models consider only four headwords to achieve satisfactory performances. In experiments for PP attachment, my methods outperformed a Maximum Entropy classifier which also considered four headwords.The performances of my methods and of the Stanford parsers were similar, while the Stanford parsers had access to the complete sentences to judge the attachments. In experiments for prepositional suggestions, my methods found the correct prepositions 53.14% of the time, which is not as good as the best performing system today.

This study reconfirms that semantic information is instrument for both PP attachment and prepositional suggestions. High level semantic information helped to offer good performances, and hierarchical semantic synsets helped to improve the observed results. I believe that the reported results are valuable for future studies of PP attachment and prepositional suggestions, which are key components for machine translation and text proofreading.

致謝

寫到此頁代表結束這本論文的時候到了，內心充滿了無限感謝的話想說，深深地感謝這一路陪我走過來的人。

很幸運劉昭麟老師能成我的指導老師，這一路很感謝老師給了我一次又一次的機會，讓我知道原來我也可以做到，讓我知道原來還有其它的選擇，讓我知道原來世界沒有這麼難，增加了我人生路途更多的選擇，使我不再只是空想羨慕他人。在這短短的二年，老師不僅在學業上幫助我，也在各式各樣大小事情上幫助我，使我成長。會有這一切的一切都要感謝我敬愛的　劉昭麟老師。

一直以都在背後支持我的家人，感謝你們長久以來不斷支持我所做的決定，讓我能夠無所固慮地探索這個世界。謝謝父母總是在我做事投入時，適時地提醒我，謝謝媽媽的「吃飯囉」，謝謝爸爸的「早點休息」。也謝謝妹妹總是幫我這個忘東忘西的姊姊跑腿。因為有你們，所以才有今天。

感謝昀彥這些年一直陪著我，不僅老是聽著我碎碎念，還帶我看到了許多無限的可能，讓我接觸了我從來沒有想過的事，使我思想更加的開闊。也因為有你，如今我的興趣變成了培養新的興趣，謝謝你帶我看到這個世界有趣的一面。

這二年我也要謝謝 MIG 的夥伴們的陪伴，謝謝學長姊怡軒、建良和裕淇不吝嗇的給我許多建議，謝謝同屆的柏廷和瑞平一起互相扶持，謝謝學弟瑋杰和孫暐的協助幫忙。也謝謝更多的 MIG 夥伴，因為你們，使我這二年的生活更加的豐富。

也謝謝口試委員　張嘉惠老師與　高照明老師的指導。

最後，特別感謝在這幾個月裡，被我不斷地纏著幫我校對論文的媽媽和昀彥。

家琦 2012 年 9 月

目錄

圖目錄

表目錄

第 1 章　緒論

英文介系詞的使用對於英文母語的使用者而言是很直覺，即使英文母語的使用者不知道文法結構，仍然可以精確地表達語義。但對於電腦而言卻很難知道語義，因此不容易判斷正確的修飾對象。對於非英文母語的使用者，自然且正確地表達是有困難的。在現今資訊科技盛行爆炸的時代，我們期望透過大量資料以及資訊技術來輔助人類解決問題，並將我們研究應用於電腦自動化的流程。

1.1　研究背景

英文介系詞一般出現在動詞語片裡，一個動詞片語的結構表示成「**動詞 -名詞片語一 -介系詞 -名詞片語二 (V-NP1-P-NP2)**」。其中「**介系詞 -名詞片語二**」的結構稱為介系詞片語。由動詞片語結構所衍生出來的兩個有趣問題，也就是**介系詞片語定位**與**介系詞推薦**。本研究，我們將深入探討並且試圖解決這二個介系詞相關的議題。

介系詞片語定位問題是我們要如何定位介系詞片語修飾對象？如果站在人類的角度來看，很自然地我們可以設想一些情境，進而能夠馬上判斷出合理的介系詞定位。另一方面，如果我們想利用資訊技術來解決這個問題的話，我們會直覺地認為如果電腦也可以演算出一個適當的情境那麼電腦也許可以像人類一樣正確地判斷出介系詞定位。也就是說，在這樣的直覺裡，我們假設電腦有能力去全面且完整地解析、瞭解甚至推導

出語義，但這個假設在這裡顯然和現實情況不符合。因為，若要能夠瞭解語義，那麼電腦必然需要能夠先瞭解該介系詞片語修飾的對象，此時電腦才有可能真正認識到完整的語義。然而，電腦其實需要利用句子語義幫忙做定位介系詞問題，因此我們不可能使電腦先瞭解到完整的語義後，才解決定位介系詞片語問題。這類似雞生蛋、蛋生雞的問題；解決語義問題需要能夠定位介系詞片語，而定位介系詞片語也需要靠語義。

以一個具體的例子做說明，以句 1為例子。首先我們先想像這句話的情境，在我們主觀的認知中比較容易連想的語境是：這群小孩用湯匙吃蛋糕。根據剛才想像的語境那麼“with a spoon”這個介系詞片語修飾的應該是“ate”這個動詞，如句 2底線所標示。但是其實句 1也可以有另外一種語境的可能，如句 3的情況，對於句 3的解讀應該是這群小孩吃的是旁邊有放湯匙的蛋糕，這時“with a spoon”修飾的對象就是“the cake”這個名詞片語。

句 1. The children ate the cake with a spoon.[1]

句 2. The children ate the cake with a spoon.

句 3. The children ate the cake with a spoon.

英文介系詞推薦的問題是如何推薦正確的介系詞使得動詞片語能正確地表達語義。這對於英文母語的使用者是很自然可以判斷的問題，但對於非母語的使用者來說缺少了可以自然使用介系詞的直覺，只能透過介系詞的功能面決定它的用途。有些介系詞在功能面是類似的，例如 in、on、at 在時間的用途上是經常被混淆的。一般而言，“at”是比較強調某個時間點，“on”則是強調特定的日期，“in”是某個時段。非英文母語的使用者只能透過這些大略的準則判斷介系詞的使用，然而有些介系詞在相似的功能面上容易是受到使用者的母語影響，再以句 1為例，有些使用者可能會誤用為句 4，因為在以中文為母語的情況下，by 或許會被理解成為「倚靠」的意思。

句 4. The children ate the cake by a spoon.

[1] 出自 Chris Manning 和 Hinrich Schütze，Foundations of statistical natural language prochessing 書中 8.3 節

因為介系詞的使用是如此的廣泛，但是讓電腦瞭解語義和非英文母語的使用者來說都是有相當的門檻，所以我們對於介系詞的議題感到有興趣。為了可以更精確地使用介系詞，我們將深入探討這二種介系詞的議題。如果能夠解決這些議題，就可以將此應用做為機器翻譯基石和文本校對用途。

1.2 研究方法

我們的研究嘗試找出上下文無關（context-free）的解決方案。這二個問題共通的部分是動詞片語，其結構是「動詞 -名詞片語一 -介系詞 -名詞片語二」的結構，簡化為四個中心詞「*動詞 -名詞一 -介系詞 -名詞二（V-N1-P-N2）*」。中心詞的定義為詞組中最核心被修飾的詞，以句 1為例，將句 1拆成句 5到句 8好幾個詞組，底線的部分是中心詞；句 5是一個動詞片語，動詞片語的中心詞是動詞；句 6與句 8是名詞片語，名詞片語的中心詞是名詞；句 7是介系詞片語，中心詞是介系詞。我們直接探討動詞片語所抽出的四個主要中心詞並以此做為研究的出發點。再利用 WordNet[2]階層式的概念將中心詞提升到較抽象的語義層級，也就是找出上位詞，並利用資訊技術從大量的語料中找尋是否有一套準則能定位介系詞片語和推薦正確的介系詞。

句 5. ate the cake with spoon — *動詞片語*

句 6. the cake — *名詞片語一*

句 7. with a spoon — *介系詞片語*

句 8. a spoon — *名詞片語二*

我們以一套一般化的方法同時應用於介系詞片語定位與介系詞推薦問題上，研究方法可參考圖 1.1。第一部分語料處理，這部分包了各種語料的前處理：斷句、剖析、

[2]http://wordnet.princeton.edu/

中心詞抽取、雜訊過濾和挑選具挑戰性介系詞。第二部分是特徵處理，這部分是特徵數值化。第三部分是應用 WordNet 階層式的概念挑選特徵。第四部分是模型的建構，我們共設計了三種不同程度的模型解決問題。最後一部分，則是實際測試和評量模型成效。

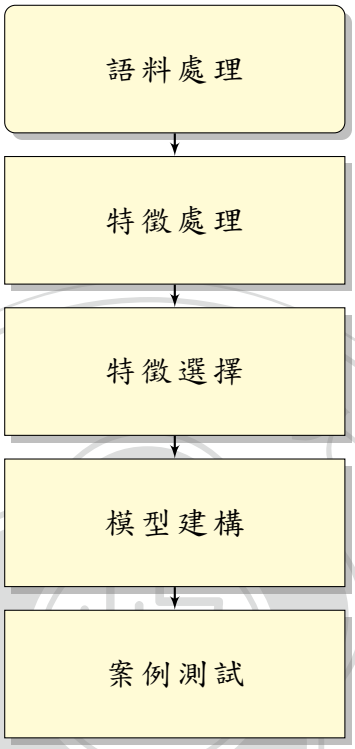

圖 1.1: 研究架構流程圖

在本研究，我們將介系詞片語定位問題與介系詞推薦問題分別做了一些假設與簡化。介系詞片語定位問題，在現實生活中，可能有的答案包含：修飾動詞、修飾名詞、二者皆可或其它。為了將這個複雜的問題轉換成較單純的二分類問題，我們簡化為只有修飾動詞與修飾名詞二種可能。介系詞推薦的問題，在只提供動詞、名詞一和名詞二的資訊下，答案可能不只一個介系詞。因此我們將問題簡化成只有一個答案，只處理只有一個答案的案例。另外，只挑選數量較多的介系詞做實驗。

介系詞片語定位的問題依上述的假設是一個二分類的問題，介系詞推薦的問題則是一個多分類問題，所以，顯然地，我們可以看出推薦問題可能比定位問題的難度要高。

額外值得一提的是，針對介系詞片語定位的問題，許多學者大多希望能夠對所有介系詞找出一套一般化的通則。然而我們從 Ratnaparkhi 等人 [20] 所彙整的中心詞語料庫，也就是 RRR 語料庫[3]，統計各個介系詞數量分布的情況，結果如表 1.1所示，其中 NPP 為修飾名詞的介系詞片語，而 VPP 為修飾動詞的介系詞片語。可以發現每一個介系詞的定位情況都不相同，因此在我們的研究會針對各個介系詞歸納適用的準則。

1.3 研究成果

研究成果的部分，依研究的問題可以分成介系詞片語定位與介系詞推薦二部分。

介系詞片語定位的問題，成果可以分成四大類：(一) 與同樣以中心詞做為出發點的研究做比較，比較的研究方法是 Ratnaparkhi 等人 [20] 提出的最大熵值法，若使用我們的方法效果是比較好；(二) 與以上下文之語義為出發的研究比較，比較的對象是 Klein 和 Manning[12] 的 Stanford 剖析器，雙方實驗的成果則是差不多。我們的優點在於所使用的語義資訊是較少的，缺點則是我們假設中心詞是已知的資訊。在考慮上下文的情況，雖需要的資訊是比較多，但是沒有中心詞的問題；(三) 我們直接使用高層抽象語義也可以有不錯的分類效果，但使用階層式選擇尋找抽象語義的方法比直接使用高層抽象語義更好。(四) 為每個介系詞特製化的分類器效果雖然與混合介系詞差不多，但特製化的分類器可以觀察各個介系詞相關連的抽象詞彙，且混合介系詞訓練語料量遠較各別的介系詞大，因此若能提升語料量，相信特製化的分類器可以表現的再更好。

介系詞推薦的問題，目前是較難與現有的方法做比較，因為目前現有的方法所使用得語料庫都不相同。因此，我們只能依靠大量語料所做得成果和語料庫中介系詞的分布情況，參考我們實驗成果的好壞。成果可以分成二大類：(一) 以 RRR 語料庫為主的實驗，我們從語料庫中挑選了 6 個數量差不多的介系詞做實驗，實驗成果顯示我們的方法是比隨意猜測好；(二) 以華爾街日報與紐約時報所組成的大型語料庫為主的實

[3] https://sites.google.com/site/adwaitratnaparkhi/publications/ppa.tar.gz?attredirects=0&d=1

表 1.1: RRR 語料庫，NPP 與 VPP 的數量

	NPP	VPP	Total
about	187	86	273
above	6	15	21
across	7	20	27
after	23	89	112
against	95	110	205
along	3	6	9
alongside	0	2	2
amid	1	14	15
among	42	57	99
amongst	1	0	1
around	11	25	36
as	123	497	620
at	166	594	760
because	8	23	31
before	10	64	74
behind	9	9	18
below	0	5	5
beneath	0	2	2
beside	0	1	1
besides	2	1	3
between	109	28	137
beyond	6	11	17
but	2	0	2
by	151	326	477
de	2	0	2
despite	0	14	14
down	1	2	3
during	8	102	110
except	2	2	4
for	1342	1310	2652
from	360	716	1076
if	1	2	3
in	1999	2061	4060
inside	1	2	3
into	28	341	369

	NPP	VPP	Total
like	30	21	51
near	4	8	12
next	2	23	25
notwithstanding	0	1	1
of	6553	61	6614
off	8	28	36
on	736	826	1562
onto	1	1	2
out	0	3	3
outside	4	3	7
over	62	132	194
past	0	4	4
per	12	3	15
plus	1	0	1
since	10	24	34
than	82	11	93
through	13	127	140
throughout	2	10	12
to	566	1486	2052
toward	26	13	39
towards	1	1	2
under	25	92	117
unlike	0	3	3
until	1	29	30
unto	1	0	1
up	1	3	4
upon	1	9	10
versus	2	0	2
via	1	10	11
whether	1	0	1
while	0	1	1
with	397	739	1136
within	10	48	58
without	6	68	74
Total	13265	10325	23590

驗，因為語料量遠較 RRR 語料庫大，所以我們選擇詞頻最高的 11 個介系詞做實驗。並參考以 RRR 語料庫為主的實驗中，找到最佳模型的方法，並再重複一次方法流程。實驗的成果顯示我們的模型易受到各個類別語料量的影響，模型容易偏好決策數量較多的類別。我們參考比較的對象是 De Felice 和 Pulman[6] 考慮上下文語義的方法，視窗大

小（window size）設為 6。我們的方法與 De Felice 和 Pulman 的方法相比較是有一小段差距，但我們使用得語義資訊較少，這也表示我們實驗成果仍有進步的空間。

第 2 章　文獻回顧

介系詞的相關研究議題，一直是許多學者努力研究的目標，Baldwin 等人 [3] 於 2009 年時，回顧近十年各式各樣介系詞相關的議題，其中包含了本研究有興趣的二個介系詞議題。本章將回顧過往介系詞議題等相關研究，包含了以四個中心詞或以上下文等方法解決介系詞片語定位問題、介系詞歧義問題和介系詞推薦問題。依據 Baldwin 等人的觀點，使用中心詞處理問題是較屬於比較偏向語法（syntex）層面的方法。雖然本研究也是以中心詞分析為主，但亦使用了 WordNet 查詢中心詞的語義，所以本研究不單純只在語法的層次，我們也進入了語義的層次。

近幾年，也出現了許多不少與介系詞相關的工作坊（workshop），如 SemEval 2007 Task 6[1]和 Helping Our Own 2012 Shared Task[2]。或是相關的介系詞專案，如 The Preposition Project（Litkowski 和 Hargraves[14]）。這些都一再顯示了介系詞的重要性。

2.1　介系片語定位

介系詞片語定位一直是自然語言處理的一個大問題。對於人類而言，在有語境的情況下，可以較輕易地判斷介系詞片語的定位對象，但是人類對於介系詞片語定位的判定正確率也不是百分之百。在 Ratnaparkhi 等人 [20] 曾經做過請人判讀介系詞片語定位的實

[1] http://nlp.cs.swarthmore.edu/semeval/index.php
[2] http://clt.mq.edu.au/research/projects/hoo/hoo2012/index.html

驗。若只給定動詞片語中的四種中心詞，那麼人類判讀的準確率可以到達 88.2%。但是如果再加入上下文等語境，那麼人類的判讀可以高達 93.2%。而 Ratnaparkhi 等人的研究方法，電腦判讀僅有 78.0% 的準確率。

從早期開始，有不少學者採用機率統計的方式試圖解決介系詞片語定位問題（如 Hindle 和 Rooth[10]、Liu 等人 [15] 和 Ratnaparkhi 等人）。經常使用得基本特徵資訊包含動詞片語的四個中心詞：動詞、名詞一、介系詞和名詞二。透過四個中心詞，再經由機率統計模型計算介系詞片語可能的定位。

然而對假設已知四個中心詞，Atterer 和 Schütze[2] 指出這個假設不是憑空而來。但本研究依舊假設中心詞是已知條件，將中心詞的取得視為前處理的一部分，我們抽取中心詞的研究則是依靠 Stanford 剖析器[3]，而 Stanford 剖析則是建立在 Collins[4] 的研究之上。

對於介系詞片語定位而言，語料的來源也是讓人頭疼的問題。到現在最常被使用得語料庫都還是 Penn Treebank 3（以下簡稱 PTB3）。但在現今介系詞定位問題幾乎都倚靠機器學習方法解決，語料多寡便成了最直接影響實驗成果的因素之一。依照語料是否有答案可以將學習的方式分成二類：監督式學習（supervised learning）與非監式學習（unsupervised learning）；前者需要有答案，後者不需要。Ratnaparkhi 等人以監督式學習的演算法最大熵值法（Maximum Entropy）訓練模型。監督式學習往往語料較為稀少，非監督式學習語料來源較不受限，因此數量較多。Pantel 和 Lin[18] 使用非監督式學習方法，透過大量的語料做實驗。也有學者指出比起單獨使用監督式或非監督式學習方法，混合二種方法而成的半監督式學習方法更可以有效的預測介系詞片語定位問題，如 Volk[24] 與 Coppola 等人 [5] 等。一般來說，相對於非監督式學習的方法，監督式學習的效果比較好。本研究是以監督式學習解決問題。

從語義歧義的問題角度看，也有不少關於介系詞片語定位的研究是先處理語義歧問題再決解定位問題。這時，有些學者會考慮上下文資訊的特徵，如 Olteanu 和

[3] http://nlp.stanford.edu/nlp/javadoc/javanlp/edu/stanford/nlp/trees/SemanticHeadFinder.html

Moldovan[17]。這樣做得好處是可以看較全面的資訊，但使用得特徵也會比較多。此外，介系詞本身也有語義歧義的問題，例如 O'Hara 和 Wiebe[16]、Tratz 和 Hovy[23] 和 Hovy 等人 [11] 都曾試圖決解此問題。本研究則是不處理上述這些問題。

在不少文獻中，可以看到每個介系詞均有自己的特色，如 Stetina 和 Nagao[21] 試圖為每一個介系詞製作合適的分類器。且大部分的介系詞更是有慣用方式，可參考表 1.1，例如介系詞“of”大多數的時候都是定位名詞。因此 Coppola 等人將語料中有“of”的案例去除。另外，Coppola 等人也去除了名詞一是代名詞的情況，因為代名詞通常不會被定位為修飾名詞。將這些接近幾乎慣用的案例去除後，可以使得實驗結果更可以被信賴。所以，在我們的研究中，我們將挑選具有挑戰性的介系詞，並為每個介系詞特製化分類器。

實驗比較部分，我們的研究除了與同樣都是使用四個中心詞研究比較外，同時也與剖析器做比較。然而 Agirre 等人 [1] 指出這是不公平的比較，因此他們透過改良剖析器的效果證明他們的研究方法確實可以公平的與剖析器比較。但在這裡因為時間、人力有限的情況下，我們無法一一為每個剖析器改善其效果，因此我們仍是直接與現有的剖析器比較研究成果。

2.2 介系詞推薦

推薦問題與定位問題的歷史相比較，是屬於比較年輕的問題。目前許多研究大多視為是文本校對的應用，且視為是「介系詞校正」的問題，較早的相關研究有 De Felice 和 Pulman[6]、Gamon 等人 [8] 和 Tetreault 和 Chodorow[22]。

Han 等人 [9] 使用真實的誤用的語料庫建構校正系統。Leacock 等人 [13] 以網際網路為基礎透過搜尋引擎建議合適的介系詞。在 Helping Our Own 2012 Shared Task 更是有一半的工作著重於介系詞校正的議題上，相關研究如 Wu 等人 [25] 和 Quan 等人 [19]。

綜合大部分的研究結果顯示，使用真實犯錯語料是比較困難的問題。

校正嚴格來說可以分成二階段：第一個階段是偵錯，第二個階段是更正。De Felice 和 Pulman[7]、Gamon 等人和 Helping Our Own 2012 Shared Task 都是二個階段皆著重。De Felice 和 Pulman[6] 是著重於後者。本研究也是著重於後者，廣義來說，我們將推薦視為是一種校正，但因為本研究不包含偵錯，所以我們強調是介系詞推薦。

De Felice 和 Pulman[6] 的研究與本研究的介系詞推薦是較相近，同樣是使用文法正確的語料庫訓練模型，且將實驗限縮在常用的介系詞。然而使用得語料庫不太相同，因此無法直接比較，但我們參考 Gamon 等人藉由語料庫中介系詞的分布，參考比較自己目前的實驗效果。

第 3 章　語料處理

本章將介紹本研究所使用得語料庫以及詞彙資料庫 WordNet。我們所使用的語料庫，不論直接或間接均來自於 Penn Treebank。我們不僅直接從 PTB3 彙整出需要的資訊，也間接的使用了 Ratnaoarkhi 等人 [20] 從 PTB 0.5 彙整而成的 RRR 語料庫。此外，我們也從華爾街日報[1]與紐約時報[2]的網站上蒐集報導。WordNet 這部辭典是用於查詢詞彙意義，用以將詞彙抽象化。本章第一部分為語料介紹，包含 RRR、PTB 3 的華爾街日報和網路上蒐集的華爾街日報與紐約時報；第二部分為介紹 WordNet 和詞彙概念；第三部分為語料前處理，包含斷句與剖析、中心抽取、雜訊過濾和挑選出具挑戰性的介系詞；第四部分是前處理完畢的目的語料。

3.1　語料庫

我們使用的語料庫共有三種種類。RRR 與 PTB3 是目前現有的語料庫，其中 PTB3 會經由前處理處理成 RRR 的格式。華爾街日報與紐約時報則是從網路上蒐集而來，經由斷句、剖析後，再辨識出「動詞 -名詞片語一 -介系詞 -名詞片語二」的結構，並抽取結構的中心詞，最後處理成 RRR 格式。本研究使用 RRR 與 PTB3 來作為介系詞片語定位問題的語料庫，用 RRR 以及自行蒐集的報導資料當作介系詞推薦的語料庫。

[1] http://asia.wsj.com/home-page
[2] http://www.nytimes.com/

3.1.1 RRR

RRR 是由 Ratnaparkhi、Reynar 和 Roukos 三人製作，因此合稱為 RRR 語料庫。RRR 語料庫是從 PTB0.5 版中分離出來的，其中每一筆資料都紀錄 PTB0.5 動詞片語中的四個中心詞與定位標記，如表 3.1所示，我們將這種紀錄方式稱之為 RRR 格式。

表 3.1裡每一筆資料都是實際語料庫所表示。每筆資料在資料庫裡都有一個編號表示 PTB0.5 的原句，並且有標記該介系詞片語的定位對象：以 V 表示修飾動詞，而以 N 表示修飾名詞一。RRR 語料裡共有 23500 多句，分布情況可參考表 1.1。

表 3.1: RRR 語料庫實際範例

編號	動詞	名詞一	介系詞	名詞二	定位
0	join	board	as	director	V
1	is	chairman	of	N.V.	N
2	named	director	of	conglomerate	N

3.1.2 PTB3

Penn Treebank 是一個將自然語言結構化的資料庫，在許多自然語言處理的研究都被視為是黃金標準。本研究使用的版本是現行版本第三版，其中內容包含了三年份華爾街日報共 2499 篇報導，共有 98732 句結構化的句子，並且將這些句子分成 25 節。採用 Penn Treebank 風格做語法標記。

本研究中所關心句子是含有介系詞的動詞片語，例如說，圖 3.1是 PTB3 裡一個真實的結構樹，而底線部分是動詞片語也就是我們關心的結構。第 3 行到第 9 行是一個 VP 表示動詞片語，第 4 行到第 7 行是 NP 表示名詞片語，第 8 行到第 9 行是 PP 表示介系詞片語，在 PP 內部的第 9 有一個 NP 表示名詞片語，前者稱為名詞片語一，後者稱為名詞片語二。在這四個片語中都各包含了一個中心詞，圖中以粗體表示中心詞，動詞片語中心詞是“set”，名詞片語一的中心詞是“floor”，介系詞片語中心詞“on”，最後

名詞片語二的中心詞為“bidding”。

```
1.   ( (S
2.       (NP-SBJ (DT The) (JJ Venezuelan) (JJ central) (NN bank) )
3.       (VP (VBD set)
4.           (NP
5.               (NP (DT a)
6.               (ADJP (CD 30) (NN %) )
7.               (NN floor) )
8.           (PP (IN on)
9.               (NP (DT the) (NN bidding) ))))
10.      (. .) ))
```

圖 3.1: PTB3 的結構樹

3.1.3 華爾街日報與紐約時報

我們從華爾街日報與紐約時報的網站上蒐集了 2011 年部分報導內容，其中包含了華爾街日報的 68983 句和紐約時報的 55358 句。內容屬性上，華爾街日報是屬於財經類報導，而紐約時報是屬於綜合類的報導。這二類報導文章的句型句法都是屬於較現代的用法。

圖 3.2是一篇實際的段落文章，目標是抓取動詞片語的四個中心詞。以圖 3.2的第一句為例，底線部分是動詞片語，粗體是中心詞。

Officials have long shunned proposals that would make banks and other creditors **share** some **losses on** Greek **debt**. But European leaders are taking the calculated risk that they can avoid spooking investors by expanding the aid package to include other troubled countries on Europe’s periphery.

