簡介AI文字處理技術與應用

淡江大學資訊管理學系
魏世杰副教授
sekewei@mail.tku.edu.tw

大綱

- 語料前處理
- 文字的向量表示法
- 文字的比對法
- 文字檢索的技術與應用
- 文字探勘的技術與應用
- 文字生成的技術與應用
- 現況與未來

語料前處理

- 語料集(Corpus)
  - 單語系(Monolingual)
  - 雙語系(Bilingual): 平行語料(Parallel Corpus)
  - 網頁蒐集工具: scrapy, Beautiful Soup
- 單語化工具 → 系統字彙(Lexicon)
  - 英文: nltk
  - 中文: Jieba, MMSeg, CKIP斷詞系統
  - 日文: MeCab, ChaSen, Juman++
- 詞性 (POS, Parts of Speech)

文字的向量表示法

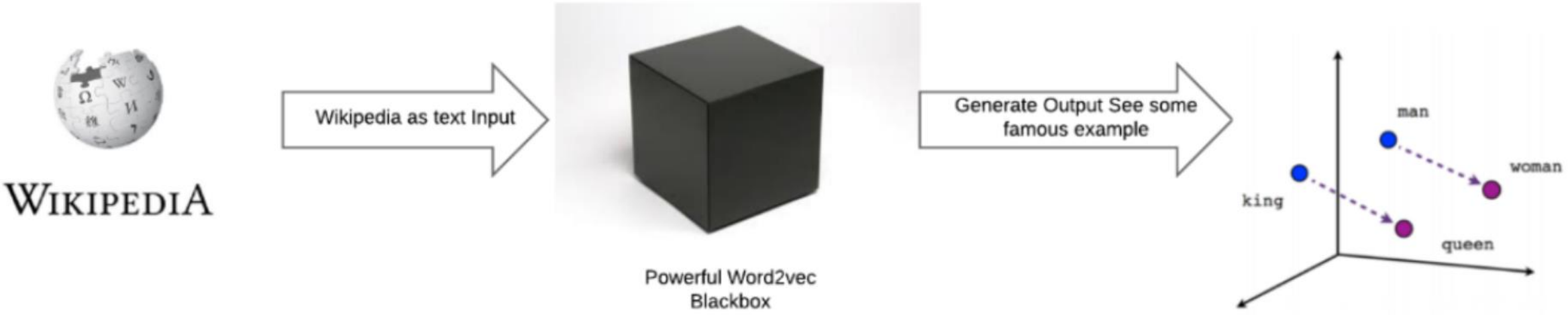
- 詞向量
  - 單熱向量 (One Hot Vector): 稀疏向量，每個維度代表某詞有無。只有代表該詞維度為1，其餘維度皆為0
  - 內嵌向量 (Embedding Vector): 稠密向量，每個維度代表某綜合成份意義。很少維度值為0
- 文件向量
  - 布林向量
  - TF-IDF向量
  - 內嵌向量

Table 4.6 A small document collection: six documents over 10 terms.

d	Document D_d
1	Pease porridge hot, pease porridge cold,
2	Pease porridge in the pot,
3	Nine days old.
4	In the pot cold, in the pot hot,
5	Pease porridge, pease porridge,
6	Eat the lot.

(Witten, 1999)