圖 3.2: 紐約時報的段落文章

3.2 詞彙資料庫：WordNet

WordNet 是一個詞彙的資料庫，收入動詞、名詞、形容詞和副詞四種詞性的詞彙，並以階層式的架構描述詞彙語義的關係，我們使用的版本是 WordNet3.0。對於 WordNet 我們所關心的詞性是動詞和名詞，以及 WordNet 描述詞彙之間的二種關係。這二種關係的概念分別是：（一）上下位詞關係或 *IS A* ，越在上位的詞彙表示越是越抽象的概念，越下位則是越具體的概念；（二）種類（lexicographic），其中動詞有 25 種種類，名詞則有 15 種種類，可參考附錄 I。圖 3.3是一個 WordNet 的名詞結構範例，每一個節點可以分為上下二部分：上半部分表示同義詞集（synset），在 WordNet 收入的名詞與動詞的同義詞集共有 171359 個[3]。若一個節點有二個以上的詞彙，表示這些詞彙互為同義詞，可以互相替換，例如“cutlery”和“eating utensil”互為同義詞而我們以 $\{cutlery, eating\ utensil\}$ 來表示一個同義詞集。樹狀結構的父母與小孩的關係是上下位詞或 *IS A* 的概念，例如，“spoon”的下位詞是“wooden spoon”，上位詞是“cutlery”或“eating utensil”，我們以 $\{wooden\ spoon\} - \{spoon\} - \{cutlery, eating\ utensil\}$ 來表示這種關係。從這個例子我們可以看到上位詞的概念比下位詞抽象，因此我們定義同義詞集的抽象化是以該同義詞集的上位詞表示；圖 3.3中一個節點下半部表示的是這個同義詞集的種類，各種種類之間也是有階層式的關係。

許多研究都把 WordNet 解釋為是一個樹狀結構的辭典，我們這邊也是如此。一旦將 WordNet 簡化當成樹狀結構，我們定義越在上位的詞彙稱為越高層，反之越下位的詞彙稱為越低層。由上下位詞的觀點來看，$\{entity\}$ 是所有名詞的最抽象化的詞，也就是根節點（root）；從種類的觀點來看，$\{entity\}$ 是屬於 *noun.Tops* 種類。*noun.Tops* 位於種類的最上層，也被稱為唯一始點（unique beginners）[4]。同樣都是種類的始點還有 11 個抽象概念的同義詞集；動詞不同於名詞的結構，動詞並不具有一個共通的根節點，它

[3] http://wordnet.princeton.edu/wordnet/man/wnstats.7WN.html
[4] http://wordnet.princeton.edu/wordnet/man/uniqbeg.7WN.html

是一個森林結構。不過有時在某些研究上為了某些原因，會再在森林的結構上再多加一個虛擬的根節點 $\{**root**\}$，使森林變為樹狀結構。

雖然簡化成樹狀結構會有利於我們使用以及解釋 WordNet，但 WordNet 實質上並不是樹狀結構，例如說在圖 3.3右下角的同義詞集 $\{wooden\ spoon\}$ 有 $\{woodenware\}$ 和 $\{spoon\}$ 二個上位詞。所以在本研究中，我們不將它簡化為樹狀結構，而是將它視為是一個單向網路結構。

在現實的生活中，一個詞彙可能會有多個意思，而透過 WordNet 我們也可以將這些意思都找出，這些不同的意思在 WordNet 裡，稱之為詞義（sense）。例如“spoon”這個詞彙作“湯匙”解釋時，對應到 WordNet 是同義詞集 $\{spoon\}$；而當“spoon”作“一匙的量”解釋時，對應到 WordNet 則是同樣詞集 $\{spoon, spoonful\}$，而這二個由“spoon”所查詢的同義詞集的上位詞也不一樣。像這樣一個詞彙可以有多個意義的問題被稱為語義歧義，在本研究中，我們不解決語義歧義問題。

除了詞彙的語義查詢之外，WordNet 還提供我們一個詞彙當作某一個詞義的頻率。以“spoon”為例，若表示 $\{spoon\}$ 同義詞集，則頻率是 1。若表示 $\{spoon, spoonful\}$ 這個同義詞集，則頻率也是 1。若以“spoonful”查詢 WordNet，則 $\{spoon, spoonful\}$ 頻率是 0。

本研究使用 WordNet 的目的是用以將詞彙抽象化，也就是我們使用 WordNet 將詞彙的概念作更概括性的解釋。在這裡，我們透過查詢同義詞集的上位詞來達到抽象化的目的。比較句 9的“a spoon”和句 10的“a fork”，“a spoon”和“a fork”都是屬於動詞片語中的名詞片語二。這二句看起來是相似的句子，在介系詞片語定位的問題裡，一般直覺上，比較容易想像這二個句子的介系詞片語定位於動詞“ate”。因為對於“ate”這個動詞而言，“spoon”和“fork”都是一種進食的工具，所以如果我們同樣把名詞二代換為任何種類進食的工具，如圖 3.3下方的虛框中的詞彙“table knife”、“wooden spoon”和“tablespoon”等，那麼我們應該也可以很大膽的推測在一般情況下這樣的例句皆是

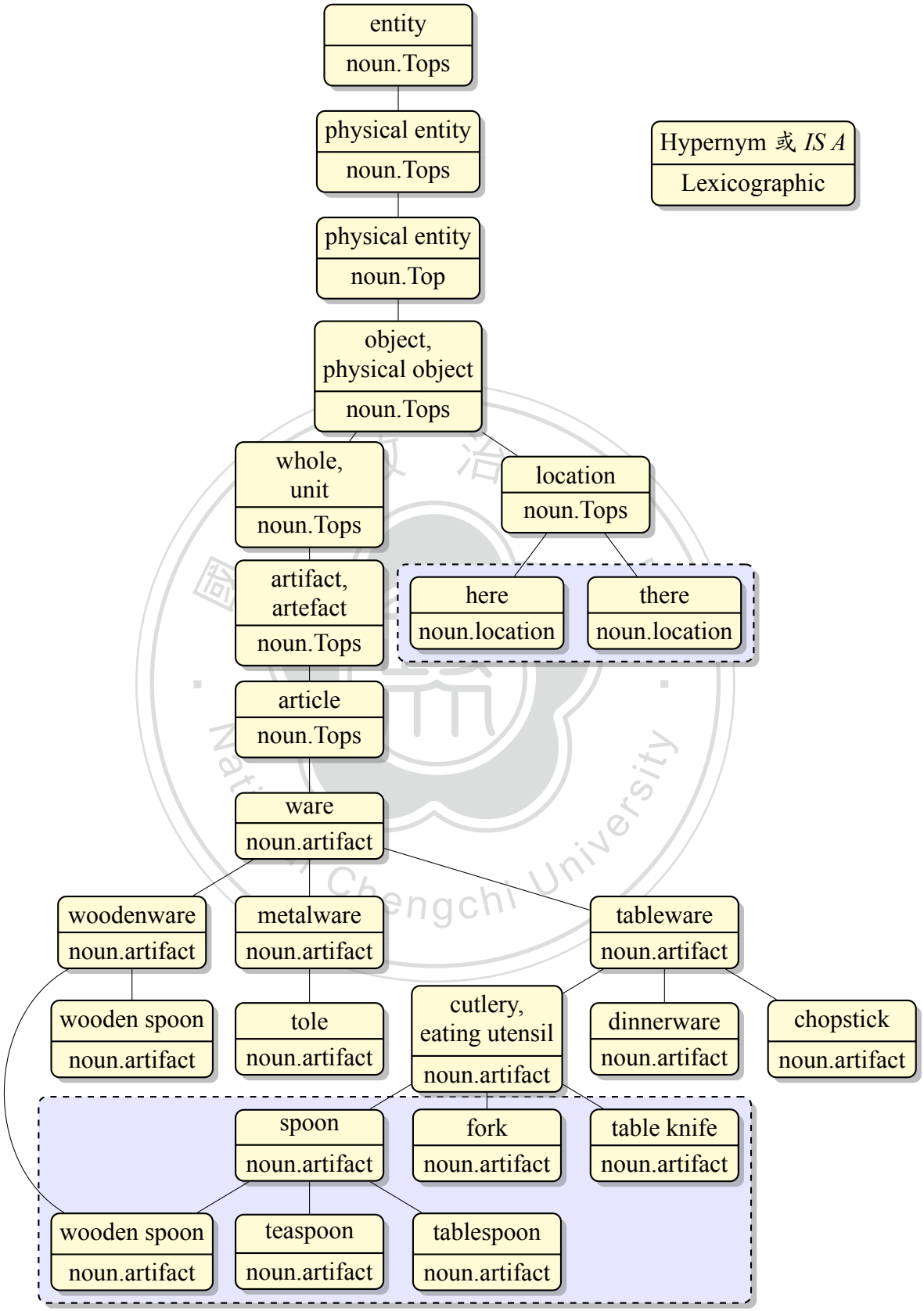
entity
noun.Tops
physical entity
noun.Tops
physical entity
noun.Top
object,
physical object
noun.Tops
whole,
unit
noun.Tops
location
noun.Tops
artifact,
artefact
noun.Tops
here
noun.location
there
noun.location
article
noun.Tops
ware
noun.artifact
woodenware
noun.artifact
metalware
noun.artifact
tableware
noun.artifact
wooden spoon
noun.artifact
tole
noun.artifact
cutlery,
eating utensil
noun.artifact
dinnerware
noun.artifact
chopstick
noun.artifact
spoon
noun.artifact
fork
noun.artifact
table knife
noun.artifact
wooden spoon
noun.artifact
teaspoon
noun.artifact
tablespoon
noun.artifact
Hypernym 或 IS A
Lexicographic

圖 3.3: WordNet

定位於動詞。在介系詞推薦的問題裡，同樣的不管把名詞二代換成何種進食工具，都不影響介系詞是使用“with”。在這個例子中，這些可互相代換的名詞二，它們的共同點是在逐步抽象化的過程中擁有共同的上位詞 $\{cutlery, eatingutensil\}$，透過這樣的抽象化，我們便有機會解決這二個介系詞的問題。圖 3.3右上的虛框中的“here”與“there”這二個詞彙在語義上也有相似的部分，因此，透過抽象化的過程，我們就會找出這兩個詞彙共有的上位詞 $\{location\}$。

句 9. The children ate the cake with a spoon.

句 10. The children ate the cake with a fork.

3.3 前處理

前處理的部分包含了句子的斷句與剖析、中心詞抽取、雜訊過濾以及挑選有挑戰性的介系詞等工作。流程圖可參考圖 3.4，圖中上半部是前處理的流程，下半部表示的是語料庫進入前處理的階段。使用華爾街日報與紐約時報需要從斷句與剖析句子的流程開始處理；使用 PTB3 語料庫，則是從結構樹中抽取中心詞的流程開始處理；使用 RRR 語料庫直接從雜訊過濾開始處理。最後所有語料彙整成 RRR 的資料格式，再統一處理雜訊。雜訊過濾是一件重要的工作，雜訊包含了中心詞是定冠詞、代名詞等情況或是碰撞問題等情況。對於介系詞片語定位問題，挑選挑戰性介系詞是找出修飾動詞與修飾名詞機率相近的介系詞。每個語料庫介系詞分布情況大致上差不多，但仍有些許差異，因此我們以 RRR 語料庫為主。對於介系詞推薦的問題，則是找到數量較多或是差不多的介系詞。

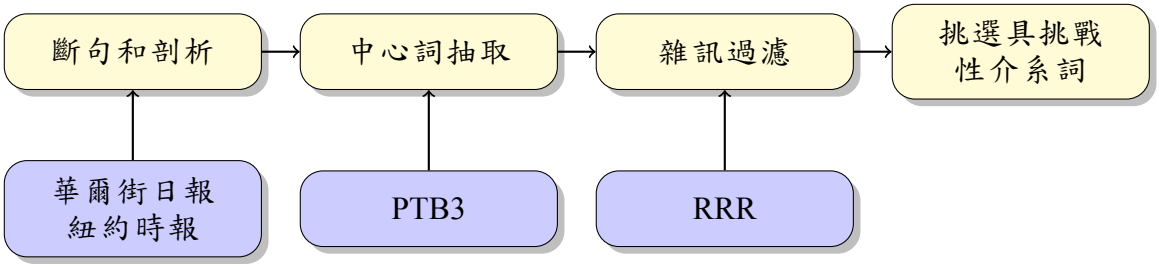

圖 3.4: 前處理流程圖

3.3.1 句子剖析與斷句

我們從華爾街日報與紐約時報的網站上蒐集大量的文章報導，並將文章斷句與剖析成結構樹。我們先利用 Stanford 剖析器[5]與 Lingpipe[6]將所搜集的語料斷句，僅留下二者斷句結果有共識的句子。接著再利用 Stanford 剖析器剖析留下的句子，使用的文法（grammar）是 wsjFactored.ser.gz，並將 MAX_ITEMS 參數設為 500000，剖析後可得到結構樹。

3.3.2 中心詞抽取

我們的目標是從結構樹抽出動詞片語的四個中心詞，圖 3.5和圖 3.6表示二種的動詞片語的結構，以 Penn Treebank 風格表示，二圖分別表示介系詞片語修飾對象是名詞和動詞。圖 3.5和圖 3.6裡 VP 下方最左邊的節點表示是不同形態的動詞，如過去式、過去分詞等；IN 表示的是介系詞；而在 Penn Treebank 風格的語法標記下，“to”這個介系詞會另外被表示成 TO。圖 3.5與圖 3.6這二個結構最大的不同點在於 PP 這個節點是掛在 NP 或是 VP 之下。我們以圖 3.1中的結構樹作為中心詞抽取的例子，句 11是將圖 3.1平面化之後的結果，底線是我們要抽取的目標，它符合圖 3.6結構。

[5]Stanford Parser 2.0 版（2012 年 2 月 3 日），http://nlp.stanford.edu/software/lex-parser.shtml
[6]http://alias-i.com/lingpipe/

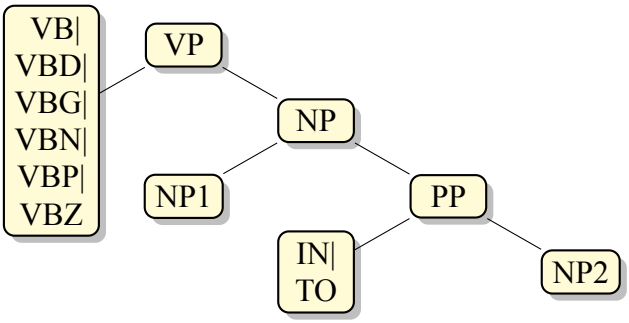

圖 3.5: 動詞片語: 修飾名詞

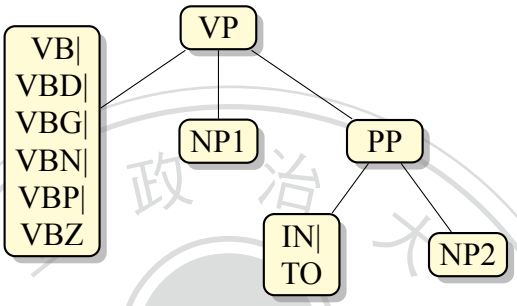

圖 3.6: 動詞片語: 修飾動詞

我們使用 Stanford Tregrex[7]並利用圖 3.7和圖 3.8的樣式比對含有上述二種動詞片語結構的句子。如果一個句子符合圖 3.7和圖 3.8的樣式，就可以從動詞片語中比對四個主要詞組結構，句 11被比對的詞組結構如表 3.2裡片語一欄所示。再利用 Stanford 剖析器的 SemanticHeadFinder[8]類別將動詞片語的四個主要詞組結構的中心詞找出，最後得到的結果如表 3.2裡中心詞一欄所示。

句 11. ((S (NP-SBJ (DT The) (JJ Venezuelan) (JJ central) (NN bank))
(VP (VBD set) (NP (PP (NP (DT a)(ADJP (CD 30) (NN %))(NN floor))
(IN on)(NP (DT the) (NN bidding)))))(. .)))

[7]Stanford Tregrex 2.0.1 版（2012 年 1 月 6 日），http://nlp.stanford.edu/software/tregex.shtml
[8]http://nlp.stanford.edu/nlp/javadoc/javanlp/edu/stanford/nlp/trees/SemanticHeadFinder.html

```
"@/VP.?/  [" +
  "< (VB=verb $++ (NP < (@/NP.?/=np1 $++ (@/PP.?/=pp < (IN=prep < PREP $++ @/NP.?/=np2))))) |" +
  "< (VBD=verb $++ (NP < (@/NP.?/=np1 $++ (@/PP.?/=pp < (IN=prep < PREP $++ @/NP.?/=np2))))) |" +
  "< (VBG=verb $++ (NP < (@/NP.?/=np1 $++ (@/PP.?/=pp < (IN=prep < PREP $++ @/NP.?/=np2))))) |" +
  "< (VBN=verb $++ (NP < (@/NP.?/=np1 $++ (@/PP.?/=pp < (IN=prep < PREP $++ @/NP.?/=np2))))) |" +
  "< (VBP=verb $++ (NP < (@/NP.?/=np1 $++ (@/PP.?/=pp < (IN=prep < PREP $++ @/NP.?/=np2))))) |" +
  "< (VBZ=verb $++ (NP < (@/NP.?/=np1 $++ (@/PP.?/=pp < (IN=prep < PREP $++ @/NP.?/=np2))))) |" +
  "< (VB=verb $++ (NP < (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2))))) |" +
  "< (VBD=verb $++ (NP < (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2))))) |" +
  "< (VBG=verb $++ (NP < (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2))))) |" +
  "< (VBN=verb $++ (NP < (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2))))) |" +
  "< (VBP=verb $++ (NP < (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2))))) |" +
  "< (VBZ=verb $++ (NP < (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2)))))" +
  "]"
```

圖 3.7: 修飾名詞樣式


```
"@/VP.?/  [" +
  "< (VB=verb $++ (@/NP.?/=np1 $++ (@/PP.?/=pp < (IN=prep < PREP $++ @/NP.?/=np2)))) |" +
  "< (VBD=verb $++ (@/NP.?/=np1 $++ (@/PP.?/=pp < (IN=prep < PREP $++ @/NP.?/=np2)))) |" +
  "< (VBG=verb $++ (@/NP.?/=np1 $++ (@/PP.?/=pp < (IN=prep < PREP $++ @/NP.?/=np2)))) |" +
  "< (VBN=verb $++ (@/NP.?/=np1 $++ (@/PP.?/=pp < (IN=prep < PREP $++ @/NP.?/=np2)))) |" +
  "< (VBP=verb $++ (@/NP.?/=np1 $++ (@/PP.?/=pp < (IN=prep < PREP $++ @/NP.?/=np2)))) |" +
  "< (VBZ=verb $++ (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2)))) |" +
  "< (VB=verb $++ (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2)))) |" +
  "< (VBD=verb $++ (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2)))) |" +
  "< (VBG=verb $++ (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2)))) |" +
  "< (VBN=verb $++ (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2)))) |" +
  "< (VBP=verb $++ (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2)))) |" +
  "< (VBZ=verb $++ (@/NP.?/=np1 $++ (@/PP.?/=pp < (TO=prep < PREP $++ @/NP.?/=np2))))" +
  "]"
```

圖 3.8: 修飾動詞樣式

表 3.2: 中心詞抽取

	片語	中心詞
動詞	(VBD set)	set
名詞片語一	(NP (NP (DT a) (ADJP (CD 30) (NN %)) (NN floor)))	floor
介系詞	(IN for)	for
名詞片語二	(NP (DT the) (NN bidding))	bidding

3.3.3 雜訊過濾

雜訊過濾的目的是為了提升語料的品質，增進實驗的可靠度。底下我們將一一介紹在本研究中被視為雜訊的情況。

通常我們會認為詞彙有一些固定的用法，因此通常我們假設詞彙有固定的語義與詞性。舉個例子，“google”在一般的認知中，很直覺地會被認定是名詞，因為這是一個公司的名稱。然而“google”在目前可以說是國際上知名度最高的搜尋引擎[9]，因此不僅成了搜尋引擎的代名詞，更成了搜尋的別稱，“google”一詞也因此出現了新的用法，例如我們有時可能會聽到“Have you ever googled that?”，在這個例句中“google”已經被當成是動詞在使用了，然而“google”作為動詞的用法在以前的辭典中是不會被記載。

在我們的語料庫裡，也有不少句子因為語法上的關係，可能會有一些特殊的符號和數字被當成是中心詞，例如：“%”可能會出在名詞的位置。然而這些符號和數字在我們的方法中是很難抽象化的，因此我們會事先將這些符號和數字過濾。

除了上述情況之外，我們也發現雖然 RRR 的語料庫經由 Ratnaparkhi 等人整理過，但 Pantel 和 Lin[18] 在 RRR 語料庫裡找到 133 筆名詞一或名詞二為“the”，PTB3 裡也有出現“the”的被當成是名詞的案例。另外在 RRR 與 PTB3 語料庫也均有一些名詞是“a”或“an”的情況。類似的情況，我們亦將之視為雜訊。

此外，我們會先利用 WordNet 做詞幹還原。接著，再給定還原後的詞彙和詞性，

[9] 2012 年 9 月 5 日

如果 WordNet 沒有查詢任何同義詞集，那麼也會被過濾。

Coppola 等人 [5] 曾提到，如果名詞一是代名詞，則介系詞片語有較高的機率是定位於動詞。另一方面，代名詞不被收入於 WordNet 內，因此當名詞是代名詞的情況在我們的二個研究問題中也會過濾。

對於介系詞定位問題，碰撞是指當有二個以上的動詞片語具有四個相同的中心詞但介系詞定位卻不相同的情況。對於介系詞推薦問題，碰撞是指當動詞、名詞一和名詞二相同，但介系詞有二個以上的情況。上述這二類的案例，目前在本研究中暫不處理，因此也將之視為雜訊。

當我們處理 PTB3 的語料時，會發現某些句子因為語境等因素而使得名詞一被省略而標記成“-NONE- *”，此時若使用圖 3.7和圖 3.8的樣式比對，這類的語料也會被我們比對到。然而這類的語料實際上我們無從得知本來的名詞一，因此這類的情況也被我們過濾。

3.3.4 挑選具挑戰性的介系詞

挑選具挑戰性的介系詞的目的是希望可以挑選出較值得做實驗的介系詞。而挑戰性的介系詞對於介系詞片語定位與介系詞推薦問題各有不同的定義。

以往有許多學者的目標是希望能做一套適用一般化介系詞片語定位的分類器，但根據我們初步的觀察表 1.1，每種介系詞對於修飾名詞與修飾動詞都有不同偏好程度，因此我們認為如果能夠針對各別的介系詞片語做分類，效果可能會比較好。

但有些介系詞片語定位偏好很明顯，隨意猜測也可以達到不錯的效果，例如表 1.1中的“of”，我們可以看到“of”相當的偏好修飾名詞，根據其修飾名詞與修飾動詞的數量來看，只要直接猜測“of”是修飾名詞，那麼約有高於 9 成的機率可以答對，因此，我們希望可以排除這種情況的介系詞。

我們採用 Entropy 如式 (3.1) 和頻率這二個指標來衡量某一個介系詞對於修飾名詞

與修飾動詞的偏好程度，數值越大表示偏好二類的程度越相當，是越具有挑戰性的介系詞。我們的目的是找數量多且平衡的介系詞，可惜往往數量多的介系詞也是最不平衡的而數量少的介系詞則反而會比較平衡。

$$Entropy = \sum_{d \in D} -Pr(d) \log_2 Pr(d) \tag{3.1}$$

式 (3.1) 中，以“of”為例，介系詞片語定位問題 D 只有二個分類：修飾名詞與修飾動詞。因此 $D = \{VPP, NPP\}$，$Pr(d)$ 為修飾名詞或修飾動詞所佔的比例。所以我們可以知道 $Pr(NPP) = 6553/6614$、$Pr(VPP) = 16/6614$，最後可以計算出 Entropy。

雖然比起介系詞數量的多寡，我們更重視介系詞平衡的情況，但是我們也不希望介系詞數過少。因此我們設立雙重門檻值的篩選機制將介系詞分成二個等級：第一個等級是最平衡且滿足一定的數量的介系詞，第二個等級是修飾名詞與修飾動詞比例約為 2 比 3、1 比 2 或反之的情況，但仍是滿足一定數量的門檻值限制。透過這樣的門檻值限制，我們可以找到較具挑戰性的介系詞，並針對個別的介系詞做特製化的分類器。

在介系詞推薦的問題裡，我們會從語料庫挑選數量較多或是數量上較接近的介系詞做實驗。

3.4 目的語料

目的語料是我們經由 3.3節前處理的方法得到的結果。介系詞片語定位問題的語料，我們將以 RRR 所選到的介系詞為主，實際上，在我們的統計中，每個語料庫的介系詞分布幾乎都是差不多的。介系詞推薦問題則是依個別介系詞數量多寡不同而選擇的介系詞。對這二個問題我們設定了一個分布線（Distribution），分布線的意義是亂猜可以達到最佳的精準度。

介系詞片語定位問題，根據的介系詞數量分布的情況，分布線定義，如式 (3.2)。

式 (3.2) 的 $Pr(VPP)$ 與 $Pr(NPP)$ 表示修飾動詞與修飾名詞在語料庫裡佔得總量的比例。

$$Distribution = Max(Pr(VPP), Pr(NPP)) \tag{3.2}$$

介系詞推薦問題，我們比較著重於分析各個介系詞分類的情況，因此設定以介系詞在語料庫佔得總量計算分布線，如式 (3.3)，其中 $|x|$ 表示 x 出現的頻率。而總體的分布線，則是以單獨分布線較高者為準。

$$Distribution = \frac{|Preposition|}{|Total|} \tag{3.3}$$

3.4.1 介系詞片語定位語料

表 3.3、表 3.4和表 3.5是 RRR 語料經由篩選過慮後的結果，我們選出“for”、“on”、“in”、“with”、“from”和“to”，其中前三個介系詞都是數量多且較平衡的情況，而後三者則是數量多但較不平衡的情況，混合表示是將這六個介系詞一起做實驗。訓練語料、驗證語料和測試語料的分布大致上都是差不多。

表 3.6、表 3.7和表 3.8是 PTB3 過濾後的結果，分布的情況與 RRR 大致是相同的。為與 Stanford 剖析器做比較，我們選用 PTB3 的 02 到 21 節做測試語料，22 節做驗證語料，00、01、23 和 24 節做測試語料。

表 3.9是表 3.8測試語料的原句句數統計。若將原句給 Stanford 剖析器剖析，可能會出現原本是動詞片語的結構，但 Stanford 剖析器卻無法辨識；或者不是動詞片語，但卻被 Stanford 剖析器誤認的情況。若將這二種情況去除可得表 3.10，細節參考 5.3.5。

表 3.3: RRR 前處理結果: 訓練語料

介系詞	v	n	總數	Entropy	v(%)	n(%)	分布線 (%)
for	829	869	1698	0.9996	48.82	51.18	51.18
on	512	485	997	0.9995	51.35	48.65	51.35
in	1392	1314	2706	0.9994	51.44	48.56	51.44
with	454	268	722	0.9516	62.88	37.12	62.88
from	451	237	688	0.9290	65.55	34.45	65.55
to	1145	394	1539	0.8207	74.40	25.60	74.40
混合	4783	3567	8350	0.9846	57.28	42.72	57.28

表 3.4: RRR 前處理結果: 驗證語料

介系詞	v	n	總數	Entropy	v(%)	n(%)	分布線 (%)
for	147	169	316	0.9965	46.52	53.48	53.48
on	120	100	220	0.9940	54.55	45.45	54.55
in	272	289	561	0.9993	48.48	51.52	51.52
with	73	47	120	0.9659	60.83	39.17	60.83
from	74	52	126	0.9779	58.73	41.27	58.73
to	190	82	272	0.8831	69.85	30.15	69.85
混合	876	739	1615	0.9948	54.24	45.76	54.24

表 3.5: RRR 前處理結果: 測試語料

介系詞	v	n	總數	Entropy	v(%)	n(%)	分布線 (%)
for	111	148	259	0.9852	42.86	57.14	57.14
on	66	93	159	0.9791	41.51	58.49	58.49
in	156	200	356	0.9890	43.82	56.18	56.18
with	55	35	90	0.9641	61.11	38.89	61.11
from	60	32	92	0.9321	65.22	34.78	65.22
to	135	76	211	0.9428	63.98	36.02	63.98
混合	583	584	1167	1.0000	49.96	50.04	50.04

表 3.6: PTB3 前處理結果: 訓練語料

介系詞	v	n	總數	Entropy	v(%)	n(%)	分布線 (%)
for	732	892	1624	0.9930	45.07	54.93	54.93
on	512	523	1035	0.9999	49.47	50.53	50.53
in	1531	1241	2772	0.9921	55.23	44.77	55.23
with	450	269	719	0.9538	62.59	37.41	62.59
from	441	290	731	0.9690	60.33	39.67	60.33
to	1064	335	1399	0.7941	76.05	23.95	76.05
混合	4730	3550	8280	0.9853	57.13	42.87	57.13

表 3.7: PTB3 前處理結果: 驗證語料

介系詞	v	n	總數	Entropy	v(%)	n(%)	分布線 (%)
for	23	39	62	0.9514	37.10	62.90	62.90
on	30	22	52	0.9829	57.69	42.31	57.69
in	72	61	133	0.9951	54.14	45.86	54.14
with	10	5	15	0.9183	66.67	33.33	66.67
from	14	10	24	0.9799	58.33	41.67	58.33
to	34	16	50	0.9044	68.00	32.00	68.00
混合	183	153	336	0.9942	54.46	45.54	54.46

表 3.8: PTB3 前處理結果: 測試語料

介系詞	v	n	總數	Entropy	v(%)	n(%)	分布線 (%)
for	115	189	304	0.9568	37.83	62.17	62.17
on	104	101	205	0.9998	50.73	49.27	50.73
in	288	289	577	1.0000	49.91	50.09	50.09
with	74	51	125	0.9754	59.20	40.80	59.20
from	77	46	123	0.9537	62.60	37.40	62.60
to	182	77	259	0.8780	70.27	29.73	70.27
combine	840	753	1593	0.9978	52.73	47.27	52.73

表 3.9: PTB3 前處理結果：測試語料原句句數

介系詞	句數
for	296
on	203
in	546
with	122
from	120
to	232

表 3.10: PTB3 前處理結果：與 Stanford 剖析器比較用途之測試語料

介系詞	v	n	總數	Entropy	v(%)	n(%)	分布線 (%)
for	89	158	247	0.9430	36.03	63.97	63.97
on	87	86	173	1.0000	50.29	49.71	50.29
in	233	236	469	1.0000	49.68	50.32	50.32
with	63	48	111	0.9868	56.76	43.24	56.76
from	66	39	105	0.9518	62.86	37.14	62.86
to	153	64	217	0.8750	70.51	29.49	70.51
混合	691	631	1322	0.9985	52.27	47.73	52.27

3.4.2 介系詞推薦語料

表 3.11、表 3.12和表 3.13是使用 RRR 語料所挑選出的介系詞，在 RRR 語料數量較少的情況下，我們盡量挑選了數量較多且差不多的介系詞，測試語料的數量是希望最少也接近 100 筆左右。“of”是屬於數量較多的介系詞，但由於它的數量級與其它 6 個介系詞相差過多，因此“of”不在我們的實驗之中。

表 3.14、表 3.15和表 3.16是華爾街日報與紐約時報前處理後的結果，這是數量較大的語料庫，我們將處理後的語料以 6 比 2 比 2 切成訓練、驗證、測試語料，並從中選出數量較多的 11 個介系詞做實驗。整體而言，分布與 RRR 的結果是差不多的。

表 3.11: RRR 前處理結果: 訓練語料

介系詞	數量	分布線 (%)
for	1604	20.64
on	945	12.16
in	2471	31.80
with	650	8.85
from	688	8.36
to	1413	18.18
總數	7771	31.80

表 3.12: RRR 前處理結果: 驗證語料

介系詞	數量	分布線 (%)
for	310	19.61
on	214	13.54
in	549	34.72
with	118	7.46
from	125	7.91
to	265	16.76
總數	1581	34.72

表 3.13: RRR 前處理結果: 測試語料

介系詞	數量	分布線 (%)
for	263	22.91
on	158	13.76
in	345	30.05
with	90	7.84
from	91	7.93
to	201	17.51
總數	1148	30.05

表 3.14: 華爾街日報與紐約時報前處理結果: 訓練語料

介系詞	數量	分布線 (%)
of	7341	28.36
in	5353	20.68
for	2892	11.17
to	2471	9.55
on	2248	8.68
with	1625	6.28
from	1300	5.02
at	1109	4.28
as	694	2.68
by	522	2.02
about	329	1.27
總數	25884	28.36

表 3.15: 華爾街日報與紐約時報前處理結果: 驗證語料

介系詞	數量	分布線 (%)
of	2424	28.09
in	1828	21.19
for	894	10.36
to	829	9.61
on	725	8.40
with	596	6.91
from	451	5.23
at	369	4.28
as	211	2.45
by	171	1.98
about	130	1.51
總數	8628	28.09

表 3.16: 華爾街日報與紐約時報前處理結果: 測試語料

介系詞	數量	分布線 (%)
of	2390	27.71
in	1801	20.88
for	916	10.62
to	892	10.34
on	768	8.91
with	572	6.63
from	413	4.79
at	359	4.16
as	239	2.77
by	162	1.88
about	112	1.30
總數	8624	27.71

第 4 章　研究方法

經過第3章語料處理後，可以得到動詞片語的四個中心詞：動詞、名詞一、介系詞和名詞二。在我們的研究裡，我們將這四個中心詞視為是已知條件，在這樣的條件下，對介系詞片語定位與介系詞推薦問題建構一般化的模型。研究方法可以分成三部分，第一部分特徵處理將介紹特徵數值化；第二部分介紹利用 WordNet 做階層式的特徵選擇；第三部分介紹三種不同階段的模型建構方法。

4.1　特徵處理

經過 WordNet 查詢而來的同義詞集被我們視為是特徵，然而這樣的特徵只是一個符號，但在現行許多機器學習的演算法，大多都是需要量化後的數值，因此如何將特徵量化是一個重要的議題。

本節特徵的處理包含了特徵量化與特徵加權，特徵加權可視為是廣義特徵量化的過程，因為加權本身也是將特徵數值化的一個過程。而本這節特徵量化特別強調如何表現特徵的存在，我們稱之為狹義特徵量化的定義。特徵加權主要是基於狹義特徵量化再給予不同的詮釋面向。

廣義特徵量化的過程在目前並沒有一個公認的做法，將一個文字符號量化的過程很容易因為研究者本身的偏見（bias）而造成資訊的遺失，而往往這些遺失的資訊會造

成最後研究的成果有所偏差。但若研究者本沒有偏見，也無法向前邁進。所以如何量化特徵才是合理，只能靠研究者對於特徵量化的詮釋與實驗的驗證。

就本節所指的狹義特徵量化的方法可分為三種：二元（binary）法、平均法和累計法，特徵加權則是考慮詞義頻率與語義深度二種方法，其中針對平均法與累計法量化的結果再乘上特徵加權的結果。量化的過程中，我們將會考慮一個詞彙的所有詞義包含所有同義詞集與該同義詞集至根節點間所有的上位詞，這些上位詞同時也是同義詞集。

4.1.1 特徵量化

我們對於這三種量化的方式都有不同的詮釋：在一個透過 WordNet 查詢的詞彙中，從查詢到的第一個同義詞集到根節點所有的同義詞集，二元法考慮的面向是將所有節點都視為是均等的存在；平均法考慮的是每個節點平均負擔的語義；累計法考慮的是每一條至根節路徑中每一個節點被使用的頻率。下面我們將一一介紹每一種量化的方式：

二元法

二元法表示我們的特徵值只有 1 與 0，這代表所有同義詞集都視為是均等的存在。一個詞彙透過 WordNet 可以查詢的同義詞集以及至根節點間所有的同義詞集，凡是用到的同義詞集均以 1 表示，反之沒有用到以 0 表示。

以動詞“eat”為例，如圖 4.1所示，圖中是所有“eat”的同義詞集以及至根節點間所有的同義詞集，所有的節點都被標示為 1。

平均法

二元法單純只考慮了同義詞集出現與否，然而一個同義詞集可能會有二個以上的上位詞，這使得一個同義詞集到根節點的路徑不只一條，因此我們認為這些分叉的路徑應該

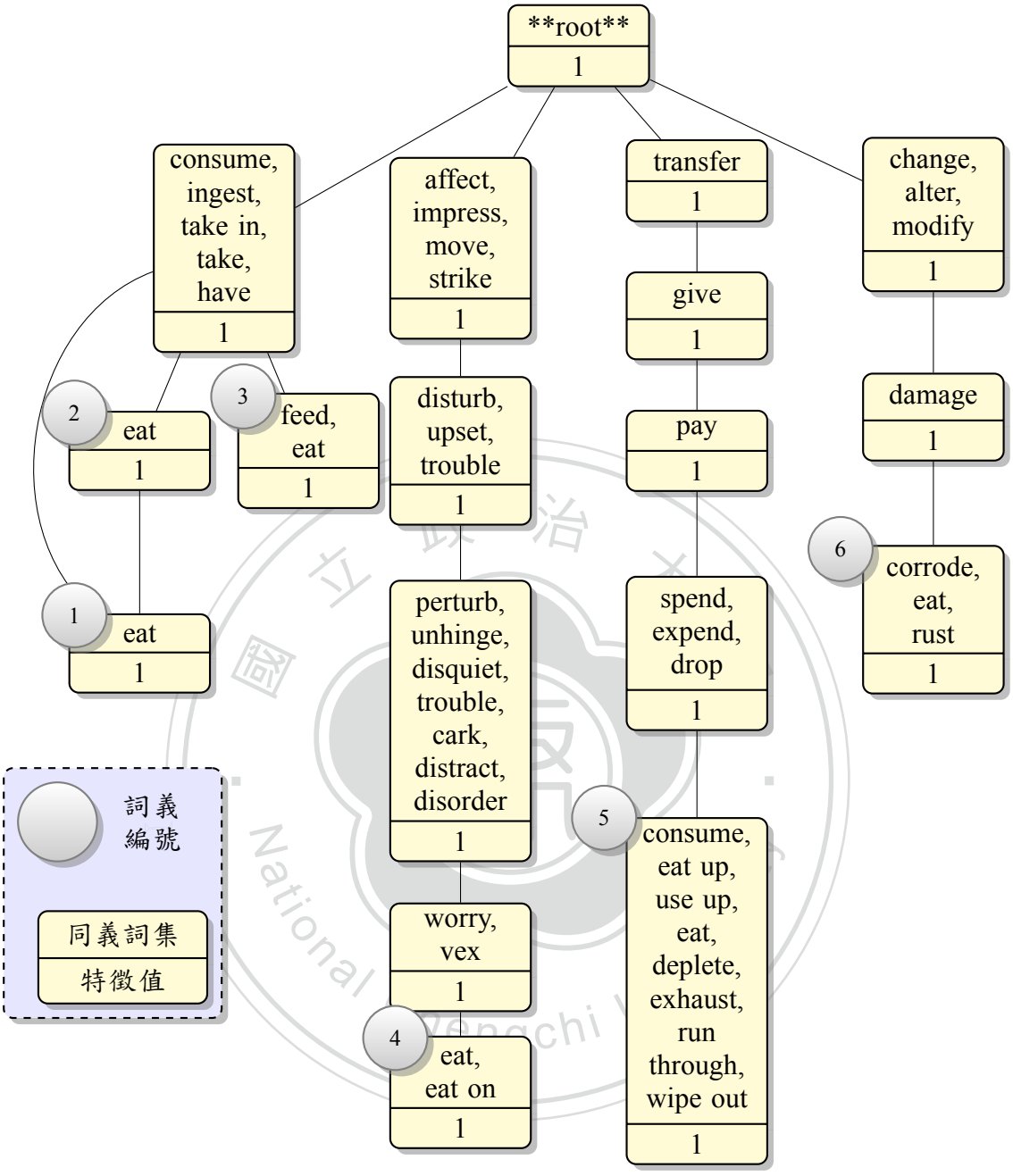

圖 4.1: 二元量化

平均分擔這個詞彙的語義，這代表每個同義詞集在該詞彙裡平均負擔的語義。將原本是二元法的特徵值除以路徑數，若分叉的路徑有相同的同義詞集，那麼我們會再將該節點的特徵值合併，最後再將所有的路徑計算平均。藉此衡量同義詞集在一個詞彙中的重要性。

以圖 4.2左半部的詞義編號 1 與詞義編號 2 作例子。路線量化的結果如表 4.1，詞義

編號 2 的路徑是 $\{eat\}-\{consume, ingest, takein, take, have\}-\{**root**\}$，因為詞義編號 2 ，只有走過這條路徑此，因此每個被走過的節點量化結果皆為 1。詞義編號 1 的路徑二條分別是 $\{eat\}-\{eat\}-\{consume, ingest, takein, take, have\}-\{**root**\}$ 和 $\{eat\}-\{consume, ingest, takein, take, have\}-\{**root**\}$，這時候我們會把 1 平均分擔於這二條路徑，因此這二條路徑上的每個量化的節點各是 0.5。接著我們合併同一個詞彙中相同的同義詞集，在詞義編號 1 的例子裡，二條路徑有 3 個同義詞集 $\{eat\}$、$\{consume, ingest, takein, take, have\}$ 和 $\{**root**\}$ 是重複的，因此我們把原本分擔於二條路徑上的 0.5 相加，使之成為 1，X 的部分視為 0，結果如表 4.2所示。最後，再計算每個同義詞集平均被經過的次數。以表 4.2詞義編號 1 與詞義編號 2 中的二個 $\{eat\}$ 為例，較抽象的 $\{eat\}$ 被經過次數只有一次，因此合併每個詞義量化後的結果再除以 1；較具體的 $\{eat\}$ 被經過的次數有二次，因此量化的結果相加後再除以 2。量化的結果如表 4.3所示。

表 4.1: 平均法範例 -路線量化

	詞義編號 1		詞義編號 2
同義詞集	路徑 1	路徑 2	路徑 1
$\{eat\}$	0.5	0.5	X
$\{eat\}$	0.5	X	1
$\{consume, ingest, takein, take, have\}$	0.5	0.5	1
$\{**root**\}$	0.5	0.5	1

表 4.2: 平均法範例 -合併同義詞集

	詞義編號 1	詞義編號 2
同義詞集	路徑 1	路徑 1
$\{eat\}$	1	X
$\{eat\}$	0.5	1
$\{consume, ingest, takein, take, have\}$	1	1
$\{**root**\}$	1	1

表 4.3: 平均法範例 -合併路徑量化

同義詞集	合併	量化結果
$\{eat\}$	1	1
$\{eat\}$	1.5	0.75

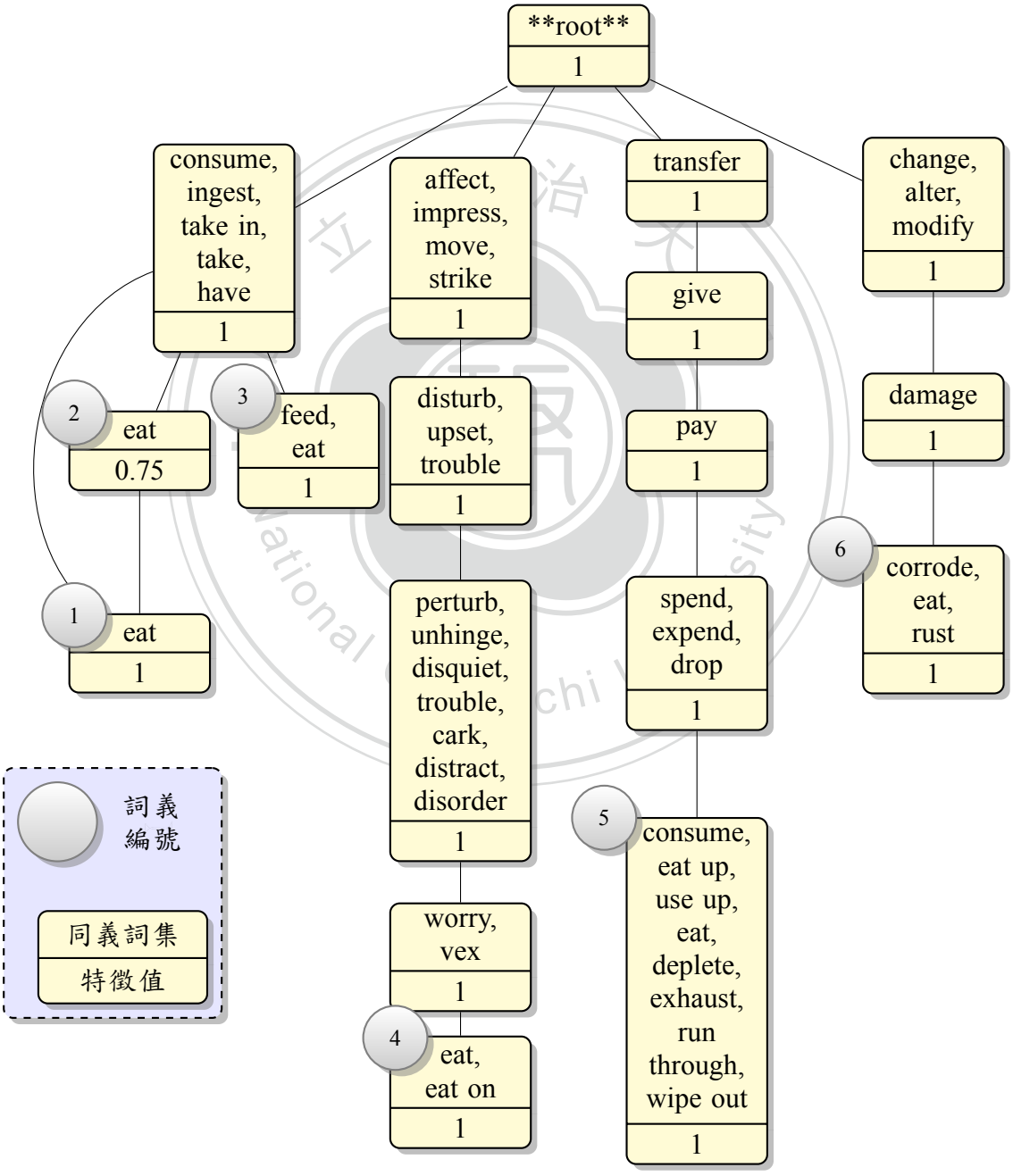

圖 4.2: 平均法

累計法

與平均法相比較，累計法是比較重視上位詞，越是上位的同義詞集越是抽象，也表示被經過的次數會越多次，被我們視為是較具代表性的同義詞集。

例子同樣是“eat”，表 4.4是一開始路線量化的結果與表 4.1的結果相同。下一步再將詞義相同的同義詞集合併如表 4.5與表 4.2結果相同。最後，將所有同義詞集量化結果加總合併並做正規化，正規化是除以詞彙的詞義數量，所以在這裡是除以 6，最後結果如表 4.6所示。

表 4.4: 累計法範例 -路線量化

	詞義編號 1		詞義編號 2
同義詞集	路徑 1	路徑 2	路徑 1
$\{eat\}$	0.5	0.5	X
$\{eat\}$	0.5	X	1
$\{consume, ingest, takein, take, have\}$	0.5	0.5	1
$\{**root**\}$	0.5	0.5	1

表 4.5: 累計法範例 -合併同義詞集

	詞義編號 1	詞義編號 2
同義詞集	路徑 1	路徑 1
$\{eat\}$	1	X
$\{eat\}$	0.5	1
$\{consume, ingest, takein, take, have\}$	1	1
$\{**root**\}$	1	1

表 4.6: 累計法範例 -計算詞頻量化

同義詞集	合併	平均
$\{eat\}$	1	0.2
$\{eat\}$	1.5	0.3

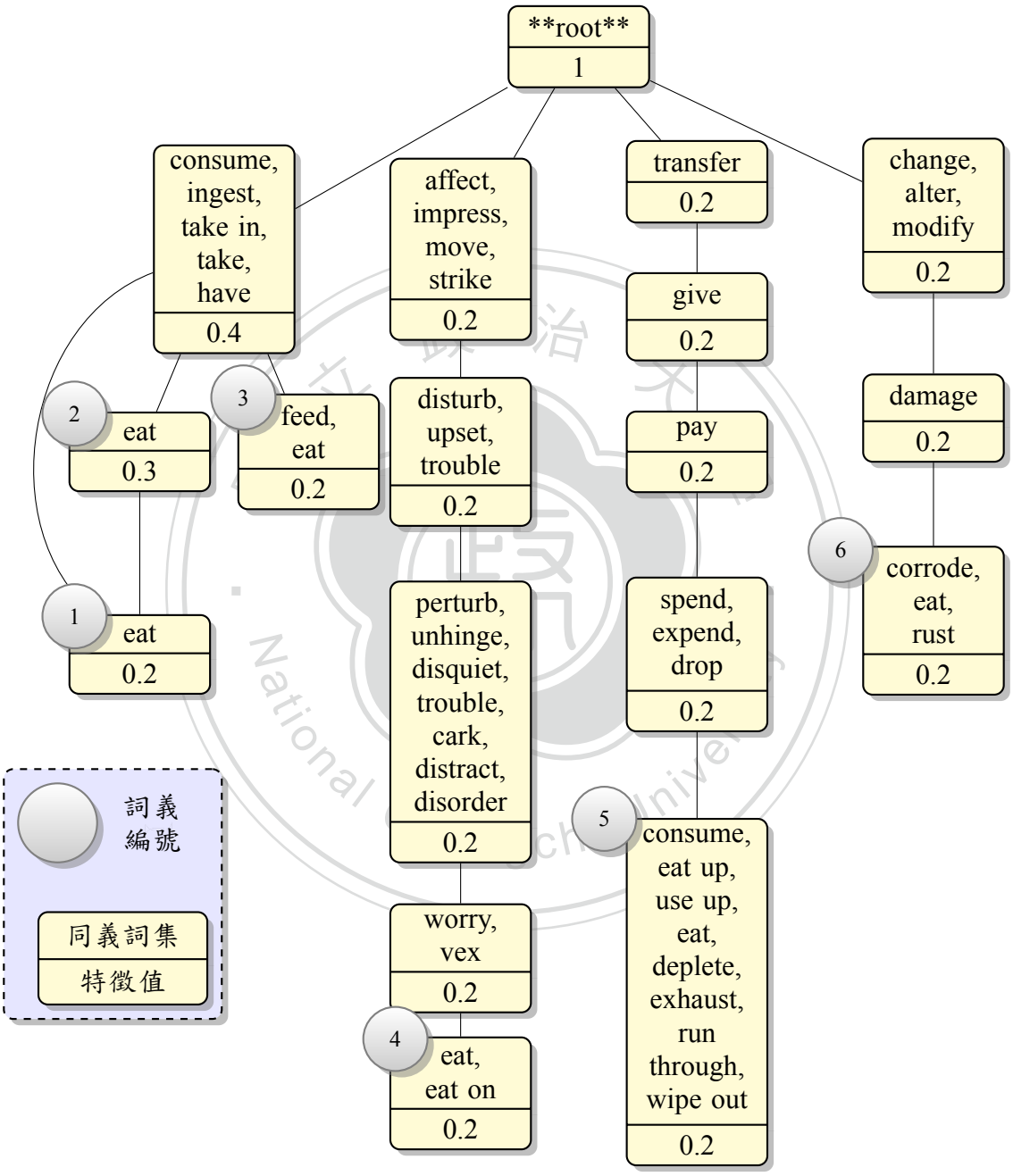
root
1
consume, ingest, take in, take, have
0.4
affect, impress, move, strike
0.2
transfer
0.2
change, alter, modify
0.2
give
0.2
damage
0.2
2
eat
0.3
3
feed, eat
0.2
disturb, upset, trouble
0.2
pay
0.2
6
corrode, eat, rust
0.2
1
eat
0.2
perturb, unhinge, disquiet, trouble, cark, distract, disorder
0.2
spend, expend, drop
0.2
5
consume, eat up, use up, eat, deplete, exhaust, run through, wipe out
0.2
worry, vex
0.2
4
eat, eat on
0.2
詞義編號
同義詞集
特徵值

圖 4.3: 累計量化

4.1.2 特徵加權

我們也對二種不同的特徵加權的方法各有不同詮釋：同樣是考慮透過 WordNet 查詢的詞彙，從查詢到的同義詞集到根節點間所有的同義詞集，詞義頻率考慮的面向是找出較常被使用到的詞義，這可以幫助我們處理一些語義歧義的問題；語義深度考慮到 WordNet 是階層式的架構，在 4.1.1節是以類似詞袋（bag of word）概念量化的方式中，多加入了一些階層式的概念。底下我們分別介紹二種加權方式：

詞義頻率

頻率是取自於 WordNet 所記載的頻率，在 WordNet 所記載的頻率不僅是針對字面的頻率，而是有考慮詞義的頻率。雖然在我們的研究中不做語義歧義處理，但利用這點我們可以稍辨識常用詞義為何。通常頻率越高表示越常被用到。

$$feature\ value \times \frac{\log_2(freq+2)}{\log_2(max\ freq+2)} \tag{4.1}$$

式 (4.1) 表示以詞義頻率加權的方法，其中 $feature\ value$ 表示 4.1.1節中的平均法或累計法，$freq$ 表示詞義頻率，$max\ freq$ 表示整個語料庫的頻率最大值，除以 $max\ freq$ 的目的是正規化，把值的範圍限定在 0 到 1 之間。式 (4.1) 中取 log 是因為我們不希望詞頻極值差距太大，加 2 為了做平滑化（smoothing），使得頻率取 log 之後至少是 1，而不是 0。

以動詞“eat”的 $\{eat\} - \{consume, ingest, takein, take, have\} - \{**root**\}$ 這條路徑為例，透過 WordNet 我們可以查詢到 $\{eat\}$ 這個同義詞集在 WordNet 的詞頻是 13，因此我們以 13 代表這條路徑在這個詞彙中的重要性。若代入式 (4.1) 中，則 $freq = 13$，結果如表 4.7所示。

表 4.7: 語義頻率量化

同義詞集	詞義頻率
$\{eat\}$	13
$\{consume, ingest, takein, take, have\}$	13
$\{**root**\}$	13

語義深度

語義的深度取自於該節點到根節點所經過的節點數量，也就是樹的深度。當到根節點的路徑不只一條時，則會計算平均深度長。當語義的深度越深，則該同義詞集被我們視為越不重要，反之越淺則越重要。因此我們將語義深度以倒數表示，再乘上同義詞集在 4.1.1節中量化的結果。

$$feature\ value \times \frac{1}{deep} \tag{4.2}$$

式 (4.2) 表示的是語義深度加權的方法，其中 $deep$ 表示是平均深度。同樣以 $\{eat\} - \{consume, ingest, takein, take, have\} - \{**root**\}$ 這條路徑為例，節點深度如表 4.8所示。

表 4.8: 語義深度量化

同義詞集	深度
$\{eat\}$	3
$\{consume, ingest, takein, take, have\}$	2
$\{**root**\}$	1

4.2 特徵選擇

一個詞彙可能多義，但我們不做語義歧義處理，而是將所有可能的語義及其不同層次的抽象化語義均納入特徵池 (feature pool)，所以特徵池的特徵數量會非常的多，特徵池表

示所有候選的同義詞集。由於特徵池非常龐大，而的特徵是從 WordNet 查詢來，所以這些特徵彼此存在著階層式的關係。因此我們設計了一套階層式特徵選擇的方法，而透過這樣階層式的選擇後，便可以找出具代表性的特徵，觀察與介系詞最相關連的語義層次為何，並瞭解我們研究的問題在何種語義層次上是可以被解決的。最後我們利用具代表性的特徵建構介系片語定位與介系詞推薦模型。

4.2.1 階層式選擇

階層式選擇方法可參考演算法 4.1所示，整個演算法的流程以圖 4.4表示。首先，我們會將所有案例中的三個中心詞動詞、名詞一和名詞二（介系詞片語定位問題是給定已知的介系詞；介系詞推薦問題的介系詞則是答案）透過 WordNet 查詢同義詞集及到根節點間的所有同義詞集，並且將這些同義詞集都放到特徵池裡。首先，先從特徵池選出所有案例的最底層的同義詞集將之視為初始的特徵，利用 4.1節的方法將特徵數值化。再透過 4.2.2節篩選條件過濾不具代表性的同義詞集，被留下的同義詞集會繼續參選下一個世代的階層式選擇；被過濾掉的同義詞集則會被拋棄，在下個世代階層式選擇中，會以上位詞來取代。被保留下的特徵與被新選上的同義詞集也就是被拋棄的同義詞集的上位詞，會再被數值化，然後重做階層式選擇，如此反覆直到終止條件成立。這裡我們所設定的終止條件是當特徵量過少即會停止。另一方面，由於同義詞集在越高層次特徵量會越少，因此透過這樣階層式的選擇，可達到縮減特徵的目的。

我們將以圖 4.5為範例解釋階層式選擇流程。圖 4.5是一個簡化的 WordNet 結構，每一個節點都代表一個同義詞集，其中 S_i 代表同義詞集的編號。表 4.9是簡化的語料庫，假設我們只有三筆案例，將三個中心詞簡化成一個，每一個中心詞透過 WordNet 查詢後，都至少有一條從該節點到根節點 S_4 的路徑。同義詞集量化的過程，我們以二元法做為範例。

演算法 4.1 階層式特徵選擇

輸入： 語料庫

輸出： N 個世代具代表性特徵

find_representational_features_for_each_generation(Corpus)

{將語料庫每個案例的動詞、名詞一和名詞二做特徵處理}

for all c ∈ Corpus **do**

{特徵數值化，參考 4.1節}

{fp 表示特徵池 (feature pool)}

$fp_{candidate}$ ← feature_processing(c)

end for

i = 0 { 初始世代 }

for all c ∈ Corpus **do**

{g_i 表示第 $_i$ 世代的具代表性特徵}

{將最底層的特徵選為初使代特徵}

g_i ← find_all_leaves_in_candidate_feature_pool(c)

end for

while (terminal_ conditions_ cannot_ be_satisfied) **do**

{針對該世代，建立每筆案例的特徵向量}

ft = build_feature_vector($fp_{candidate}$, g_i)

{特徵選擇，參考 4.2.2節的選擇條件}

fp_{keeped}, $fp_{abandoned}$ = select_feature(ft)

i = i + 1 { 進入下一個世代 }

{被保留的特徵，加入下一個世代}

g_i ← fp_{keeped}

{被拋棄的特徵，以上位詞取代，加入下一個世代}

for all f ∈ $fp_{abandoned}$ **do**

g_i ← find_hypenyms(f, fp_{keeped}, $fp_{abandoned}$)

end for

end while

{回傳所有世代具代表性特徵}

return g

演算法 4.2 尋找上位詞

find_hypenyms(f, fp$_{keeped}$, fp$_{abandoned}$)

{透過 WordNet 將 f 所有上位詞找出}
{ch 表示候選上位詞}
ch = find_canditante_hynernyms(f)
{檢查所有上位詞是否在 fp$_{keeped}$ 和 fp$_{abandoned}$}
{若不存在則視為是新的特徵}
for all h ∈ ch **do**
    if (h ∉ fp$_{keeped}$ ∨ h ∉ fp$_{abandoned}$) **then**
        hypernyms ← h
    end if
end for
{回傳所有新增上位詞}
return hypernyms

表 4.9: 階層式選擇範例: 簡化語料庫案例

案例編號	同義詞集至根節點路徑
VP1	$\{S_1\} - \{S_3\} - \{S_4\}$
VP2	$\{S_2\} - \{S_3\} - \{S_4\}$
VP3	$\{S_5\} - \{S_5\} - \{S_7\} - \{S_4\}$

第 0 個世代被視為是初始的世代，首先我們會挑出所有案例最底層的同義詞集並且放到特徵池，第一次被選上的同義詞集有 S_1、S_2 和 S_5，再以二元法量化，結果如表 4.10。透過 4.2.2節特徵篩選條件過濾後，假設 S_1 與 S_5 是這個世代被選出需要淘汰的特徵，那麼我們就會挑選 S_1 的上位詞 S_3 和 S_5 的上位詞 S_6 補上。此時 S_3 同時也是 S_2 的上位詞，因此對於 VP2 的案例而言，VP2 多出一個新的參選特徵。接著再重新量化新選出的特徵參選下一個世代選擇，結果如表 4.11。接著在第二代參選中，如果我們再淘汰 S_3，就會再以 S_4 補上如表 4.12所示，那對於 VP2 這個案例 S_3 就會被視為是不存在。在第三代的參選中淘汰 S_2 則應補上 S_4，但 S_4 已經存在，所以這個世代特徵只會減少不會新增，如表 4.13。

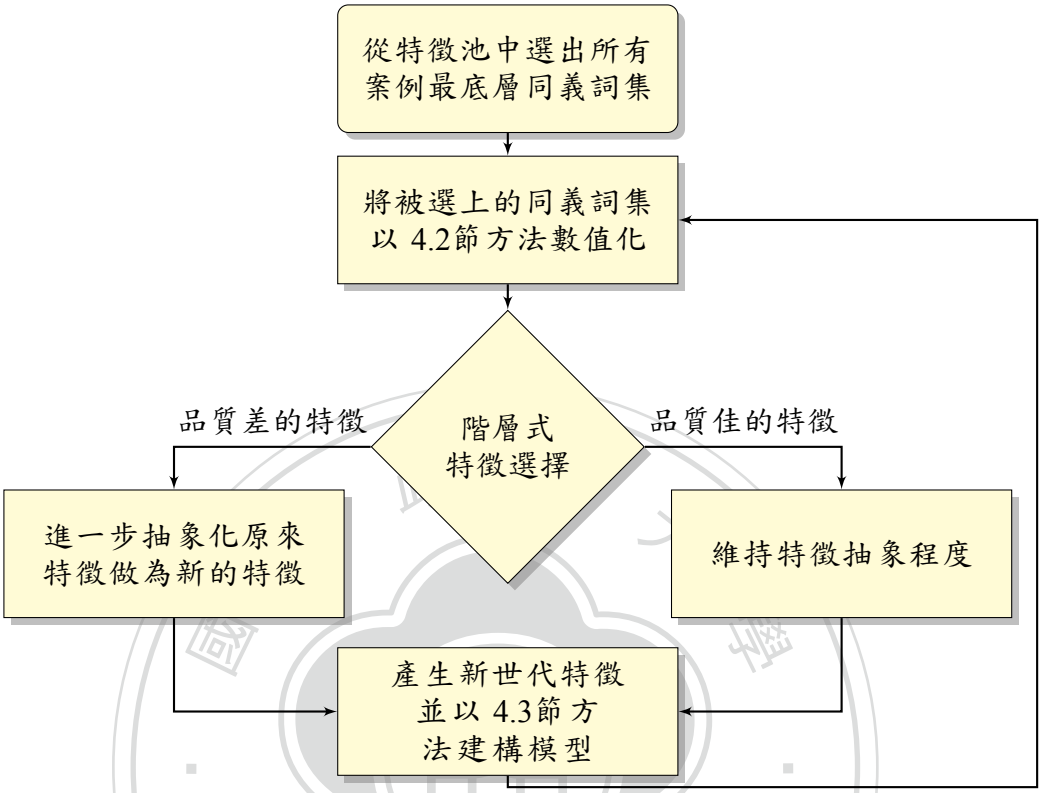

圖 4.4: 階層式選擇流程圖

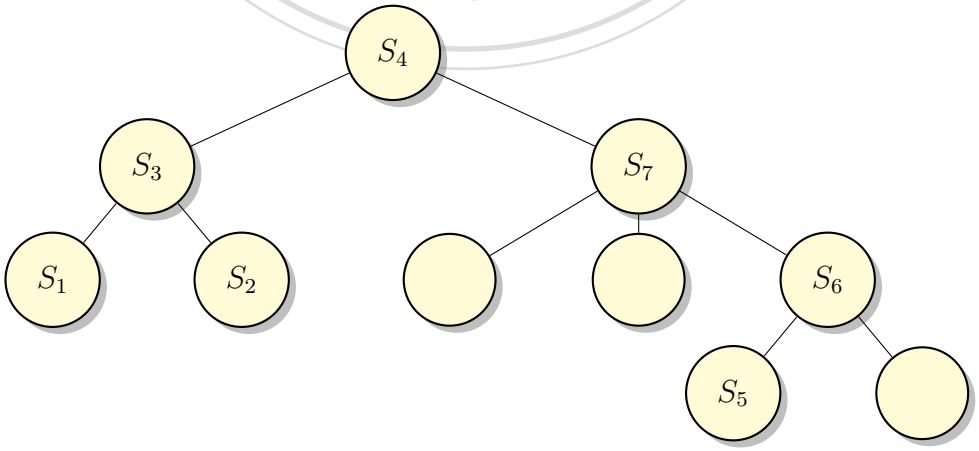

圖 4.5: 階層式選擇範例: 簡化的 WordNet 結構

表 4.10: 階層式選擇範例: 第 0 世代

編號	S_1	S_2	S_5
VP1	1	0	0
VP2	0	1	0
VP3	0	0	1

表 4.11: 階層式選擇範例: 第 1 世代

編號	S_3	S_2	S_6
VP1	1	0	0
VP2	1	1	0
VP3	0	0	1

表 4.12: 階層式選擇範例: 第 2 世代

編號	S_4	S_2	S_6
VP1	1	0	0
VP2	1	1	0
VP3	1	0	1

表 4.13: 階層式選擇範例: 第 3 世代

編號	S_4	S_6
VP1	1	0
VP2	1	0
VP3	1	1

4.2.2 篩選條件

篩選條件的方式可以分為三種：（一）以計算該特徵使用的頻率並過濾低頻的部分，（二）計算熵比例（gain ratio）並且過濾熵比例等於 0 的特徵，（三）計算共現（collocation）同義詞集。被過濾的同義詞集表示其抽象化的程度不夠因此不具代表性，所以我們將其再度抽象化並以上位詞取代。

詞頻

在我們特徵選擇方法中，假設越抽象化的同義詞集應該是越重要且越常被使用。因此我們統計每個世代的特徵在案例中出現在的次數，將該回合的特徵的頻率依多寡排列並設二個門檻值，當特徵頻率低於 10 或門檻值時就會被過濾。

熵比例

熵比例是我們用來計算該特徵代表性的方法之一，以 $\{entity\}$ 為例子，$\{entity\}$ 這個特徵是名詞的同義詞集最抽象化的概念，所有的名詞的根節點一定是 $\{entity\}$，因此當這個同義詞集被選上做為特徵後，語料庫中的所有案例都會有 $\{entity\}$ 這個特徵，雖然它是詞頻最高的特徵，但因為它高到每個案例都有，因此這個特徵的重要性反而大大降低，所以我們透過熵比例把計算為 0 的特徵過濾。

共現同義詞集

除了單個同義詞集的頻率外，我們也統計了共現同義詞集頻率，也就是在這回合作的同義詞集中，任二個同義詞集一起出現的頻率。由於我們是將動詞、名詞一與名詞二的同義詞集混合在一起組合成最後的特徵，所以再我們統計共現同義詞集時，我們定了一些規則，將較不合理的狀況去除。不合理的組合如下：

- 動詞 + 動詞
- 名詞一 + 名詞一
- 名詞二 + 名詞二

另外，我們大膽假設了一些情況，如果名詞二與名詞一或名詞二與動詞的關聯性較強，那麼我們也把動詞 + 名詞一的組合刪除。

- 動詞 + 名詞一

透過上面的規則，我們希望可以再減少一些不具代性的特徵。

4.3 模型建構

特徵選擇後，下一步是使用機器學習的演算法訓練模型，用以決策類別。模型的建構可分為三個階段：第一階段所建構的模型我們視為是基準模型，也就是模型決策的結果會被我們視為是基線。第二階段，則是利用較熱門的機器學習演算法建構模型，我們挑選的演算法有 SVM（Support Vector Machines）、C4.5 和 Naïve Bayes。第三階段是根據第二階段所產生的模型，再建構一個高階模型（meta learner），高階模型的演算法同樣是採用 SVM、C4.5 和 Naïve Bayes。

4.3.1 基準模型建構

我們針對介系詞片語定位問題設計了一套類似於 Naïve Bayes 的演算法，在給定特定的介系詞之下，我們考慮的不再是單一特徵而是動詞、名詞一與名詞二的同義詞集組合。在考慮同義詞集組合的情況下，我們有機會知道何種組合是較有機會可以解決介系詞定位問題。

考慮動詞片語的四個中心詞，若限定在特定介系詞 P 的情況下，則以 $\vec{W} = (V, N1, N2)$ 表示一個動詞片語，其中 V、$N1$、$N2$ 分別表示動詞、名詞一以及名詞二。若我們想要解決的介系詞片語定位問題 D 是定位修飾動詞或修飾名詞一，則 $D = \{VPP, NPP\}$ 。更進一步，我們可計算修飾動詞的機率 $Pr = (D = VPP|\vec{W})$ 和修飾名詞的機率 $Pr = (D = NPP|\vec{W})$。透過 WordNet 查詢後，動詞的同義詞集表示為 $V = \{s_{v_1}, s_{v_2}, \cdots, s_{v_i}\}$，名詞一表示為 $N1 = \{s_{n1_1}, s_{n1_2}, \cdots, s_{n1_j}\}$，名詞二表示為 $N2 = \{s_{n2_1}, s_{n2_2}, \cdots, s_{n2_k}\}$。以 $\vec{S} = \{s_{v_i}, s_{n1_j}, s_{n2_k}\}$ 代表 $\vec{W}$ 一個可能的同義詞集組合特

徵，以 $R(\vec{S})$ 表示所有可能的同義詞集組合特徵。

因此我們可以將模型表示成式 (4.3)，式 (4.4) 我們假設 D 與 $\vec{W}$ 在已知 $\vec{S}$ 的情形下是條件獨立，$\vec{S}$ 展開後得到式 (4.5)，若再假設式 (4.5) 個別詞彙在語境中所負擔的詞義角色與其它詞彙無關，則最後可以得到式 (4.6)。

$$Pr(D|\vec{W}) = \sum_{\vec{S}\in R(\vec{S})} Pr(D,\vec{S}|\vec{W}) = \sum_{\vec{S}\in R(\vec{S})} Pr(\vec{S}|\vec{W}) \times Pr(\vec{D}|\vec{W},\vec{S}) \tag{4.3}$$

$$= \sum_{\vec{S}\in R(\vec{S})} Pr(\vec{S}|\vec{W}) \times Pr(\vec{D}|\vec{S}) \tag{4.4}$$

$$= \sum_{\vec{S}\in R(\vec{S})} Pr(s_{v_i}, s_{n1_j}, s_{n2_k}|V, N1, N2) \times Pr(D|s_{v_i}, s_{n1_j}, s_{n2_k}) \tag{4.5}$$

$$= \sum_{\vec{S}\in R(\vec{S})} Pr(s_{v_i}|V) \times P(s_{n1_i}|N1) \times Pr(s_{n2_i}|N2) \times Pr(D|s_{vi}, s_{n1_j}, s_{n2_k}) \tag{4.6}$$

在給定特定介系詞的情況下，4.3.2節所介紹的 Naïve Bayes，可表示圖 4.6的結構，其中 S_i 表示每一個世代的同義詞集。如果類比於 Naïve Bayes，可以將基準模型的結構表示成如圖 4.7。

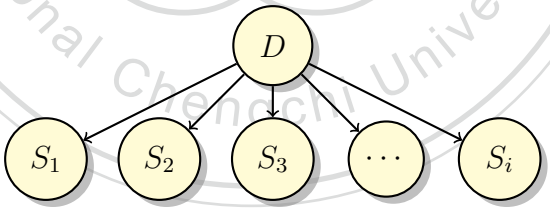

圖 4.6: Naïve Bayes 結構

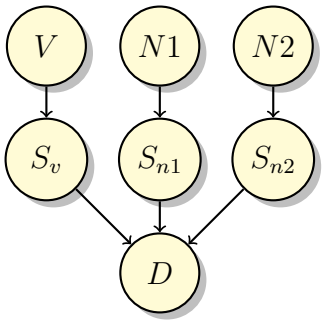

圖 4.7: 基準模型結構

根據式 (4.6)，最後推得的結果中 $Pr(s_{v_i}|V)$、$P(s_{n1_j}|N1)$ 和 $Pr(s_{n2_k}|N2)$ 項，以動詞為例，可經由式 (4.7) 而得，其中 $WN(S)$ 表示一個詞彙的其中一個同義詞集經由 WordNet 查詢得到的頻率。由於某些詞義詞頻可能為零，因此我們使用 *Laplace estimator* 概念做平滑化（smooth），而式 (4.7) 中 $|V|$ 表示 V 的詞義個數。以“eat”為例，如表 4.14所示。

$$Pr(s_{v_i}|V) = \frac{WN(s_{v_i}) + 1}{\left(\sum_{s_{v_m} \in V} WN(S_{v_m})\right) + |V|} \tag{4.7}$$

$Pr(D|s_{v_i}, s_{n1_j}, s_{n2_k})$ 項，則是在訓練時，經由統計而得。以式 (4.8) 表示。

$$Pr(D|s_{v_i}, s_{n1_j}, s_{n2_k}) = \frac{|(s_{v_i}, s_{n1_j}, s_{n2_k}, D)|}{|(s_{v_i}, s_{n1_j}, s_{n2_k})|} \tag{4.8}$$

表 4.14: 以“eat”為例計算 $Pr(s_{vi}|V)$

同義編號	頻率	$Pr(s_{vi}|V)$
1	$61+1$	$(61+1)/84$
2	$13+1$	$(13+1)/84$
3	$4+1$	$(4+1)/84$
4	$0+1$	$(0+1)/84$
5	$0+1$	$(0+1)/84$
6	$0+1$	$(0+1)/84$
總合	84	1

4.3.2 傳統模型建構

傳統模型表示我們使用現在大家常用的熱門演算法。我們共選了三種 SVM、C4.5 和 Naïve Bayes 演算法，其中 SVM 採用的是 Libsvm-3.11[1]版本，而後二者 C4.5 與 Naïve Bayes 使用的工具是 Weka3-6-6[2]版本。

SVM 是目前受歡迎的演算法之一。它是一種線性的分類器，但可以使用核心方

[1]http://www.csie.ntu.edu.tw/ cjlin/libsvm/
[2]http://www.cs.waikato.ac.nz/ml/weka/

法（kernel method），將所有資料的點對應到其它維度空間，再用決策邊界（decision boundary）切割二類資料。我們選用的核心方法為 RBF，需要調整參數 γ 和 $cost$ 以使參數最佳化。使用 grid search 演算法調整參數，利用的工具是 libsvm 的 grid.py。將 x 軸對應到 $cost$ 參數範圍設為 -5 到 11，步數為 2。而 y 軸對映到 γ 參數範圍設為 -11 到 3，y 軸步數皆設為 2。再將 x 軸與 y 軸值代入涵數 $f(n) = 2^n$，並測試 $f(x)$ 與 $f(y)$ 分類的效果。

決策樹（Decision Tree）也是機器學習裡常用的演算法之一，相關的演算法有 ID3、C4.5 和 C5.0 等。我們採用的演算法是 C4.5。C4.5 的演算法需要調整二個參數：每個節點至少包含的案例數和 *confidence factor*。參數最佳化的處理是使用 Weka。將節點數參數設為 x，範圍設為 5 到 50，步數為 5。*confidence factor* 參數設為 y，範圍設為 0.05 到 0.45，步數為 0.05。最後直接將 x 與 y 值代入演算法測試分類效果。

Naïve Bayes 是一個簡單的機率分類器。它假設了分類器所使用的每一個特徵都是獨立不相關的，但在現實的情況中不少情況是特徵之間是互相依賴而非完全獨立的。在 4.3.1節的基準模型建構的基本假設也與 Naïve Bayes 是一樣的，不同之處在於我們同時考慮的是動詞、名詞一與名詞二等，而 Naïve Bayes 考慮的只有單一一個同義詞集，可參考圖 4.6的結構。

4.3.3 高階模型建構

在現在許多成功的分類器背後都不單使用一個分類器，而是採用多個分類器整合而成。這類似於我們將每個模型都視為是一位專家，讓每位專家都發表自己的看法，再經由機器學習演算法統整各個專家的看法整合成高階模型。這樣做的好處，可以讓我們的每個不同專長的模型有機會發揮自己的效用。

我們同樣使用第 4.3.2節所提到三種演算法：SVM、C4.5 和 Naïve Bayes 來建構高階模型。高階模型建構流程圖如 4.8，我們將所有語料分成三份：訓練語料、驗證語料、

測試語料，利用訓練語料建立傳統模型，再決策驗證語料的答案，將決策的答案當作高階模型的訓練資料，用以訓練高階模型，最後把測試語料當作最終上線的測試資料並利用訓練好的高階模型來做最後決策。

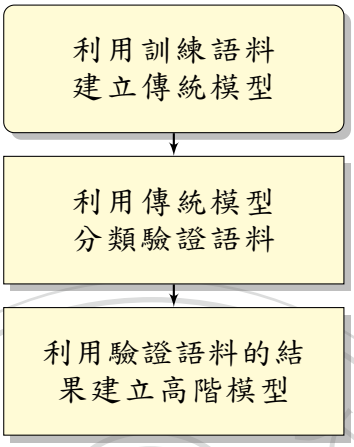

圖 4.8: 高階模型建構

第 5 章　實驗

本章將分析第 4 章建立模型的成效。首先，我們會介紹實驗設計，接著是衡量模型成效的方法，衡量的方法包含了：精準度（Accuracy）、精確率（Precision）、召回率（recall）和綜合評量（F_1-Measure）。此外，因為 RRR 和 PTB3 語料庫的語料量較少，所以也會評估個數對於效果（performance）的影響。最後我們將分析介系詞片定位與介系詞推薦的實驗成果。

5.1　實驗設計

實驗設計包含基準模型的特徵挑選、傳統模型的實驗組合設計與研究方法的比較和高階模型挑選傳統模型。

5.1.1　基準模型實驗

在我們研究方法裡，我們相信介系詞片語定位與介系詞推薦二個問題，只要動詞、名詞一與名詞二在很高層的語義層次，就可以使模型成功的分類大多數的案例。因此我們設計基準模型，利用較高層次的同義詞集做為建構模型的特徵。

我們挑選名詞一與名詞二的同義詞集是從 WordNet 名詞的根節點往下數第三層的同義詞集，共有 22 個，如表 5.1所示。

表 5.1: 基準模型所使用得名詞同義詞集

同義詞集
$\{thing\}$
$\{object, physicalobject\}$
$\{causalagent, cause, causalagency\}$
$\{matter\}$
$\{process, physicalprocess\}$
$\{substance\}$
$\{psychologicalfeature\}$
$\{attribute\}$
$\{group, grouping\}$
$\{relation\}$
$\{communication\}$
$\{measure, quantity, amount\}$
$\{otherworld\}$
$\{set\}$
$\{change\}$
$\{freshener\}$
$\{horror\}$
$\{jimdandy, jimhickey, crackerjack\}$
$\{pacifier\}$
$\{securityblanket\}$
$\{stinker\}$
$\{whacker, whopper\}$

動詞同義詞集不同於名詞是它的樹狀結構深度比較淺。在不包含根節點的動詞同義詞集結構中，最高層的同義詞集數量非常的多，因此我們改以挑選最高層同義詞集的種類，這可以使我們的特徵大大地減少。

5.1.2 傳統模型實驗

在介系詞片語定位問題裡，我們特別將傳統模型細分為單一模型與混合資料模型。二者的差別在於單一模型是給定特定介系詞建構模型，混合資料模型則是將 3.4節所挑選的 6 個具挑戰性介系詞混合一起建構模型，也就是說混合資料模型比起單一模型共多了 6 個介系詞特徵，多出的 6 個介系詞特徵以 0 與 1 表示，1 表示使用的介系詞，反之為 0。

表 5.2是我們根據 4.1.1節、 4.1.2節與 4.2.2節共設計 12 組不同條件組合的實驗，其中第 11 組與第 12 組實驗特別考慮了共現同義詞集的組合。

表 5.2: 條件組合

條件組合	詞頻	共現	量化方式	加權
1	> 中位數	X	平均法	X
2	>200	X	平均法	X
3	> 中位數	X	平均法	語義深度
4	> 中位數	X	平均法	詞義頻率
5	> 中位數	X	平均法	語義深度 × 詞義頻率
6	> 中位數	X	二元法	X
7	> 中位數	X	累計法	X
8	X	>100	平均法	X
9	X	>200	平均法	X
10	X	>200	平均法	詞義頻率
11†	X	>200	平均法	X
12‡	X	>200	平均法	X

† 不考慮：動詞＋動詞, 名詞一＋名詞一, 名詞二＋名詞二
‡ 不考慮：動詞＋動詞, 名詞一＋名詞一, 名詞二名詞二, 名詞一＋動詞

這 12 組實驗中的條件組合 1 到條件組合 7，由於特徵銳減速度較慢，因此特徵選擇時，共會跑 9 個世代。條件組合 8 至 12 則因為特徵銳減速度較快，因此只跑 7 個世代。初始的幾個世代中，因為考慮特徵量較龐大的關係，所以在建構模型時，單一模型

表 5.3: 介系詞片語定位實驗比較

比較對象	考慮特徵
最大熵值法	動詞、名詞一、介系詞與名詞二
Stanford 剖析器	上下文全句

會從第 3 個世代開始建構模型，而混合資料模型則從第 4 個世代開始建構模型。

在我們的研究裡，除了我們自己設定的基線比較外，我們另外也與其它學者的研究方法做比較。如表 5.3所示。在同樣都只考慮四個中心詞的情況下，我們的方法會與 Ratnaparkhi 等人 [20] 的最大熵值法做比較。另外，我們也與考慮上下文全句的研究做比較，比較對象是 Klein 和 Manning[12] 的 Stanford 剖析器。

5.1.3 高階模型實驗

我們設定了不同的條件，挑選品質較佳的傳統模型做為基礎，用以建構高階的模型，條件的設定如表 5.4所示。條件 1，是將所有傳統模型結果都用來訓練高階模型；條件 2，表示先計算所有模型的平均精準度，只挑選精準度高於平均的模型；條件 3，大致同條件 2，但是我們只選擇傳統模型是演算法是 SVM 的模型。事實上，演算法是 SVM 的傳統模型，在我們的實驗中成果中，幾乎是表現最好的，細節可參考 5.3。

表 5.4: 挑選傳統模型條件

條件	挑選條件
1	使用全部分類器
2	計算所有分類器精準度，並選大於平均者
3	同條件 2，但只選取使 SVM 訓練的分類器

5.2 實驗評量

我們以精準度做為評量模型總體的方式。假設模型可以答對的案例數為 T，而語料庫的案例數為 N，則精準度如式 (5.1)。

$$Accuracy = \frac{T}{N} \tag{5.1}$$

由於我們某些實驗語料量較小，因此我們還希望評量語料量對效果的影響，測量方法如式 (5.2) 所示。這裡我們將信心度（confidence）設為 80%，而 z 可以透過信心度查表得知為 1.28。

$$Performance = (Accuracy + \frac{z^2}{2N} \pm z\sqrt{\frac{Accuracy}{N} - \frac{Accuracy^2}{N} + \frac{z^2}{4N^2}})/(1 + \frac{z^2}{N}) \tag{5.2}$$

另外，我們以準確率、召回率和綜合評量評量各個類別的決策效果。若是介系詞片語定位的問題，那麼決策的類別粗分為答對與答錯。若是評量介系詞推薦問題，類別是各個介系詞。

精確率表示模型對於該類別分類的正確性。假設某一模型在語料庫中，決策某一類別數量為 E，而我們模型可以對該類別正確做決策的數量為 D，則準確率可以描述如式 (5.3)。

$$Precision = \frac{D}{E} \tag{5.3}$$

召回率表示的是我們的模型對於該類別的信心程度。假設某一類別有 G 個案例數，那麼召回率定義如式 (5.4)

$$Recall = \frac{D}{G} \tag{5.4}$$

實務上，我們希望精確率和召回率都很高。但往往結果是高精確率與低召回率或高召回率低精確率。在這樣的情況下，需要一個綜合評合的指標，因此我們採用綜合評量（F_1-$measure$），如式 (5.5)。

$$F_1\text{-}measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{5.5}$$

5.3 實驗分析：介系詞片語定位

本節將會探討 RRR 語料庫與 PTB3 語料庫實驗結果的分析，討論分析的內容包含：在不同條件組合對各個介系詞的影響、階層式特徵選擇與高層語義分析、不同條件下建構高階模型的結果、評比我們研究的方法與參考文獻中的方法。最後一項比較參考文獻的方法中，我們選擇的比較方法是最大熵值法與 Stanford 剖析器，前者方法上與我們相同的是都是由四個中心詞做發點，後者則是考慮上下文。

5.3.1 不同條件組合之分析

我們依據表 5.2所列出的不同條件組合，共設計出 6 大組的實驗組合。底下將以 RRR 語料的訓練語料所訓練而成的模型做說明，並用 RRR 的驗證語料當作內部測試做為實驗分析。我們在 PTB3 的語料庫也做了同樣的實驗，但實驗的結果是差不多的。6 個實驗組合裡，又包含 6 個不同介系詞與 1 個混合資料實驗。在比較的同時，我們也以每個實驗組合的第一個條件作為基線，並對這條基線利用式 (5.2) 計算可能的最大與最小誤差範圍。圖 5.1、圖 5.2、圖 5.3、圖 5.4、圖 5.5和圖 5.6中，每一個介系詞的精準度都有不同的誤差範圍，其中“with”和“from”的誤差範圍是最大的，因為它們的個數是比較少的；而混合資料的誤差範圍是最小，因為是數量最大的。

表 5.5、表 5.6、表 5.7、表 5.8、表 5.9和表 5.10是實驗組合的比較條件。實驗結果如圖 5.1、圖 5.2、圖 5.3、圖 5.4、圖 5.5和圖 5.6所示，這些圖下方 x 軸表示的是世代，上方表示不同的介系詞和混合資料，左方 y 軸表示精準度，以百分比表示，右方表示使用的演算法。圖 5.7、圖 5.8、圖 5.9、圖 5.10、圖 5.11和圖 5.12是與實驗組合相對應的

特徵數量圖，下方 x 軸表示世代，圖中僅顯示第 3 到第 9 個世代，左方 y 軸表示特徵數量，上方表示該圖的介系詞。

若比較三種不同演算法，可以看到這三種演算法大致上都是 SVM 的效果比較好，C4.5 次之，再次之為 Naïve Bayes。猜想 C4.5 分類較差的原因可能是因為 C4.5 在建構結構決策樹時，實際上，所用的特徵量少。以圖 5.1的 C4.5 介系詞“for”的世代 3 為例，雖然在世代 3 實驗組合 1 與 2 的特徵量分別各有 482 與 203 個，但實際在建構決策樹時，只用了與 47 與 8 個，這可能是造成 C4.5 表現不佳的原因。Naïve Bayes 則可能與演算法本身的假設有關，演算法的假設每個特徵都是獨立不相關。

細看三種演算法在每個世代的趨勢，可以發現幾乎都是隨著世代數的遞增而精準度隨著遞減。而且大多數實驗都是到某個世代，精準度會突然較大幅度的遞減，這說明大多數的實驗透過階層式特徵選擇時，較具代表性的同義詞集在前幾個世代就可以被找出，在後幾個世代的同義詞集概念太過抽象化，使得分類效果反而沒有這麼的好。

底下我們將比較不同的條件對於介系詞的影響，雖然整體而言並沒有明顯趨勢，但我們可以看出確實每個介系詞都有自己的偏好。

實驗組合一：比較不同門檻值之詞義詞頻

在表 5.5的條件設定下，比較不同門檻值的詞頻，結果如圖 5.1，每個世代的特徵量參考圖 5.7。可以看到大致上都是條件 1 的比較條件 2 好，這說明挑選詞頻的門檻值不應該設太高。若我們將條件 1 的每個世代被使用到的特徵依使用次數排序，也就是把非 0 的特徵值都視被使用的特徵。可以發現每一個世代的曲線都與 Zipf's law 的曲線相同，這表示常被用到的語義或許就只有這些，所以當門檻值設太高時，很容易再一開始就使重要的特徵被淘汰。

表 5.5: 實驗組合一

條件組合	詞頻	共現	量化方式	加權
1	> 中位數	X	平均法	X
2	>200	X	平均法	X

實驗組合二：比較不同門檻值之共現同義詞集

表 5.6條件設定下，比較不同門檻值的共現同義詞集，結果如圖 5.2所示，各個世代的特徵數量如圖 5.8。實驗組合二的精準度平均來講都比實驗組合一低，原因是因為考慮共現同義詞集與考慮詞頻相比，考慮共現同義詞集是比較嚴苛條件，所以每次篩選剩下的特徵會較少，導致精準度下降。在嚴苛的條件下，設限較高的門檻，可以看出效果是有限的，因此圖 5.2中大部分的介系詞的表現都差不多。其中高門檻值對於特徵數量較多的“in”是有影響，但 C4.5 結果不明顯，原因可能是 C4.5 建樹時選用特徵較少，而其它二個演算法效果比較明顯的。介系詞“from”也是與其它介系詞表現的比較不一樣，也可看出明顯的差距，低門檻值的結果使“from”與分布線差不多。

表 5.6: 實驗組合二

條件組合	詞頻	共現	量化方式	加權
8	X	>100	平均法	X
9	X	>200	平均法	X

實驗組合三：比較不同加權方法

表 5.7條件設定下，比較不同加權方法，結果如圖 5.3所示，每個世代的特徵量如圖 5.9所示。圖 5.3中，不同的加權的方法結果並沒有差太多。除了混合資料外條件 1 與條件 3 的結果幾乎是重疊且每個世代特徵數量幾乎是一樣多；條件 4 與條件 5 同樣也幾乎是重疊，且每個世代特徵量幾乎也是一樣多。這說明考慮語義深度是沒有什麼用途，可能因為多考慮的這個條件它的功是能與我們所設計的階層式特徵選擇方法功能相似，

因此沒有發生太大做用。

表 5.7: 實驗組合三

條件組合	詞頻	共現	量化方式	加權
1	> 中位數	X	平均法	X
3	> 中位數	X	平均法	語義深度
4	> 中位數	X	平均法	詞義頻率
5	> 中位數	X	平均法	語義深度 × 詞義頻率

實驗組合四：比較不同量化方法

表 5.8條件設定下，比較不同量化方法，結果如圖 5.4所示，每個世代的特徵量如圖 5.10所示。圖 5.4可以看到三種量化的方法整體並沒有明顯優劣趨勢。

表 5.8: 實驗組合四

條件組合	詞頻	共現	量化方式	加權
1	> 中位數	X	平均法	X
6	> 中位數	X	二元法	X
7	> 中位數	X	累計法	X

實驗組合五：比較不同共現同義詞集組合

表 5.9條件設定下，比較不同共現同義詞集組合，結果如圖 5.5所示，每個世代的特徵量如圖 5.11所示。圖 5.9中，條件 11 與條件 12 的幾乎是完全重合的，雖然條件 12 看起比條件 11 嚴苛，但是因為在考慮共現同義詞集組合中，即使某個同義詞集與其它的同義詞集沒有符合條件到達門檻，但只要其中一個同義詞集與其它同義詞集組合符合條件到達門檻值就會被保留，這可能是造成條件 11 與條件 12 結果相同的原因。除了介系詞“with”、“to”和混合資料外，其它介系詞在不考慮任何同義詞集組條件情況下，比考慮特定同義詞集組合條件 11 與條件 12 是效果好，這可能是因為在考慮較嚴苛的共現同義詞集之下，考慮組合條件使得特徵更為稀少，因此差異不大。而介系詞 with 則是因為

同義詞集特徵量過少，所以於精準度與分布線是差不多。在語料量比較大的情況下，介系詞“to”、“with”與混合資料顯示多考慮共現同義詞集組合是有用的，雖然精準度差不多，但是在考慮特定組合條件這種較嚴苛的條件下，不僅可減少特徵量，且仍有不錯的表現。

表 5.9: 實驗組合五

條件組合	詞頻	共現	量化方式	加權
8	X	>100	平均法	X
11†	X	>200	平均法	X
12‡	X	>200	平均法	X

† 不考慮：動詞＋動詞, 名詞一＋名詞一, 名詞二＋名詞二
‡ 不考慮：動詞＋動詞, 名詞一＋名詞一, 名詞二＋名詞二, 名詞一＋動詞

實驗組合六：基於詞義頻率之詞頻與共現同義詞集組合之比較

表 5.10條件設定下，是考慮在詞義頻率加權之下比較詞頻和共現同義詞集組合效果，結果如圖 5.6所示，每個世代的特徵量如圖 5.12所示。介系詞“for”、“on”、“to”除了 Naïve Bayes 考慮詞頻是明顯的精準度比較好，而“with”和“from”均是考慮詞頻較好外，其它二者演算法的表現差不多，雖然表現差不多，但由於特徵量較少，因此可以看出考慮共現同義詞集組合是有用的。觀察語料數量較多的混合資料，SVM 也是考慮共現同義詞集的效果較好，但 Naïve Bayes 模型在這二個條件下的表現差不多。在語料數量較少的介系詞“with“和“from”則是考慮詞頻的效果較好，因為考慮共現同義詞集使特徵量較少，導致精準度與分布線差不多。

表 5.10: 實驗組合六

條件組合	詞頻	共現	量化方式	加權
4	＞中位數	X	平均法	詞義頻率
10	X	>200	平均法	詞義頻率

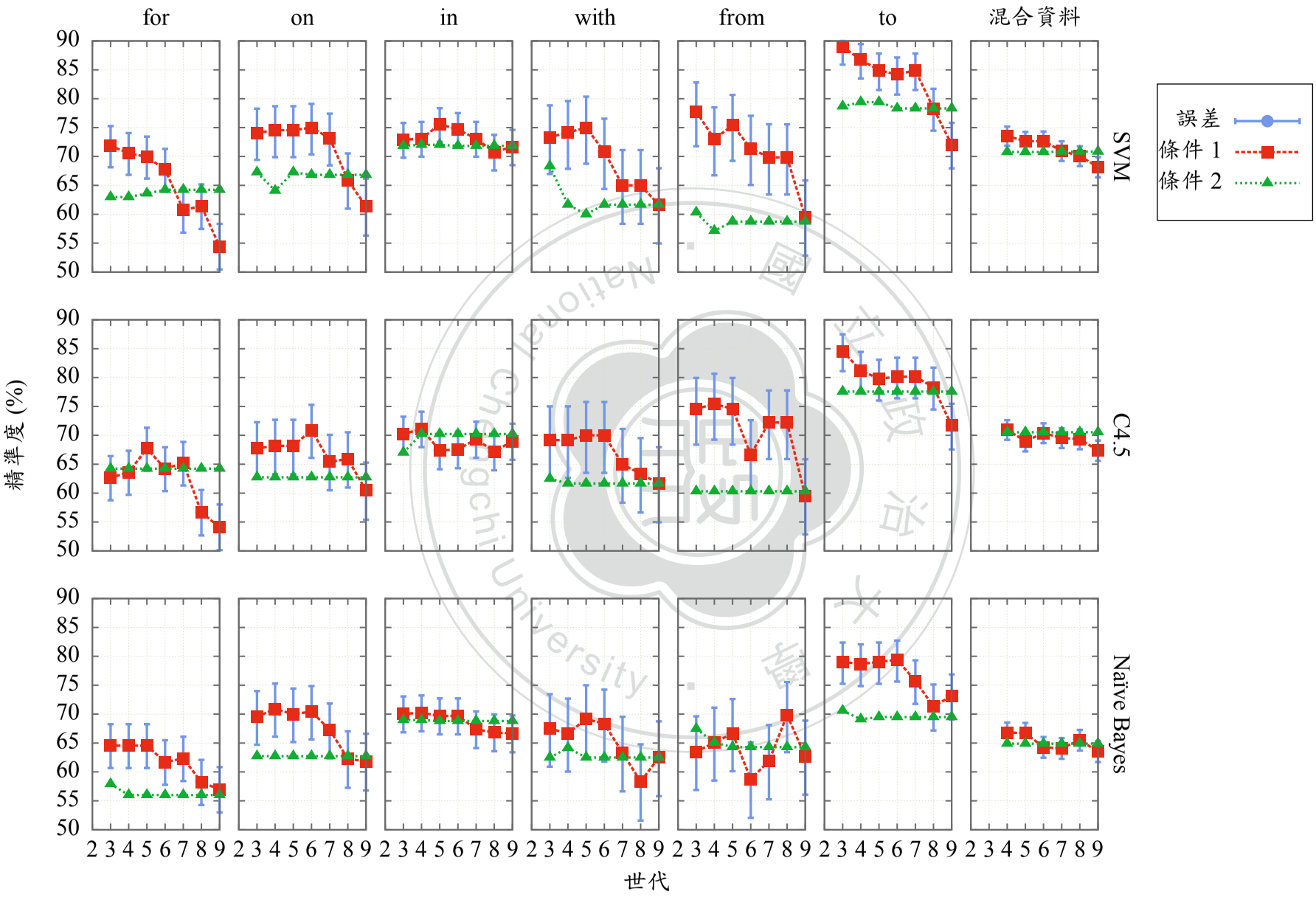

圖 5.1: 比較不同門檻值之詞義詞頻

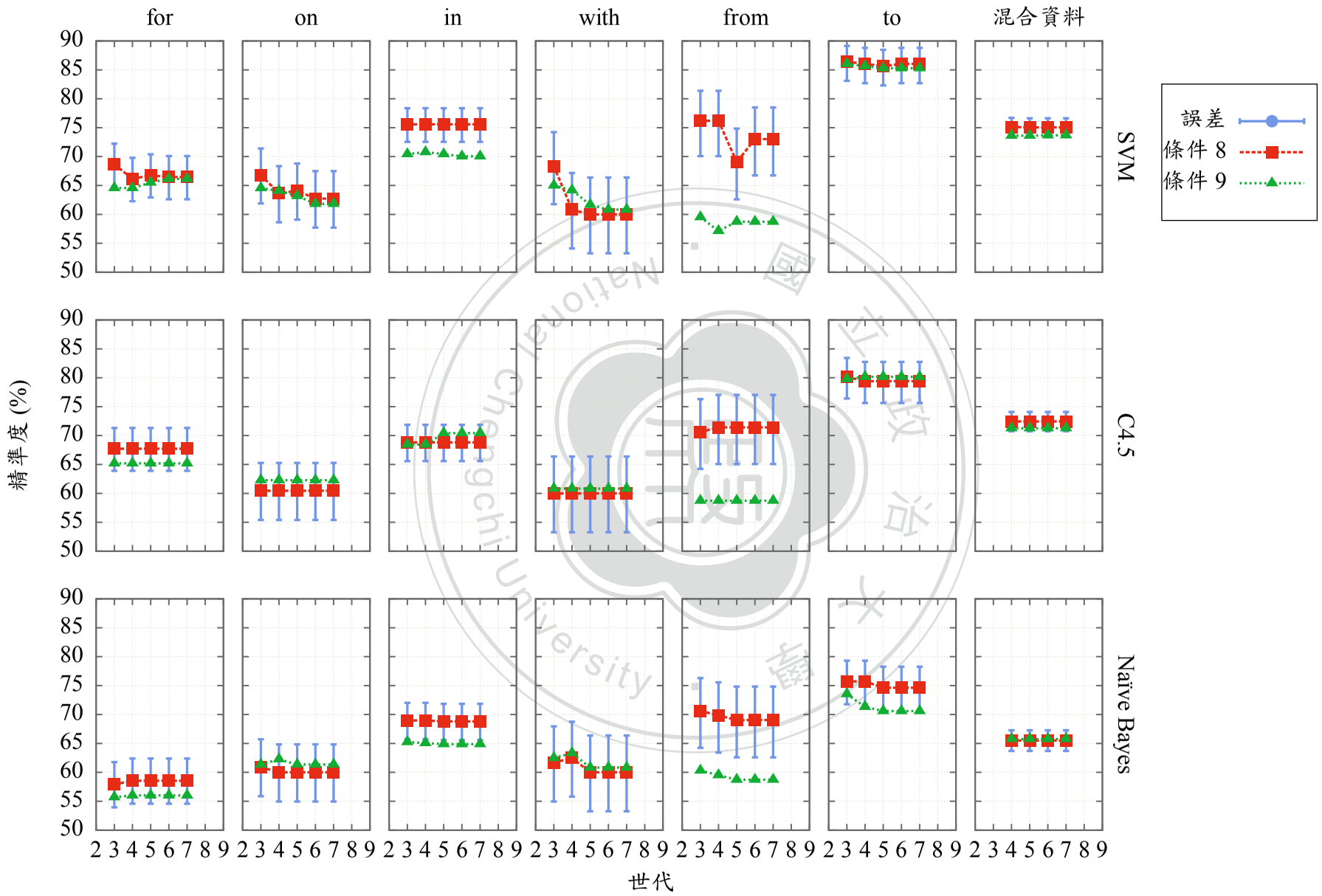

圖 5.2: 比較不同門檻值之共現同義詞集

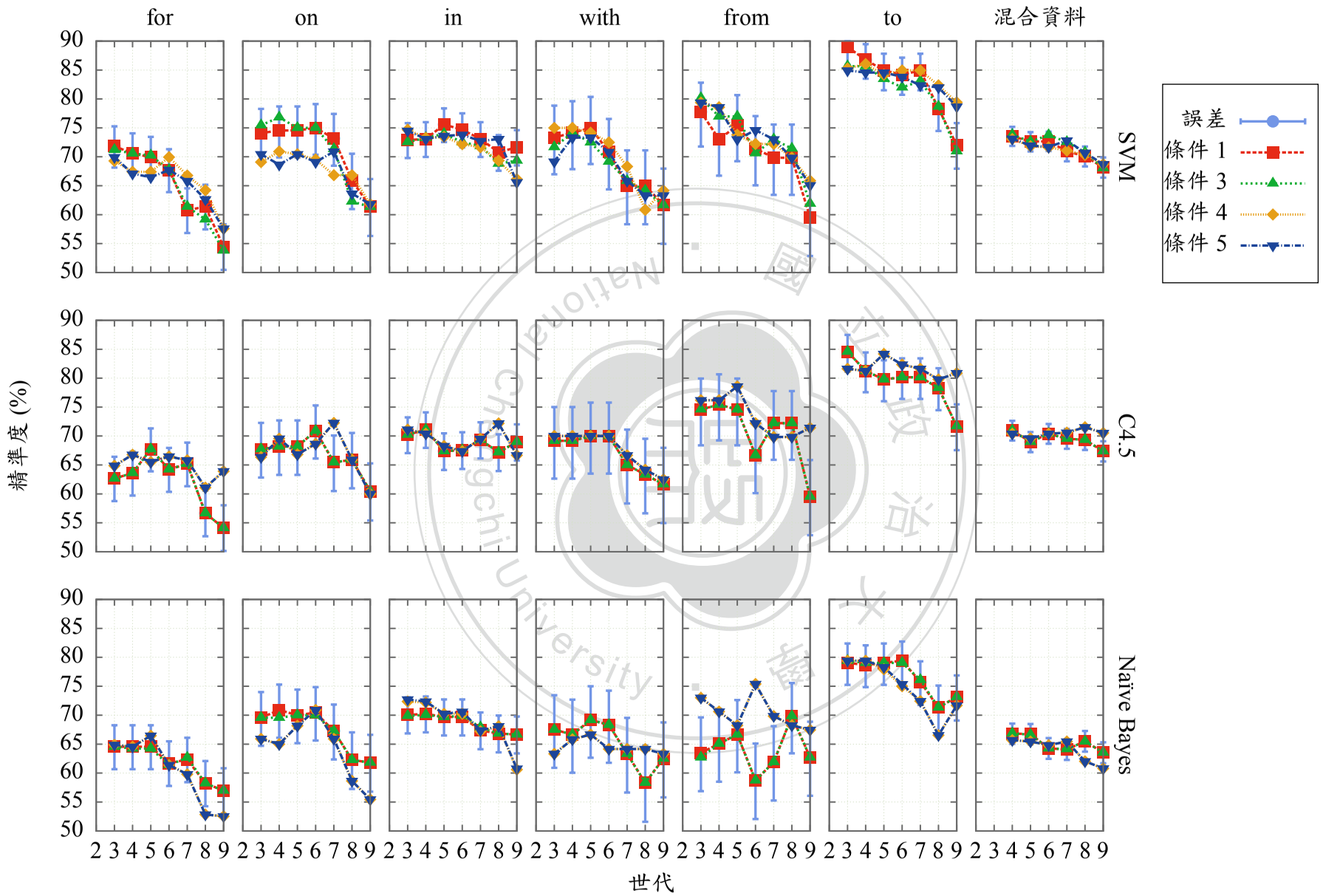

圖 5.3: 比較不同加權方法

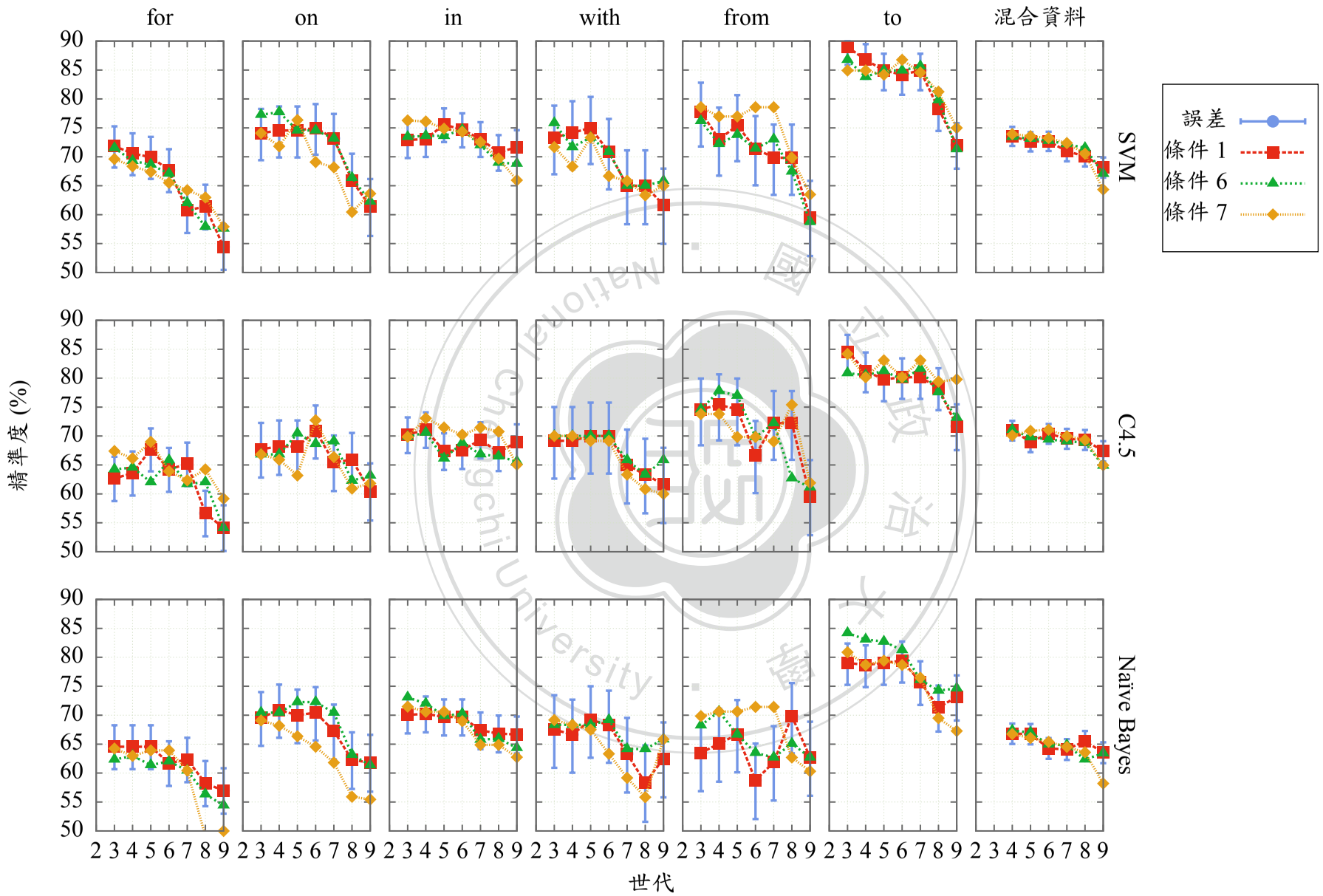

圖 5.4: 比較不同量化方法

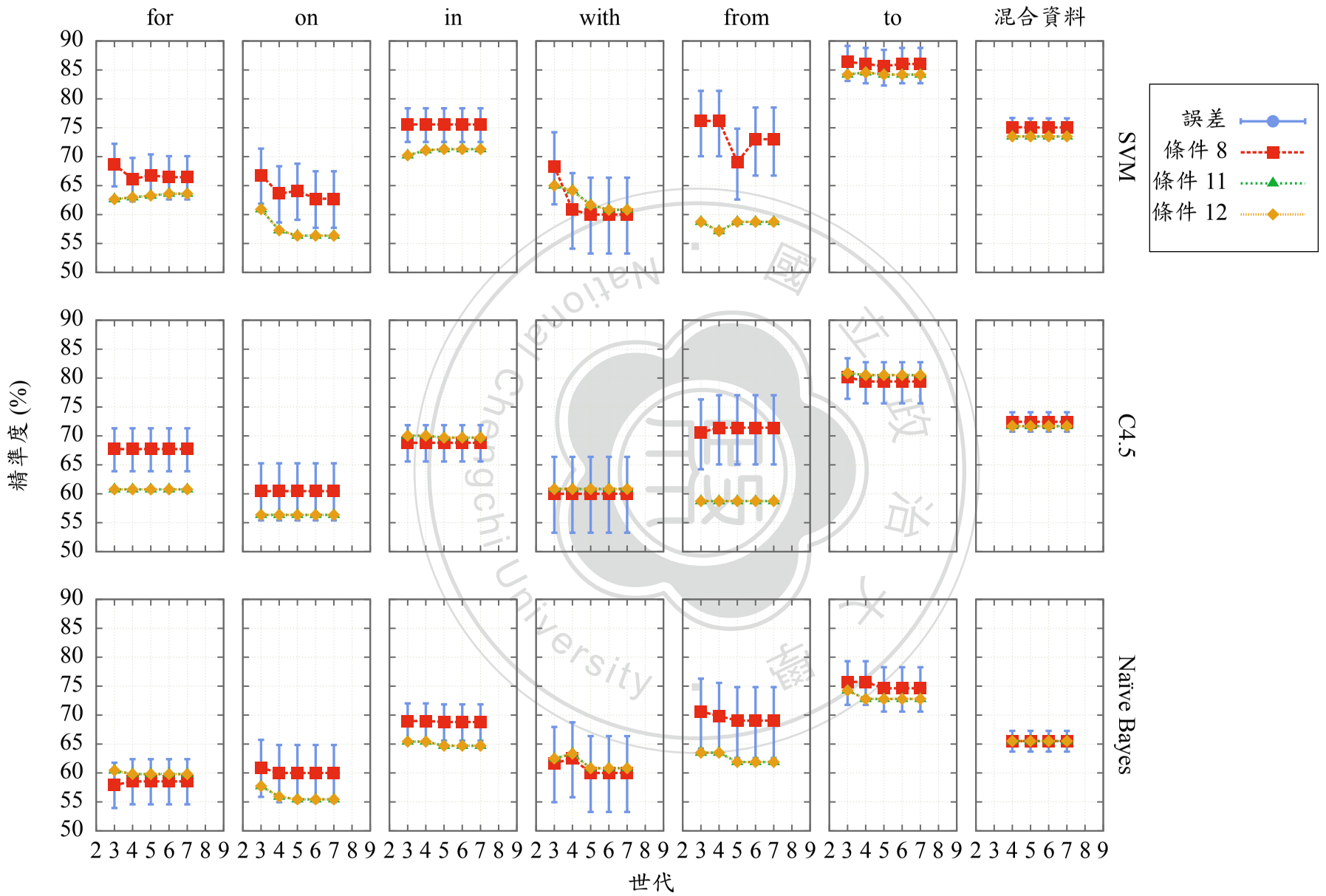

圖 5.5: 比較不同共現同義詞集組合

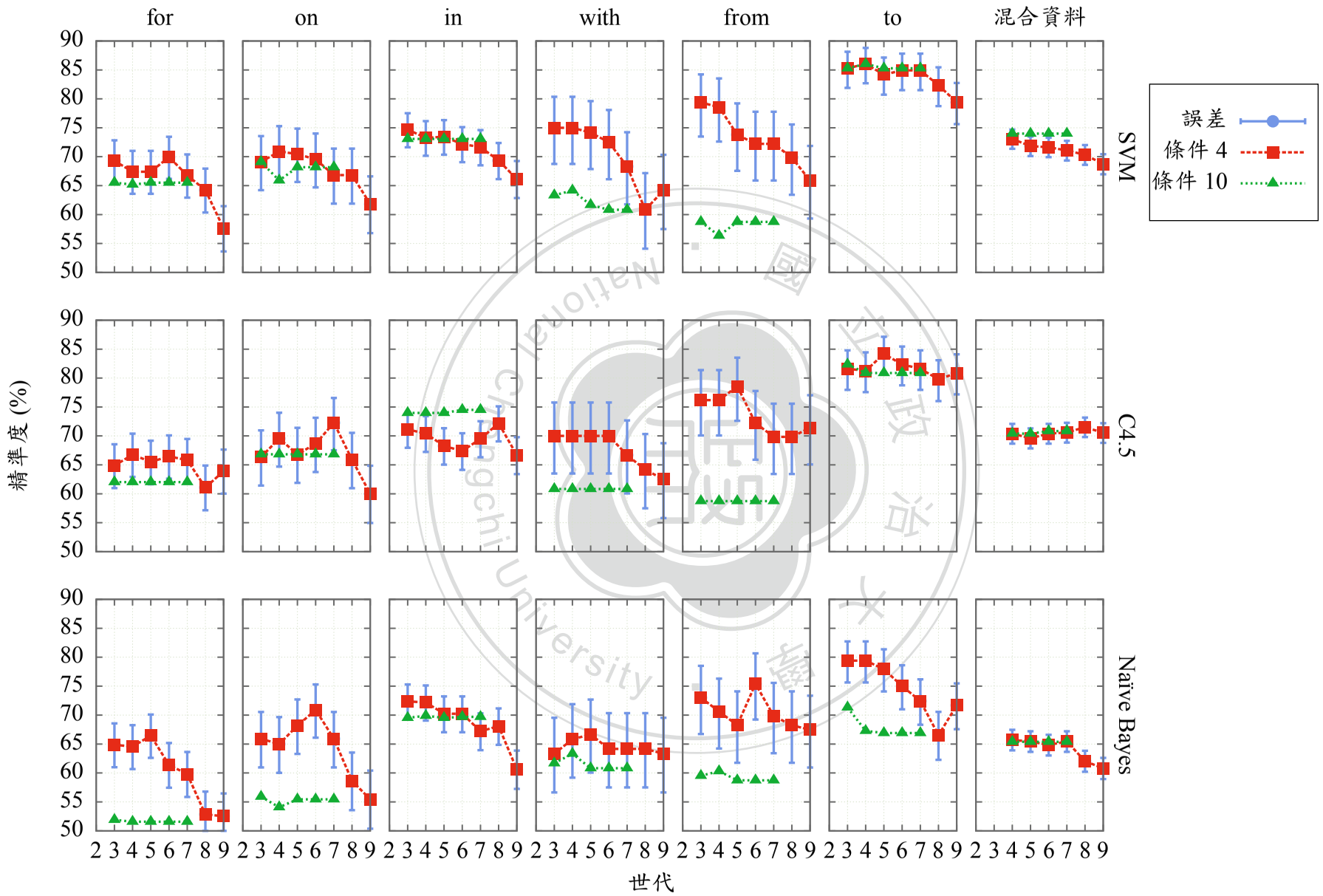

圖 5.6: 基於詞義頻率之詞頻與共現同義詞集組合之比較

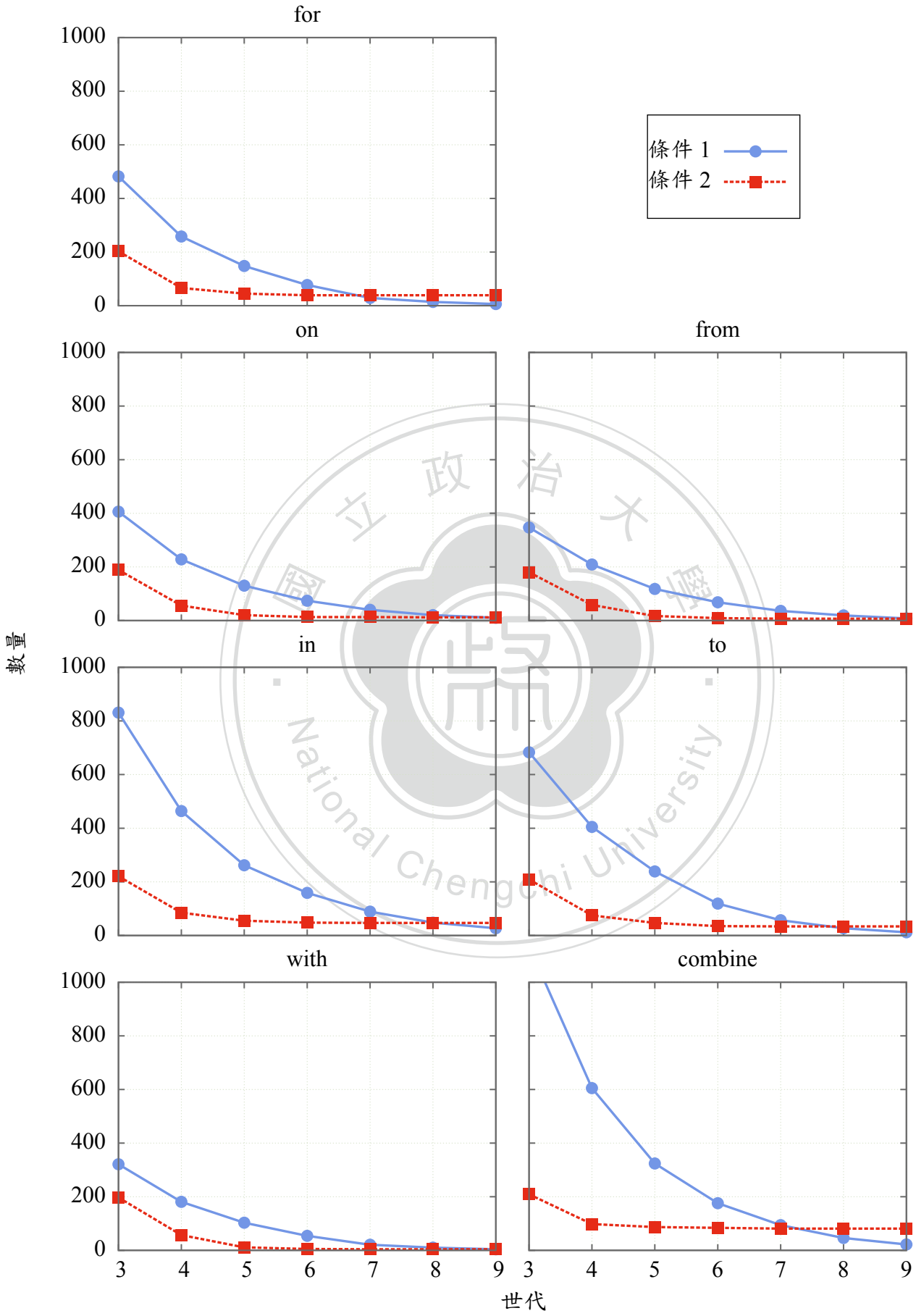

圖 5.7: 實驗組合一之特徵量

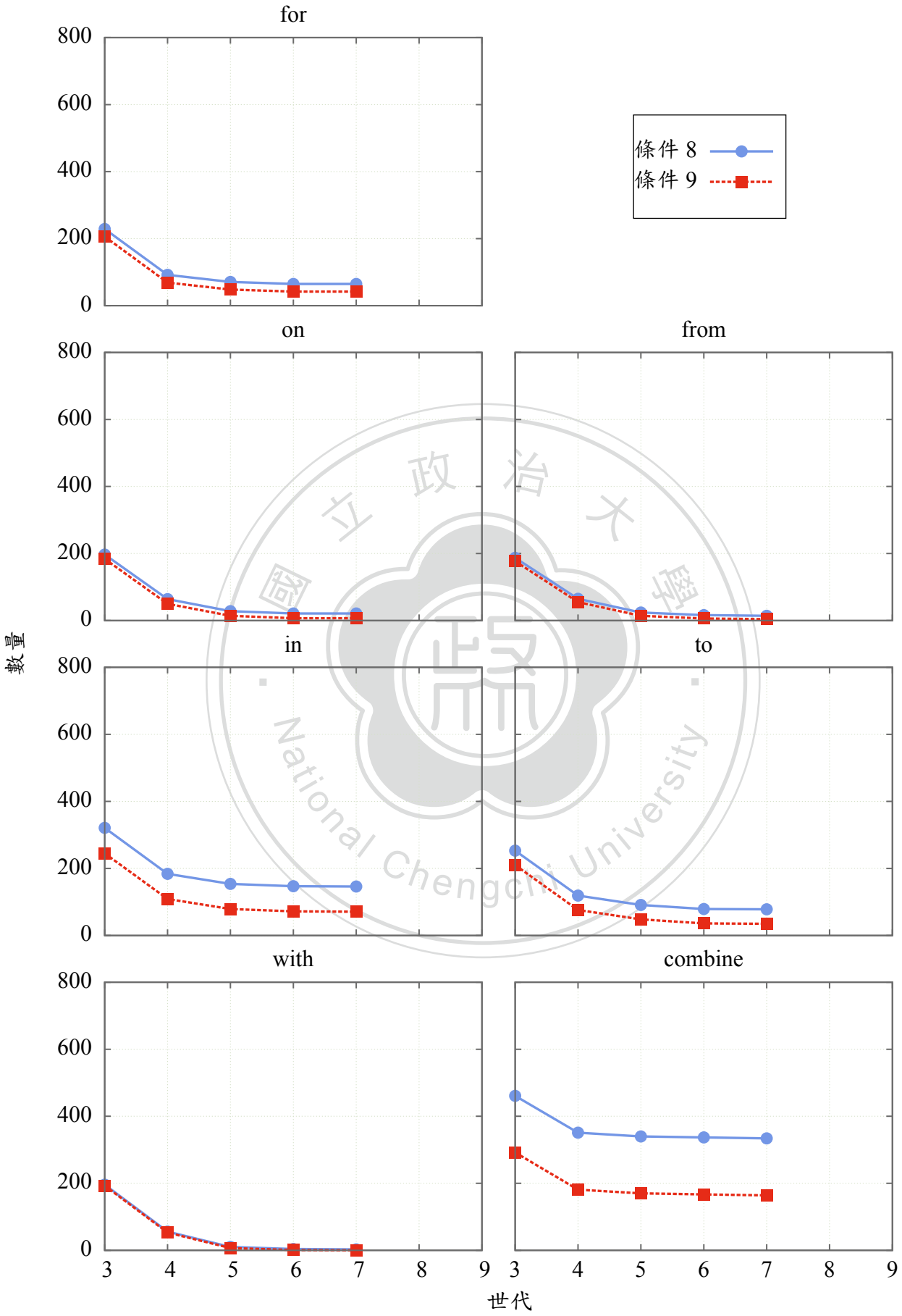

圖 5.8: 實驗組合二之特徵量

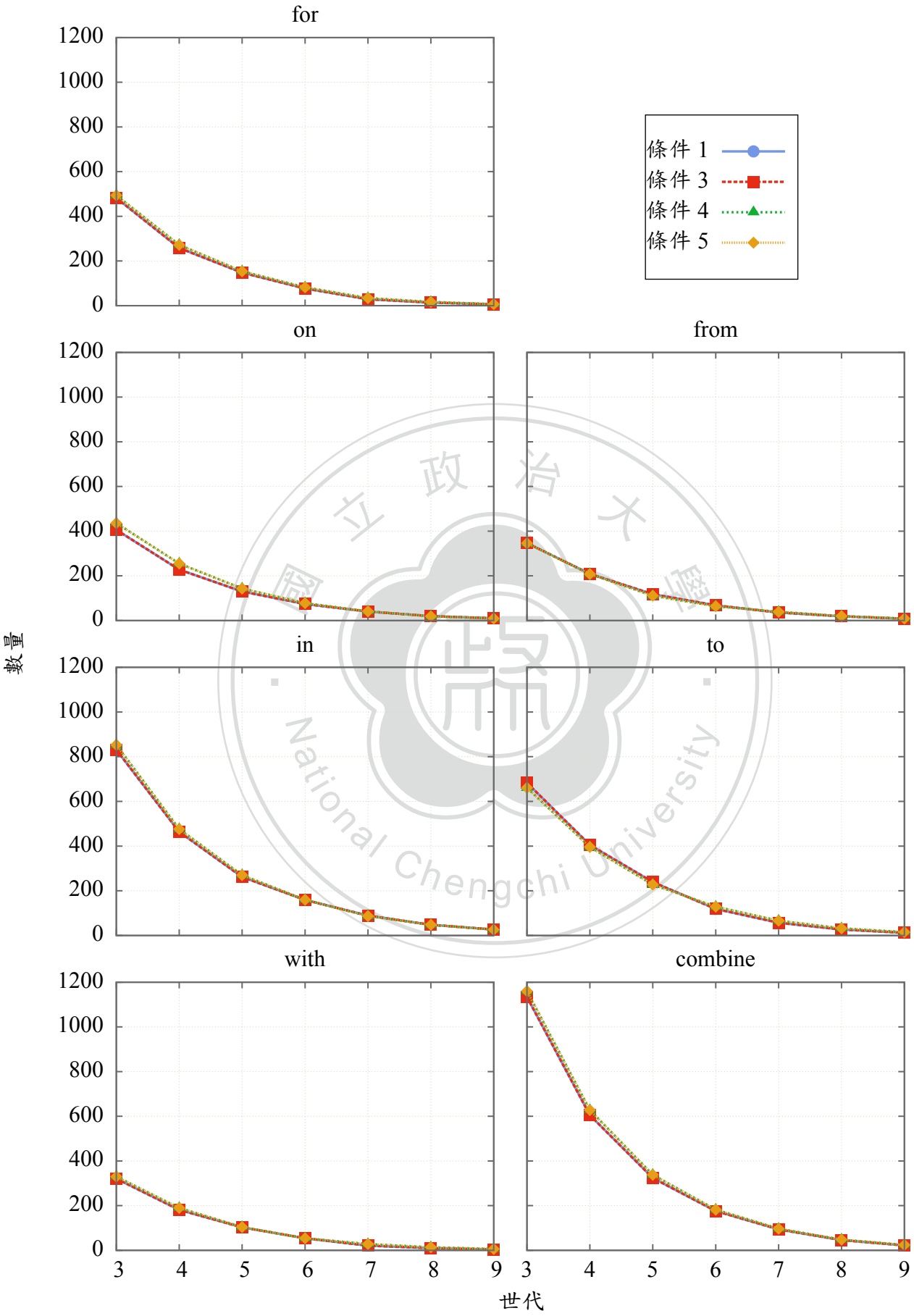

圖 5.9: 實驗組合三之特徵量

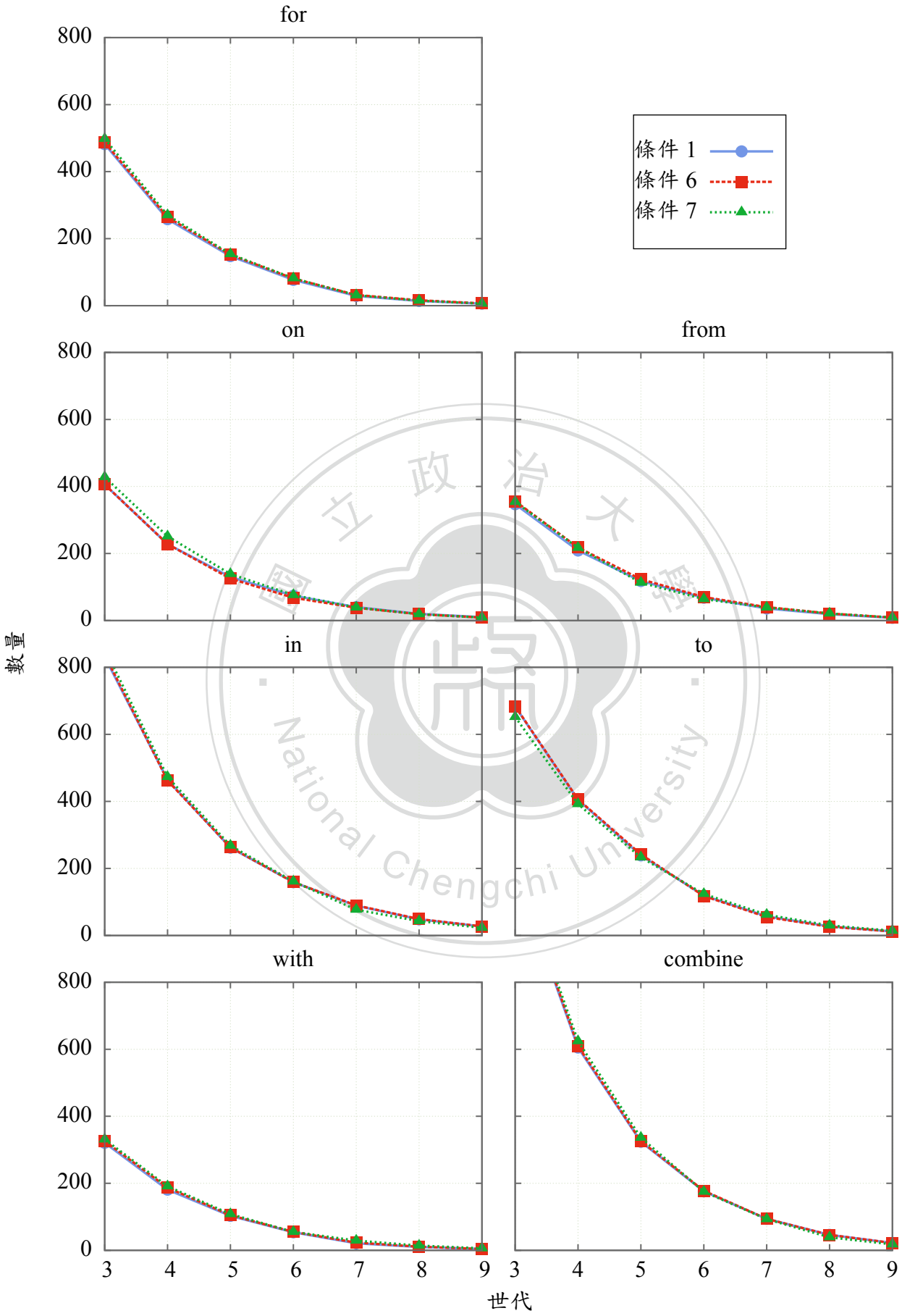

圖 5.10: 實驗組合四之特徵量

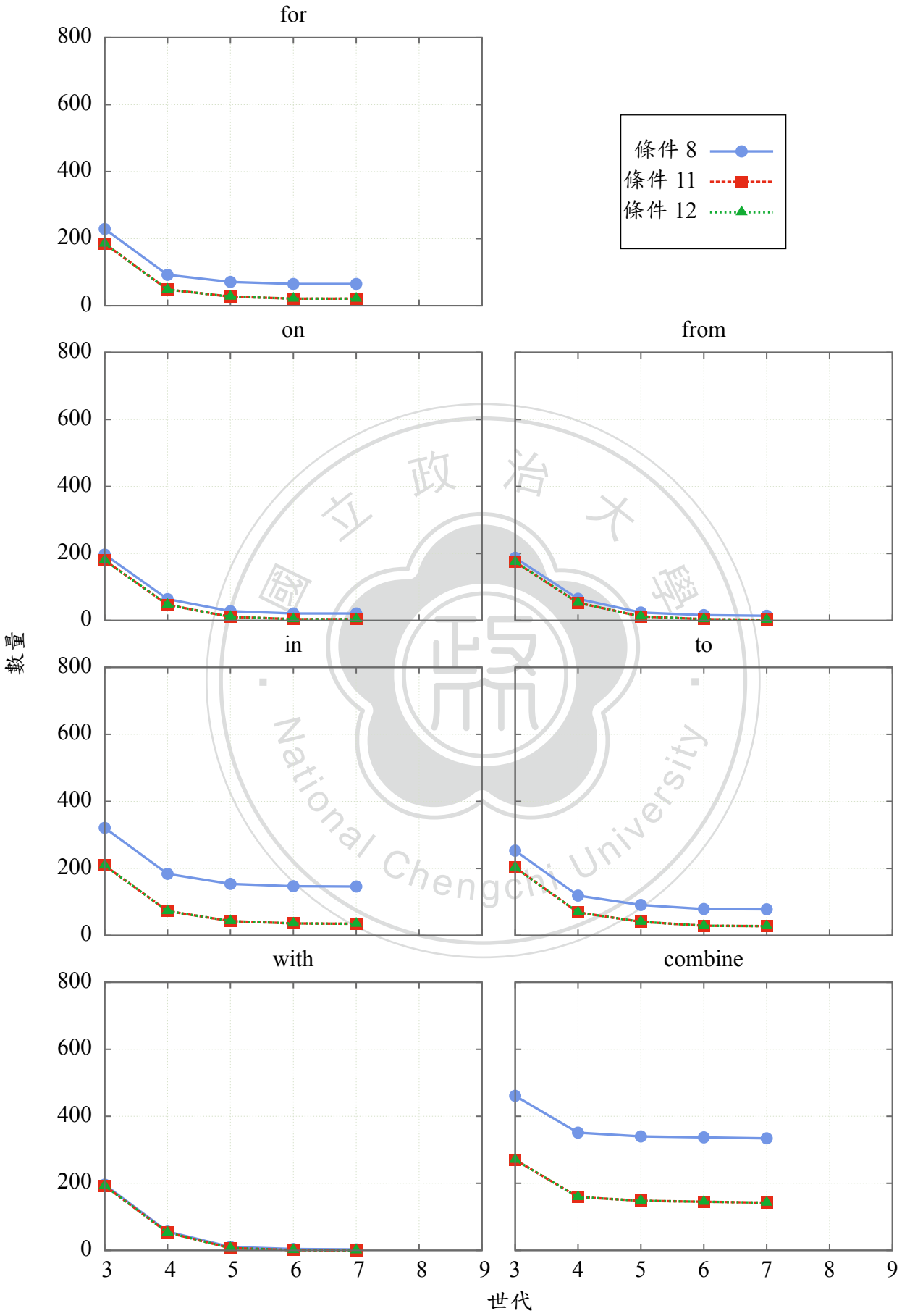

圖 5.11: 實驗組合五之特徵量

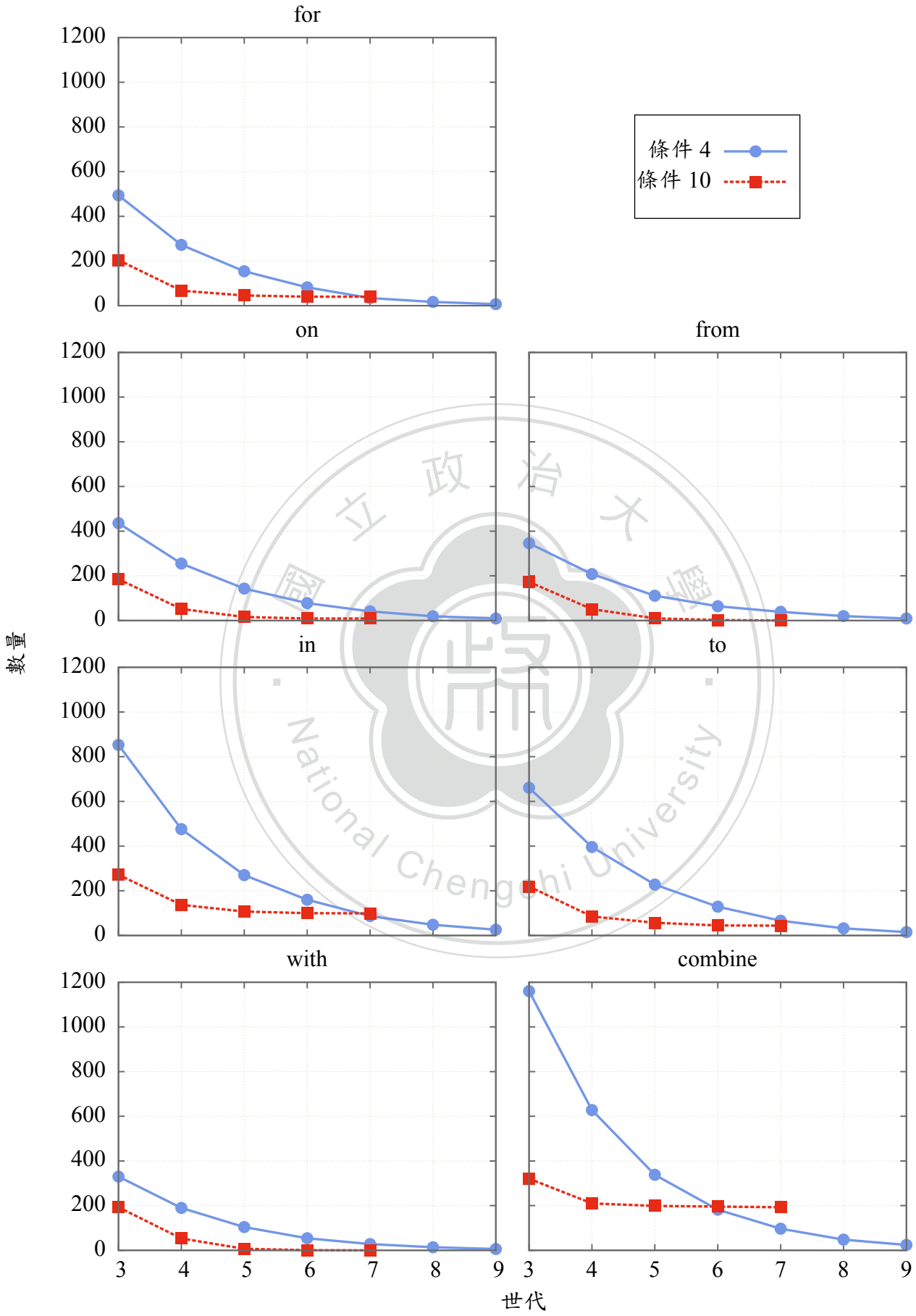

圖 5.12: 實驗組合六之特徵量

5.3.2 階層式特徵選擇之分析

我們從 5.3.1 節中的 12 組個條件選出表現較佳的模型做為比較。利用訓練語料訓練傳統模型後，再以驗證語料當作內部測試資料，用以衡量精準度，選出表現最佳的模型做為是最終比較的模型，最後用測試語料當作正式上線用資料，測試語料的精準度將被我們視為是最後比較的依據。若其中有二個以上的條件所建構的模型精準度一樣好，則會將一樣好的模型都納入考量，最後會取算數平均數計算精準度。如表 5.11和表 5.12的結果，分別是 RRR 與 PTB3 語料庫的結果。

表 5.11和表 5.12中的條件表示在該條件下表現最佳的模型，RRR 語料表現最佳的模型均是由第三個世代所選出，而且演算法均是 SVM。PTB3 語料庫的驗證語料較少，因此我們參考了 RRR 語料庫的經驗：只從第三個世代選擇且只選擇演算法是 SVM 的模型。混合資料表示所有介系詞是一起訓練，在測試時才分別計算各個介系詞的精準度，在表格最下方的混合資料表示整體一起測試的結果，在較右方混合資料欄位表示我們將混合資料結果中的每一個介系詞分開衡量精準度。基線表示是使用基準模型訓練的結果，並使用測試語料衡量精準度。算數平均表示將所有介系詞都視為是一樣重要，因此所有介系詞的權重都相同，而加權平均表示考慮到各個介系詞的數量，將數量視為是權重。這二種平均在計算時，不會將混合資料模型的精準度加入計算。

表 5.11和表 5.12中的算數平均與加權平均，都是表 5.12略勝一些，除了基線表現稍不一樣外。二者整體在語料量差不多的情況下，影響精準的主因除了訓練語料不一樣外，還可以發現測試語料中有 2 個介系詞的分布的數量不太一樣，RRR 的介系詞“on”的數量較 PTB3 少，其次是介系詞“with”。在數量較少的情況下，很容易因多答對或答誤一個案例造成精準度變動幅度大。這可能是造成二者精準度有些差異的原因。

表 5.11和表 5.12被選出的較佳條件不太一樣。原因可能出在 PTB3 驗證語料過少的關係，使得驗證語料案例較不具代表性，所以選到的最佳模型並不是最好。另一方面

也有可能是因為數量少，所以很容易造成多答對或答錯一個答案，而使得準度變化幅度大，以致我們選擇到較不好的模型。

比較表 5.11和表 5.12單一模型、混合資料模型與基線的精準度，單一模型與混合資料模型的精準度均較基準模型高。這表示透過階層式的特徵選擇比起只選擇固定且較高層次的特徵來的有效。

比較表 5.11和表 5.12單一模型、混合資料模型精準，二者的精準度差不多。雖然差不多，但單一模型可以有效的針對各個介系詞找出具代表性的同義詞集做為特徵，使我們觀察它們的關聯性，這是混合資料模型無法做到。且混合資料模型的語料量遠較單一模型大，這也可能是單一模型沒有超過混合資料模型的原因。若能提升單一模型語料量，單一模型應該還是有進步的空間。

表 5.11: RRR 傳統模型實驗結果

介系詞	條件	精準度（%）			
		單一		混合語料	
		驗証語料	測試語料	測試語料	基線
for	1	71.84	74.13	73.75	67.57
on	6	77.27	69.81	72.96	67.30
in	7	76.29	79.21	79.78	72.47
with	6	75.83	66.67	67.78	63.33
from	3	80.16	79.35	78.26	75.00
to	1	88.97	85.31	83.89	70.62
算數平均		78.39	75.75	76.07	69.38
加權平均			76.95	77.21	69.67
混合資料	8	75.11	77.21		

5.3.3 高階模型建構之分析

我們利用驗證語料測試傳統模型的結果來建構高階模型，並將測試語料做為正式上線後的測試結果，參考 4.3.3節做法。我們設定了幾種不同的標準來建構高階模型，如表 5.4所示，結果如表 5.13和表 5.15所示。二張表上半部分表示由單一模型所建構的高階模型，下半部分表示由混合資料模型所建構。NB 表示 Naïve Bayes，SVM、C4.5 和

表 5.12: PTB3 傳統模型實驗結果

介系詞	條件	精準度（%）			
		單一		混合資料	
		驗證語料	測試語料	測試語料	基線
for	4	72.58	77.63	77.63	76.64
on	3,5,6	80.77	75.29	79.02	67.32
in	5	87.97	77.47	78.68	70.54
with	5	67.20	75.20	68.00	63.20
from	1,2,5	80.00	76.15	79.67	71.54
to	1,8	88.00	83.59	81.47	71.04
算數平均		82.66	77.56	77.41	70.05
加權平均			77.94	78.22	70.84
混合資料	4	82.44	78.22		

Naïve Bayes 表示高階模型選用的演算法。高階混合資料模型，表示是由混合資料模型所建構的高階模型。高階模型與高階混合資料模型的 (1)、(2) 與 (3) 表示選擇傳統模型的條件。

表 5.13上半部分中，可以觀察到在較嚴苛的選擇條件下，精準度稍微有逐漸改善，高階 (3) 比高階 (2) 好，高階 (2) 比高階 (1) 好，但改善的幅度不大。與表 5.11單一模型的訓練語料相比改善的更是有限，且並非所有高階模型都是改善精準度，有些反而使精準度還下降。比較有趣是當傳統模型都是 SVM，可以發現高階模型是使用 Naïve Bayes 的效果最好。研究中，我們單獨對每個介系詞訓練模型，因此高階模型與條件選擇上，我們也可以單獨對每一個介系詞選出一個表現最好的。我們從表 5.13上半部中，挑選每個介系詞最好的結果，如表 5.14所示，整體平均有再改善一些。表 5.13下半部分，不管是使用何種高階模型與條件，精準度都比原來的差。

表 5.15上半部分中的結果幾乎都較表 5.11的單一模型差，可能的原因是出在 PTB3 的驗證語料量較少，使得高階模型訓練語料不足，導致高階模型的效果較差。如果將每個介系詞效果最好的高階模型挑出，結果如表 5.16所示，改善的幅度是有限。表 5.15下半部分的結果與 RRR 語料相反，但改善的幅度仍是有限的。

表 5.13: RRR 高階模型實驗結果

	精準度（%）								
模型	高階 (1)			高階 (2)			高階 (3)		
介系詞	SVM	C4.5	NB	SVM	C4.5	NB	SVM	C4.5	NB
for	68.73	68.73	72.97	74.90	71.43	72.59	74.13	71.43	72.97
on	72.33	67.30	68.55	71.07	67.30	72.33	69.81	67.30	73.58
in	76.12	79.21	75.84	77.25	79.21	77.25	77.53	79.21	78.93
with	65.56	66.67	72.22	70.00	66.67	70.00	67.78	66.67	70.00
from	79.35	82.61	77.17	72.83	82.61	76.09	75.00	82.61	80.43
to	86.26	85.31	81.04	88.15	85.31	83.41	85.78	85.31	85.31
算數平均	74.72	74.97	74.63	75.70	75.42	75.28	75.01	75.42	76.87
加權平均	75.24	75.66	74.98	76.95	76.26	76.01	76.26	76.26	77.46
	精準度（%）								
模型	高階混合資料 (1)			高階混合資料 (2)			高階混合資料 (3)		
介系詞	SVM	C4.5	NB	SVM	C4.5	NB	SVM	C4.5	NB
for	69.5	71.81	73.36	72.2	71.81	69.5	71.81	72.97	69.88
on	72.96	71.7	72.96	74.21	71.7	74.84	74.84	71.7	70.44
in	79.49	78.93	78.09	78.65	78.93	78.93	80.34	78.65	79.78
with	64.44	62.22	65.56	64.44	62.22	66.67	68.89	61.11	65.56
from	78.26	77.17	77.17	81.52	77.17	80.43	76.09	79.35	79.35
to	85.31	82.46	81.52	81.99	82.46	83.41	83.89	81.99	84.36
算數平均	74.99	74.05	74.78	75.50	74.05	75.63	75.98	74.30	74.90
加權平均	76.18	75.58	75.92	76.35	75.58	76.26	77.12	75.75	76.01

表 5.14: RRR 最佳高階模型

介系詞	精準度 (%)
for	74.13
on	73.58
in	79.21
with	72.22
from	82.61
to	88.15
算數平均	78.32
加權平均	78.66

表 5.15: PTB3 高階模型實驗結果

	精準度（%）								
模型	高階 (1)			高階 (2)			高階 (3)		
介系詞	SVM	C4.5	NB	SVM	C4.5	NB	SVM	C4.5	NB
for	71.71	67.11	77.30	75.66	74.67	76.97	76.97	74.67	79.28
on	70.24	67.32	70.24	67.32	67.32	72.68	73.66	70.24	76.10
in	78.16	75.74	76.26	77.64	75.74	76.60	78.16	75.74	78.51
with	69.60	59.20	71.20	69.60	59.20	68.00	69.60	59.20	72.00
from	72.36	62.60	76.42	73.98	62.60	72.36	77.24	62.60	72.36
to	78.76	79.54	77.99	78.76	79.54	79.54	83.40	70.27	83.78
算數平均	73.47	68.58	74.90	73.83	69.84	74.36	76.50	68.79	77.00
加權平均	74.89	71.31	75.58	75.20	72.76	75.64	77.46	71.63	78.22
	精準度（%）								
模型	高階混合資料 (1)			高階混合資料 (2)			高階混合資料 (3)		
介系詞	SVM	C4.5	NB	SVM	C4.5	NB	SVM	C4.5	NB
for	76.64	77.63	77.96	79.28	77.63	79.61	75.99	75.33	78.62
on	80.49	79.02	80.00	78.05	79.02	80.00	79.02	77.56	77.56
in	80.07	78.68	77.99	79.72	78.68	79.38	80.07	78.68	79.55
with	69.60	68.00	70.40	76.80	68.00	75.20	75.20	69.60	77.60
from	74.80	79.67	71.54	74.80	79.67	73.98	77.24	76.42	76.42
to	84.94	81.47	83.78	83.78	81.47	85.33	83.01	80.31	83.78
算數平均	77.76	77.41	76.95	78.74	77.41	78.92	78.42	76.32	78.92
加權平均	79.03	78.22	78.09	79.47	78.22	79.73	79.03	77.27	79.41

表 5.16: PTB3 最佳高階模型

介系詞	精準度 (%)
for	79.28
on	76.10
in	78.51
with	72.00
from	77.24
to	83.78
算數平均	77.82
加權平均	78.59

5.3.4 綜合評比與最大熵值法之分析

表 5.17是 RRR 語料綜合評比的結果。分布線表示亂猜可以達到最佳的精準度，參考式 (3.3)。Max Enp 表示最大熵值法。最佳高階模型表示表 5.14的結果，高階混合資料(3)SVM 表示高階模型是演算法 SVM 且利用條件 3，也就是由表 5.13中下半部分被挑選出最佳的高階混合資料模型。

表 5.17我們的每一組實驗結果，除基線不明顯外，它其結果不僅較分布線佳而且也較最大熵值法好。細看表 5.17的基線，基線的精準度與最大熵值法的精準度差不多，但基準模型只挑選較高層的同義詞集和同義詞集的種類做為特徵所訓練，因此特徵在數量上並沒有很多，再這樣的情況下，略勝於最大熵值法，也算是有不錯的表現。

表 5.17: RRR 實驗結果

介系詞	個數	分布線 (%)	精準度（%）					
			Max Enp	基線	單一	混合資料	最佳高階單一	高階混合資料 (3)SVM
for	259	57.14	62.55	67.57	74.13	73.75	74.13	71.81
on	159	58.49	67.92	67.30	69.81	72.96	73.58	74.84
in	356	56.18	75.56	72.47	79.21	79.78	79.21	80.34
with	90	61.11	60.00	63.33	66.67	67.78	72.22	68.89
from	92	65.22	69.57	75.00	79.35	78.26	82.61	76.09
to	211	63.98	72.51	70.62	85.31	83.89	88.15	83.89
算數平均		60.35	68.02	69.38	75.75	76.07	78.32	75.98
加權平均		59.21	69.41	69.67	76.95	77.21	78.66	77.12
混合資料	1167	50.04						

5.3.5 綜合評比與 Stanford 剖析器之分析

表 5.18是 PTB3 語料綜合評比的結果。其中 SP 表示 Stanford 剖析器。最佳高階模型表示表 5.16的結果，高階混合資料 (2)NB 表示高階模型是 Naïve Bayes 且利用條件 2。

表 5.18中可以看出我們的方法不僅較分布線好，且也遠較 Stanford 剖析器好。但這是對我們比較有利的情況所計算出的結果，原因是 Stanford 剖析器在剖析句子時，若沒有辦法將句子中的動詞片語辨識出，就沒有辦法判斷定位問題，這時我們會將該結果視為是錯誤分類，因此使得 Stanford 剖析器精準度降低。

上述計算的方法類似在計算召回率，因此我們重新計算評量的方式，將結果分成二類答對與答錯。而且我們發現 Stanford 剖析器除了會將原本不是動詞片語的結構誤認為是動詞片語，反之也有可能原是動詞片語的結構卻無法辨識，重新計算 Stanford 剖析器的回答狀況後，結果如表 5.19所示，P 表示精準率，R 是召回率，F 是綜合評量，由此可知 Stanford 剖析器有作答率的問題。最後，我們將二種情況的案例去除，僅留下我們方法與 Stanford 剖析器都可作答的案例，可得表 5.20。這時候比較我們與 Stanford 剖析器的結果，可以看到二者的精準度是差不多。

比較我們的方法與 Sanford 剖析器各有優缺點，如果我們能夠事先給定四個中心詞，那麼我們方法是遠勝於 Stanford 剖析器。然而 Stanford 剖析器只要提供完整的句子便能夠剖析出句子定位修飾對象，但 Stanford 剖析器剖析整個句子所考慮的語境的資訊是較多，我們的方法考慮的語境資訊是較稀少的。

表 5.18: PTB3 實驗結果（1）

介系詞	個數	分布線 (%)	精準度（%）					
			SP	基線	單一	混合資料	最佳高階單一	高階混合資料 (2)NB
for	304	62.17	60.86	76.64	77.63	77.63	79.28	79.61
on	205	50.73	63.90	67.32	75.29	79.02	76.10	80.00
in	577	50.09	64.64	70.54	77.47	78.68	78.51	79.38
with	125	59.20	64.80	63.20	75.20	68.00	72.00	75.20
from	123	62.60	66.67	71.54	76.15	79.67	77.24	73.98
to	259	70.27	66.80	71.04	83.59	81.47	83.78	85.33
算數平均		59.18	64.61	70.05	77.56	77.41	77.82	78.92
加權平均		57.44	64.34	70.84	77.94	78.22	78.59	79.73
混合資料	1593	52.73						

表 5.19: SP 答題狀況

介系詞	P(%)	R(%)	F(%)
for	87.26	74.90	80.61
on	89.73	75.72	82.13
in	88.39	79.53	83.73
with	91.01	72.97	81.00
from	88.17	78.10	82.83
to	87.82	79.72	83.57

表 5.20: PTB3 實驗結果（2）

介系詞	個數	分布線 (%)	精準度（%）		
			SP	單一	混合資料
for	247	63.97	74.90	77.63	77.63
on	173	50.29	75.72	75.28	79.02
in	469	50.32	79.53	77.47	78.68
with	111	56.76	72.97	67.20	75.20
from	105	62.86	78.10	76.15	79.67
to	217	70.51	79.72	83.59	81.47
算數平均		59.12	76.82	76.22	78.61
加權平均		57.72	77.53	77.25	78.77
混合資料	1322	52.27			

5.4 實驗分析：介系詞推薦

本節我們探討 RRR 語料庫和由華爾街日報與紐約時報組成的大語料的實驗成果，探討分析的內容有：不同條件的組合對介系詞的影響、高階模型的建構、綜合比較與大語料庫的分析。這部分的實驗成果較難與其它實驗做比較，因此我們藉由參考其它文獻的方法，以瞭解目前的成果。

5.4.1 不同條件組合之分析

介系詞推薦與介系詞片語定位問題是利用相同的方法來解決問題。整體比較結果如圖 5.13所示，x 軸表示世代，y 表示精準度百分比，第一欄到第三欄依序表示使用的演算法分別是 SVM、C4.5 和 Naïve Bayes，第一列到第六列表示不同的實驗組合，依序對應表 5.5、表 5.6、表 5.7、表 5.8、表 5.9和表 5.10。每個世代的特徵量如表 5.14所示，x 軸表示世代，y 表示特徵數量。

整體而言，圖 5.13裡，觀察三種演算法優劣有大致的驅勢，SVM 勝過 C4.5 與 Naïve Bayes，後二者又以 C4.5 較佳。對於實驗組合的而言，雖圖 5.13的精準度都差不多，但組合 1、組合 5 和組合 6 是有影響，這些組合分別是考慮門檻值的高低、共現同義詞集組合與比較詞頻和共現同義詞集的條件對於精準度的影響力。圖 5.13的組合 1，顯示選同義詞集的頻率時，門檻值不應該設太高，如果將每個世代使用的同義詞集依頻率排序，可發現與 Zipf's law 有一樣的曲線，說明常用的同義詞集是有集中的現像，因此門檻設太高容易一開始就將重要的同義集詞刪除。圖 5.13的組合 5，比較不同的共現同義詞集組合，考慮到共現同義詞集組合是比較嚴苛的條件，雖精準度差不多，但因特徵量較少，因此共現同義詞集組合條件是有用。圖 5.13的組合 6，在差不多的精準度下，考慮共現同義詞集與詞頻雖精準度差不多，但共現同義詞集是較嚴苛的條件，所以

特徵量會較少，而在較少的特徵量下，仍然可以表現與考慮詞頻相當，因此這也是有用的條件，可以有效的大幅減少不必要的特徵且效果依舊不錯。

我們先以驗證語料做為內部的測試資料，找出最佳的模型，結果在考慮條件組合 8 下由 SVM 訓練出的模型是最佳的模型。接著再用測試語料做為內部測試資料，用以衡量最後的結果，結果可參考表 5.21。我們的方法精確度遠勝於分布線，因此可說我們的模型預測的答案不是隨意猜測。若單獨看這 6 個介系詞，整體分類效果好壞依序為“to”、“in”、“on”、“for”、“from”、“with”。

表 5.21: RRR 傳統模型之結果

介系詞	個數	分布線 (%)	驗證語料			測試語料		
			P(%)	R(%)	F(%)	P(%)	R(%)	F(%)
for	263	22.91	48.64	51.94	50.23	48.13	49.05	48.59
on	158	13.76	50.00	38.79	43.68	52.14	38.61	44.36
in	345	30.05	61.94	70.86	66.10	53.90	66.09	59.37
with	90	7.84	33.33	27.97	30.41	38.36	31.11	34.36
from	91	7.93	45.65	33.60	38.71	40.28	31.87	35.58
to	201	17.51	62.64	62.64	62.64	62.56	60.70	61.62
精準度	1148	20.48		55.28%			52.00%	

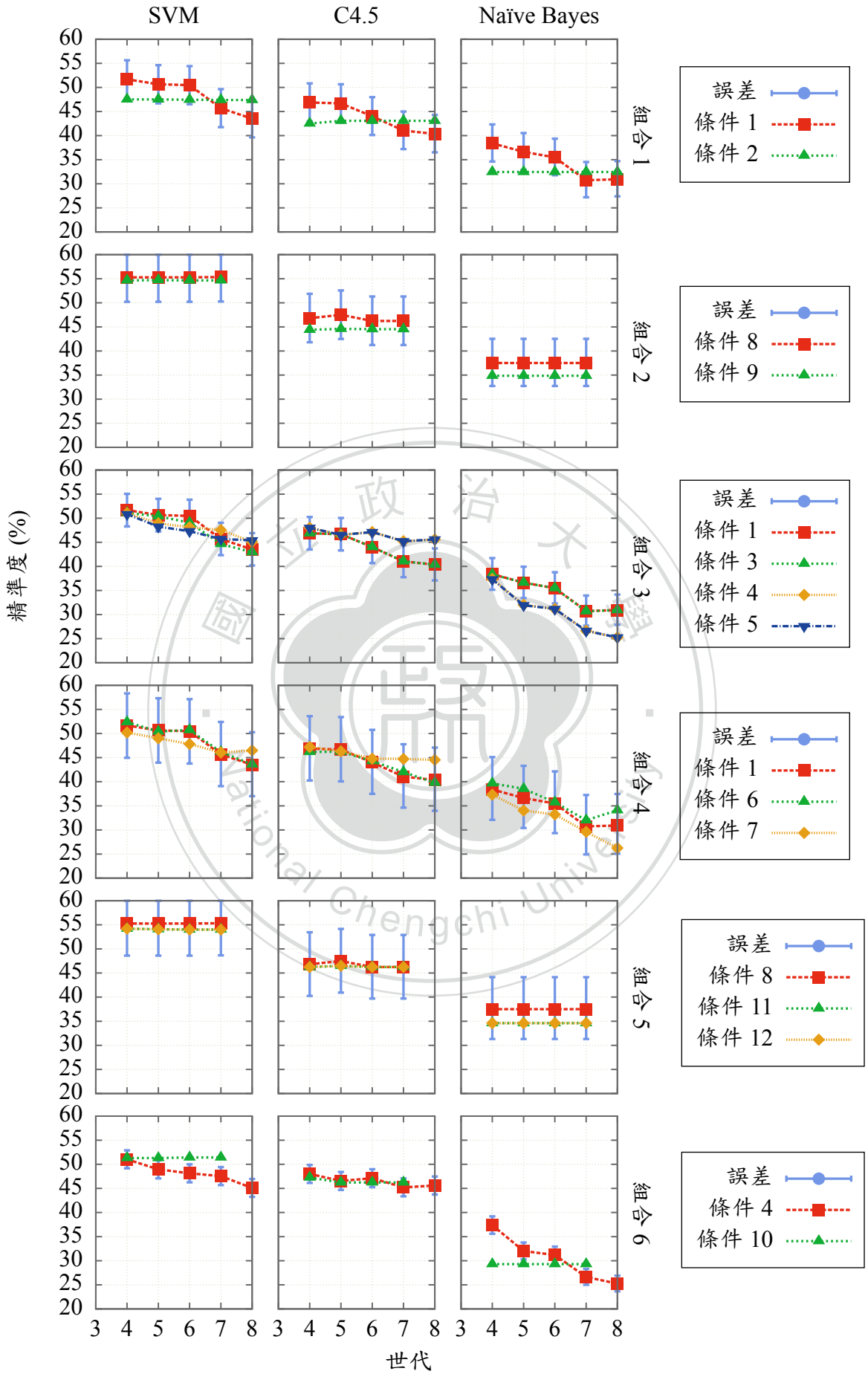

圖 5.13: 不同實驗組合之結果

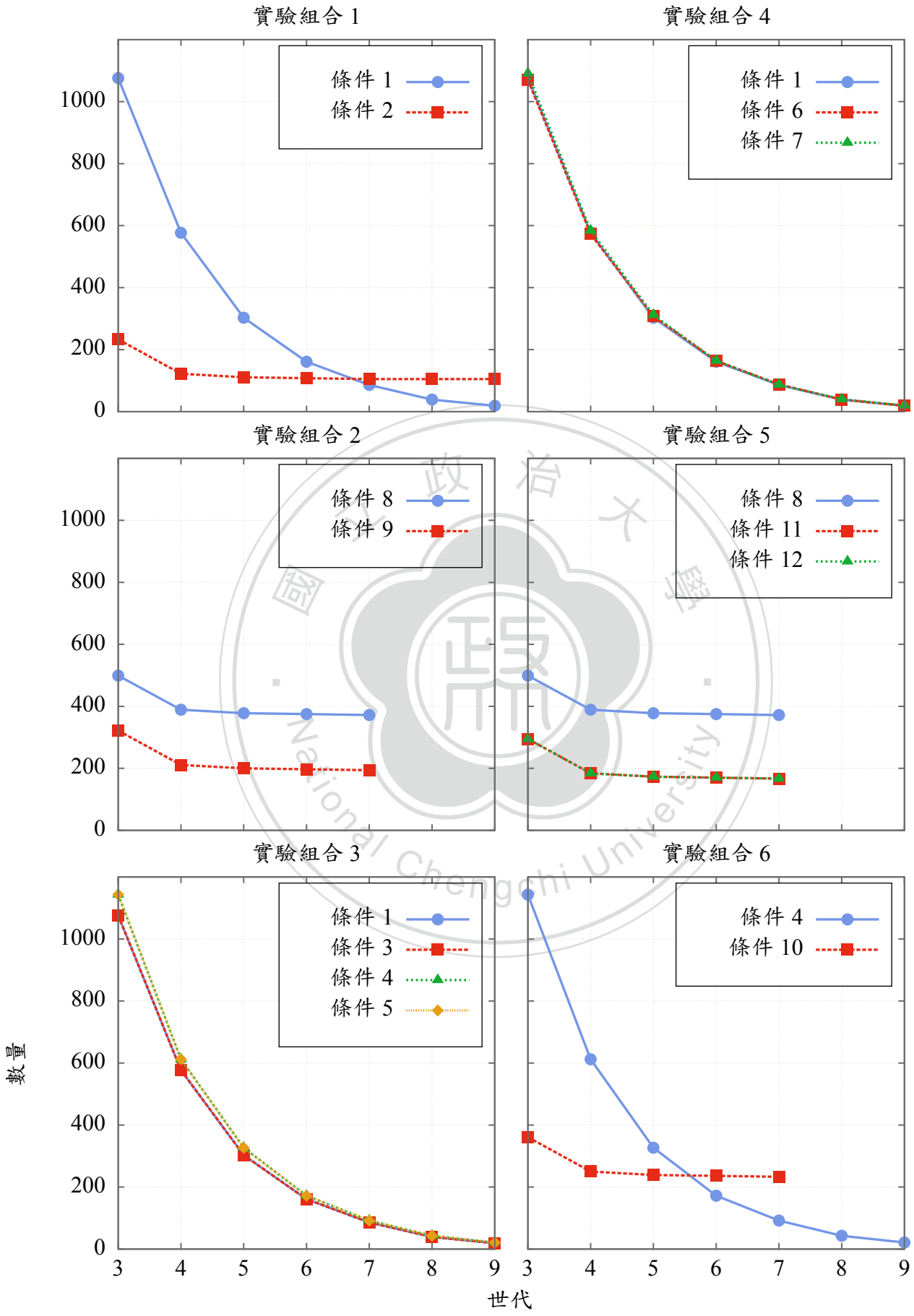

圖 5.14: 不同實驗組合之特徵量

5.4.2 高階模型建構之分析

當我們訓練完每個世代的傳統模型後，我們共可得到 62 個模型。再將驗證語料給這 62 個傳統模型測試，並參考表 5.4裡的挑選傳統模型的條件，最後利用測試的結果建構高階模型。

表 5.22是實驗的結果，P 表示精確率，R 表示召回率，F 表示綜合評量。高階模型的 (1)、(2) 與 (3) 表示選擇傳統模型的條件。

觀察高階模型建構的結果，不管是利用哪一種挑選條件或演算法，對於高階模型影響似乎沒有很大。如果再與表 5.22裡最佳的傳統模型相比較，可發現高階模型可改善的幅度是有限。

5.4.3 綜合比較

我們從表 5.21的傳統模型與高階模型中，各選擇二個表現較佳的分類器，把這四個模型決策的狀況表示成混淆矩陣（confusion matrix），藉以觀察每一個模型分類的情況，結果參考表 5.23、表 5.25、表 5.24與表 5.26。

混淆矩陣中裡，以粗體的部分表示模型的決策偏好，最佳的情況是模型的決策偏好與實際答案是完全一致，這剛好是矩陣的對角線，如表 5.23與 5.25所示。表 5.24與 5.26則是較不理想的狀態，從這二個混淆矩陣可以看到模型決策偏好是“in”。事實上，在我們目前的實驗中，幾乎每一個模型的混淆矩陣，如果對角線的數量不是最多的，那麼模型的分類決策結果最多的就都是“in”，這表示我們的模型容易與“in”混淆。這也可以說明混淆矩陣中，“in”的召回率特別高的原因。次個模型偏好的類別是“for”，分類較差的“from”和“with”幾乎是最不容易有正確決策類別，推測是訓練語料的數量較其它者為少的原因。

表 5.22: RRR 高階模型實驗結果

模型	高階 (1)								
介系詞	SVM			C4.5			NB		
	P(%)	R(%)	F(%)	P(%)	R(%)	F(%)	P(%)	R(%)	F(%)
for	47.78	53.23	50.36	41.23	56.27	47.59	48.48	42.59	45.34
on	56.56	43.67	49.29	56.38	33.54	42.06	40.53	48.73	44.25
in	51.91	74.78	61.28	54.44	65.80	59.58	60.76	55.65	58.09
with	50.00	21.98	30.53	53.66	24.18	33.33	32.71	38.46	35.35
from	40.00	13.33	20.00	42.86	20.00	27.27	27.05	36.67	31.13
to	66.87	55.22	60.49	62.56	60.70	61.62	61.54	55.72	58.49
精準度 (%)		53.14			51.39			48.87	
模型	高階 (2)								
介系詞	SVM			C4.5			NB		
	P(%)	R(%)	F(%)	P(%)	R(%)	F(%)	P(%)	R(%)	F(%)
for	46.80	52.85	49.64	41.26	57.41	48.01	50.97	49.81	50.38
on	58.72	40.51	47.94	56.38	33.54	42.06	45.06	46.20	45.63
in	51.49	75.07	61.08	54.44	65.80	59.58	60.39	63.19	61.76
with	51.22	23.08	31.82	53.66	24.18	33.33	36.17	37.36	36.76
from	42.31	12.22	18.97	42.86	16.67	24.00	30.00	33.33	31.58
to	66.86	57.21	61.66	62.56	60.70	61.62	65.52	56.72	60.80
精準度 (%)		53.05			51.39			52.26	
模型	高階 (3)								
介系詞	SVM			C4.5			NB		
	P(%)	R(%)	F(%)	P(%)	R(%)	F(%)	P(%)	R(%)	F(%)
for	45.82	52.09	48.75	50.86	44.87	47.68	50.20	48.67	49.42
on	58.43	32.91	42.11	42.34	36.71	39.32	52.99	44.94	48.63
in	51.01	73.33	60.17	50.20	71.30	58.92	55.91	65.80	60.45
with	50.00	24.18	32.59	53.85	23.08	32.31	41.67	38.46	40.00
from	45.71	17.78	25.60	24.32	20.00	21.95	35.80	32.22	33.92
to	64.32	59.20	61.66	65.34	57.21	61.01	63.83	59.70	61.70
精準度 (%)		52.18			50.17			53.14	

表 5.23: 混淆矩陣: 高階模型 (2):Naïve Bayes

		決策答案					
		for	on	in	with	from	to
實際答案	for	**128**	20	59	21	8	27
	on	24	**71**	39	8	3	13
	in	46	23	**227**	10	21	18
	with	19	6	22	**29**	7	7
	from	13	5	32	3	**35**	3
	to	25	9	27	10	10	**120**

表 5.24: 混淆矩陣: 高階模型 (3):SVM

		決策答案					
		for	on	in	with	from	to
實際答案	for	**132**	14	80	7	5	25
	on	32	53	**54**	3	2	14
	in	44	13	**261**	3	7	17
	with	28	3	**37**	14	3	5
	from	17	5	**42**	2	22	3
	to	28	5	40	2	6	**120**

表 5.25: 混淆矩陣: 表現最佳的單一模型

		決策答案					
		for	on	in	with	from	to
實際答案	for	**129**	17	63	18	9	27
	on	27	**61**	48	6	4	12
	in	50	21	**228**	10	14	22
	with	20	5	26	**28**	6	5
	from	18	4	28	5	**29**	7
	to	24	9	30	6	10	**122**

表 5.26: 混淆矩陣: 表現次佳的單一模型

		決策答案					
		for	on	in	with	from	to
實際答案	for	**127**	14	73	16	6	27
	on	29	**55**	54	6	2	12
	in	42	14	**255**	7	9	18
	with	25	3	**33**	19	3	7
	from	15	6	**40**	3	23	4
	to	34	6	50	4	6	**101**

5.4.4 大語料庫

華爾街日報與紐約時報是數量較大語料庫，在受限硬體環境的情況下，較沒有辦法如 5.4.3節將所有條件組合一一跑過一次。因此我們從 5.4.3節的結果中，挑選了表現最佳的一個實驗組合並重複實驗。表現較較佳的模型是表 5.2中條件組合 8，在條件組合 8 的條件下，我們利用華爾街日報與紐約時報再重複一次實驗流程。

表 5.27是實驗的結果，因為類別較多，所以與 RRR 語料相比較精準度是稍有些下降，但二者表現差不多。單獨看每個介系詞效果，“on”的精確率最好的，“of”的召回率是最好，綜合評量最好的是“of”。與表 5.29混淆矩陣一起分析，可以看到大家幾乎都偏好“of”，第二名是“in”，這與 5.4.3節所觀察第一名偏好的“in”是不一樣的結果。我們發現模型的偏好可能與模型的特性無關，而是與介系詞分布數量的多寡比較有關，介系詞的召回率與分布線幾乎是正相關，因此推測我們的模型成效與語料的類別量是比較有關係。

表 5.27: 華爾街日報與紐約時報實驗結果

介系詞	個數	分布線 (%)	P(%)	R(%)	F(%)
of	2390	27.71	47.22	75.36	58.06
in	1801	20.88	50.48	61.13	55.30
for	916	10.62	39.38	26.53	31.70
to	892	10.34	50.54	31.73	38.98
on	768	8.91	55.85	41.02	47.30
with	572	6.63	38.06	16.43	22.95
from	413	4.79	38.62	13.56	20.07
at	359	4.16	36.36	27.86	31.55
as	239	2.77	41.28	18.83	25.86
by	162	1.88	26.83	6.79	10.84
about	112	1.30	39.44	25.00	30.60
精準度 (%)	8624	27.71		47.28	

在介系詞推薦實驗裡，目前所回顧論文中提及的語料取得都是困難的，這使得我們較難與其它方法互相比較。然而因為介系詞推薦，只要能夠取得文章，就可以利用現有的工具，以自動化的處理方式取得我們所需的部分，所以我們可以很容易大量取得語

表 5.28: 對照組，P、R 和 F 在原文中表示到小數後第二位

介系詞	個數	分布線 (%)	P(%)	R(%)	F(%)
of	7485	38.18	88	78	83
to	4841	24.69	78	87	82
in	4278	21.82	75	78	77
on	1483	7.56	66	65	65
with	1520	7.75	73	69	70

表 5.29: 混淆矩陣: 華爾街日報與紐約時報

		決策答案										
		of	in	for	to	on	with	from	at	as	by	about
實際答案	of	**1801**	283	72	66	48	31	28	31	17	2	11
	in	421	**1101**	74	49	58	28	16	31	14	2	7
	for	**355**	188	243	42	26	26	6	21	4	1	4
	to	275	140	65	**283**	42	27	6	34	6	9	5
	on	216	124	36	24	**315**	13	12	14	7	0	7
	with	**267**	95	34	21	20	94	8	16	10	3	4
	from	**125**	117	37	28	20	7	56	17	1	4	1
	at	**111**	58	28	21	18	4	8	100	2	7	2
	as	**123**	30	10	6	7	7	2	7	45	2	0
	by	**71**	31	13	13	7	5	3	4	2	11	2
	about	**49**	14	5	7	3	5	0	0	1	0	28

料做實驗，因此在足夠大量的語料下，我們相信即使不能完全與其它方法相比較，但仍然可以做參考，知道目前研究的成果。目前的成果與 De Felice 和 Pulman[6] 的成果是有一小段差距，參考表 5.28，但本研究中所考慮的資訊的僅包三個中心詞，而 De Felice 和 Pulman 的研究則是考慮了上下文的語義而視窗大小設為 6，這也顯示我們的研究還有進步的空間。

第 6 章　結論

我們將在這節總結截至目前為止我們工作的概況以及成果，並檢討研究不足或有缺陷的部分，並希望這些問題在未來能夠得到改進，以增進模型成效。

本研究探討有關英文介系詞的二個議題：介系詞片語定位與介系詞推薦。前者一直是過去自然語言處理的重要議題，用以解決歧義的問題。後者則是偏向於文本校對，用以幫助非英文母語的學習者選擇適當的介系詞。本研究以動詞片語中的四個中心詞為出發點，透過 WordNet 查詢中心詞語義，由於 WordNet 是以階層式的架構來描述語義，我們相信在大量的案例中，各個案例必定在某種語義層次上是相似，進而可以歸納出通用規則來使我們的模型可以正確分類。在介系詞片語定位問題裡，我們根據介系詞的不同將語料初步分類，再選出日常中最常用且具有挑戰性的介系詞來做為實驗的目標，並逐一為這些介系詞製作量身打造的訓練模型。在介系詞推薦問題裡，我們則挑選數量多且彼此詞頻較接近的介系詞建構模型。而在建立模型遇到最大的挑戰在於特徵量，特徵是由詞彙透過 WordNet 查詢的同義詞集，所以可能產生的特徵數量會非常大，換句話說，在這個問題裡面特徵選擇是必須的。另一方面，因為同義詞集間具有階層關係，所以我們設計了一套階層式的特徵選擇方法。最後，我們再將所有訓練的模型彙整成高階分類器，集合眾人智慧成最終模型，用以決策我們的問題。

6.1 討論

實驗的成果根據我們的實驗結果可以分成二大類討論：（一）介系詞片語定位的問題，較困難處是在語料的取得，很難找到有專家可以大量地為我們標記介系詞片語的定位，而且本研究對於現有的語料又需要再依介系詞分類，這使得我們的語料變得非常稀少，也造成我們的實驗成果容易受到語料量的影響。實驗比較部分，若和最大熵值法比較，在只考慮四個中心詞的情況下，我們的方法遠勝於最大熵值法；若與 Stanford 剖析器相比較，兩種方法各有優缺點，Stanford 剖析器考慮的是整個句子，而我們所考慮的則是四個中心詞。我們所考慮四個中心詞在我們研究是假設已給定的情況，被視為是前處理的一部分，實驗結果則是二者差不多；如果我們比較利用同樣的方法訓練的單一模型與混合資料訓練的結果，二者效果也是差不多，針對單一模型的結果是我們可以較明確地知道究竟哪些同義詞集影響介系詞，再且混合資料實驗整體的語料量遠較單一實驗大，因此混合資料模型可以看到較多的案例，我們相信若是能夠再提升單一模型的語料數量必定能有改善的空間；從我們選擇較高度抽象化的特徵所建構之模型，也可以發現效果不差，但是若能利用 WordNet 做階層式的選擇效果更佳。（二）介系詞推薦問題雖說因為難以取得與其它研究方法中相同的語料庫作相互比較的客觀基準，但是本身語料的收集與取得是較容易的。因此在容易取得大量的語料下，我們假設所收集的語料可以包含各種可能情況，所以我們還是可以參考其它文獻方法所做的結果，進一步知道目前方法成效。實驗結果顯示我們的方法與現行較好的方法（De Felice 和 Pulman[6]）成效是有一段差距，然而我們使用的資訊是較少的，因此還是有可以進步的空間。而在研究中，我們發現我們的模型在決策時，多會受到各類別語料量的影響，目前的實驗成果只顯示混淆矩陣易偏好語料量較多的類別，那在未來我們希望能透過更多實驗探究可能的原因。

在整個研究中，我們出發點是由動詞片語的四個中心詞出發，這與 Atterer 和

Schütze[2] 所指出中心詞在現實中不是憑空而來是相違背的。雖然假設已知中心詞是不實際的，但是透過前人的抽取中心詞研究（Collins[4] 與 Stanford 剖析器），我們的模型也可以有不錯的表現。我們前處理從文章到中心詞抽取的過程固然在開始研究前就喪失部分精準度，然而仍希望能站在前人的肩膀之上，勇敢的踏出一步。

由於我們的方法是建立在將中心詞抽象化之上，因此當中心詞的詞義抽象程度越高時，越容易發生有相同特徵向量但實際答案卻不同的情況，這種情況被稱為碰撞。可以觀察到碰撞的情況有兩種：(一) 原始資料裡面有存在的碰撞；(二) 高度抽象化過程中所造成的碰撞。而可能產生碰撞的原因有：（一）資料可能本身有錯；（二）與實驗假設有關，因為我們假設一個介系詞片語只能修飾動詞或是名詞，但是實際上一個介系詞是有可能兩者都可以修飾的。例如說在 1.1節句 1關於湯匙的例子。在這裡我們只處理第一種碰撞的狀況，並視為雜訊處理。第二種碰撞的狀況我們目前並沒有額外處理，而是讓模型從中學習並取得平衡點。整體來說，我們的研究沒有針對碰撞去作特別的研究與處理，而這或許是未來可以應用重新定位技術（reattach）再改進之處。

實驗較不足處是未能與更多現有之研究方法做全面性的比較。困難點再於時間和人力有限的情況下，我們很難還原其他研究當時實驗的環境，且這些研究較少有人包裝成函式庫（library），因此無法一一比較過往實驗成效。除了與過往研究做比較外，目前有不少的實驗也會與人類本身做比較，研究目的最終是用之於人類，因此若能與人類相比較，也是可以確認本研究成效的方法之一。

6.2 未來工作

本研究的未來工作除了希望能有機會放寬討論中提及的限制之外，研究方面，也希望未來能再更深入地針對本研究的發現的問題再改良模型以提升成果；另外，在本次研究中透過 WordNet 做為抽象化語義的工具，但 WordNet 對詞彙的描述分類過於細，因此也

希望未來能透過 SUMO[1]改善。應用方面，介系詞片語定位希望未來能應用於機器翻譯用途上，對電腦方有一實際直接的幫助；介系詞推薦則是希望有一套完整且全面性的系統可以幫助使用者做文本校對，以達到語言學習目的。並期望二者未來都可以輔助人類解決問題。

[1] http://www.ontologyportal.org/index.html

參考文獻

[1] Eneko Agirre, Timothy Baldwin, and David Martinez. Improving Parsing and PP Attachment Performance with Sense Information. In *46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 2008.

[2] Michaela Atterer and Hinrich Schütze. Prepositional Phrase Attachment without Oracles. *Computational Linguistics*, 33(4):469–476, 2007.

[3] Timothy Baldwin, Valia Kordoni, and Aline Villavicencio. Prepositions in Applications: A Survey and Introduction to the Special Issue. *Computational Linguistics*, 35(2):119–149, 2009.

[4] Michael John Collins. *Head-driven Statistical Models for Natural Language Parsing*. PhD thesis, 1999.

[5] Gregory F. Coppola, Alexandra Birch, Tejaswini Deoskar, and Mark Steedman. Simple Semi-supervised Learning for Prepositional Phrase Attachment. In *Proceedings of the 12th International Conference on Parsing Technologies*, pages 129–139, 2011.

[6] Rachele De Felice and Stephen G. Pulman. Automatically Acquiring Models of Preposition Use. In *Proceedings of the Fourth ACL-SIGSEM Workshop on Prepositions*, pages 45–50, 2007.

[7] Rachele De Felice and Stephen G. Pulman. A Classifier-based Approach to Preposition and Determiner Error Correction in L2 English. In *Proceedings of the 22nd International Conference on Computational Linguistics*, volume 1, pages 169–176, 2008.

[8] Michael Gamon, Jianfeng Gao, Chris Brockett, and Re Klementiev. Using Contextual Speller Techniques and Language Modeling for ESL Error Correction. In *Proceedings of Joint Conference on Natural Language Processing 2008*, pages 449–456, 2008.

[9] Na-Rae Han, Joel Tetreault, Soo-Hwa Lee, and Jin-Young Ha. Using an Error-annotated Learner Corpus to Develop an ESL/EFL Error Correction System. In *Proceedings of the Seventh conference on International Language Resources and Evaluation*, 2010.

[10] Donald Hindle and Mats Rooth. Structural Ambiguity and Lexical Relations. *Computational Linguistics*, 19(1):103–120, 1993.

[11] Dirk Hovy, Stephen Tratz, and Eduard Hovy. What's in a Preposition?: Dimensions of Sense Disambiguation for an Interesting Word Class. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 454–462, 2010.

[12] Dan Klein and Christopher D. Manning. Fast Exact Inference with a Factored Model for Natural Language Parsing. In *Advances in Neural Information Processing Systems*, volume 15, pages 3–10, 2003.

[13] Claudia Leacock, Michael Gamon, and Chris Brockett. User Input and Interactions on Microsoft Research ESL Assistant. In *Proceedings of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 73–81, 2009.

[14] Ken C. Litkowski and Orin Hargraves. Coverage and Inheritance in The Preposition Project. In *Proceedings of the Third ACL-SIGSEM Workshop on Prepositions*, pages 37–44, 2006.

[15] Chao-Lin Liu, Jing-Shin Chang, and Keh-Yih Su. The Semantic Score Approach to the Disambiguation of PP Attachment Problem. In *Proceedings of the ROC Computational Linguistics Conference III*, pages 253–270, 1990.

[16] Tom O'Hara and Janyce Wiebe. Exploiting Semantic Role Resources for Preposition Disambiguation. *Computational Linguistics*, 35(2):151–184, 2009.

[17] Marian Olteanu and Dan Moldovan. PP-Attachment Disambiguation Using Large Context. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 273–280, 2005.

[18] Patrick Pantel and Dekang Lin. An Unsupervised Approach to Prepositional Phrase Attachment Using Contextually Similar Words. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 101–108, 2000.

[19] Li Quan, Oleksandr Kolomiyets, and Marie-Francine Moens. KU Leuven at HOO-2012: A Hybrid Approach to Detection and Correction of Determiner and Preposition Errors in Non-native English Text. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 263–271, 2012.

[20] Adwait Ratnaparkhi, Jeff Reynar, and Salim Roukos. A Maximum Entropy Model for Prepositional Phrase Attachment. In *Proceedings of the Workshop on Human Language Technology*, pages 250–255, 1994.

[21] Jiri Stetina and Makoto Nagao. Corpus Based PP Attachment Ambiguity Resolution with a Semantic Dictionary. In *Proceedings of the Fifth Workshop on Very Large Corpora*, pages 66–80, 1997.

[22] Joel R. Tetreault and Martin Chodorow. The Ups and Downs of Preposition Error Detection in ESL Writing. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, pages 865–872, 2008.

[23] Stephen Tratz and Dirk Hovy. Disambiguation of Preposition Sense Using Linguistically Motivated Features. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Student Research Workshop and Doctoral Consortium*, pages 96–100, 2009.

[24] Martin Volk. Combining Unsupervised and Supervised Methods for PP Attachment Disambiguation. In *Proceedings of the 19th International Conference on Computational Linguistics*, volume 1, pages 1–7, 2002.

[25] Jian-Cheng Wu, Joseph Chang, Yi-Chun Chen, Shih-Ting Huang, Mei-Hua Chen, and Jason S. Chang. Helping Our Own: NTHU NLPLAB System Description. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 295–301, 2012.

附錄 I　同義詞集種類

File Number	Name	Contents
00	adj.all	all adjective clusters
01	adj.pert	relational adjectives (pertainyms)
02	adv.all	all adverbs
03	noun.Tops	unique beginner for nouns
04	noun.act	nouns denoting acts or actions
05	noun.animal	nouns denoting animals
06	noun.artifact	nouns denoting man-made objects
07	noun.attribute	nouns denoting attributes of people and objects
08	noun.body	nouns denoting body parts
09	noun.cognition	nouns denoting cognitive processes and contents
10	noun.communication	nouns denoting communicative processes and contents
11	noun.event	nouns denoting natural events
12	noun.feeling	nouns denoting feelings and emotions
13	noun.food	nouns denoting foods and drinks
14	noun.group	nouns denoting groupings of people or objects
15	noun.location	nouns denoting spatial position
16	noun.motive	nouns denoting goals

17	noun.object	nouns denoting natural objects (not man-made)
18	noun.person	nouns denoting people
19	noun.phenomenon	nouns denoting natural phenomena
20	noun.plant	nouns denoting plants
21	noun.possession	nouns denoting possession and transfer of possession
22	noun.process	nouns denoting natural processes
23	noun.quantity	nouns denoting quantities and units of measure
24	noun.relation	nouns denoting relations between people or things or ideas
25	noun.shape	nouns denoting two and three dimensional shapes
26	noun.state	nouns denoting stable states of affairs
27	noun.substance	nouns denoting substances
28	noun.time	nouns denoting time and temporal relations
29	verb.body	verbs of grooming, dressing and bodily care
30	verb.change	verbs of size, temperature change, intensifying, etc.
31	verb.cognition	verbs of thinking, judging, analyzing, doubting
32	verb.communication	verbs of telling, asking, ordering, singing
33	verb.competition	verbs of fighting, athletic activities
34	verb.consumption	verbs of eating and drinking
35	verb.contact	verbs of touching, hitting, tying, digging
36	verb.creation	verbs of sewing, baking, painting, performing
37	verb.emotion	verbs of feeling
38	verb.motion	verbs of walking, flying, swimming
39	verb.perception	verbs of seeing, hearing, feeling
40	verb.possession	verbs of buying, selling, owning

41	verb.social	verbs of political and social activities and events
42	verb.stative	verbs of being, having, spatial relations
43	verb.weather	verbs of raining, snowing, thawing, thundering
44	adj.ppl	participial adjectives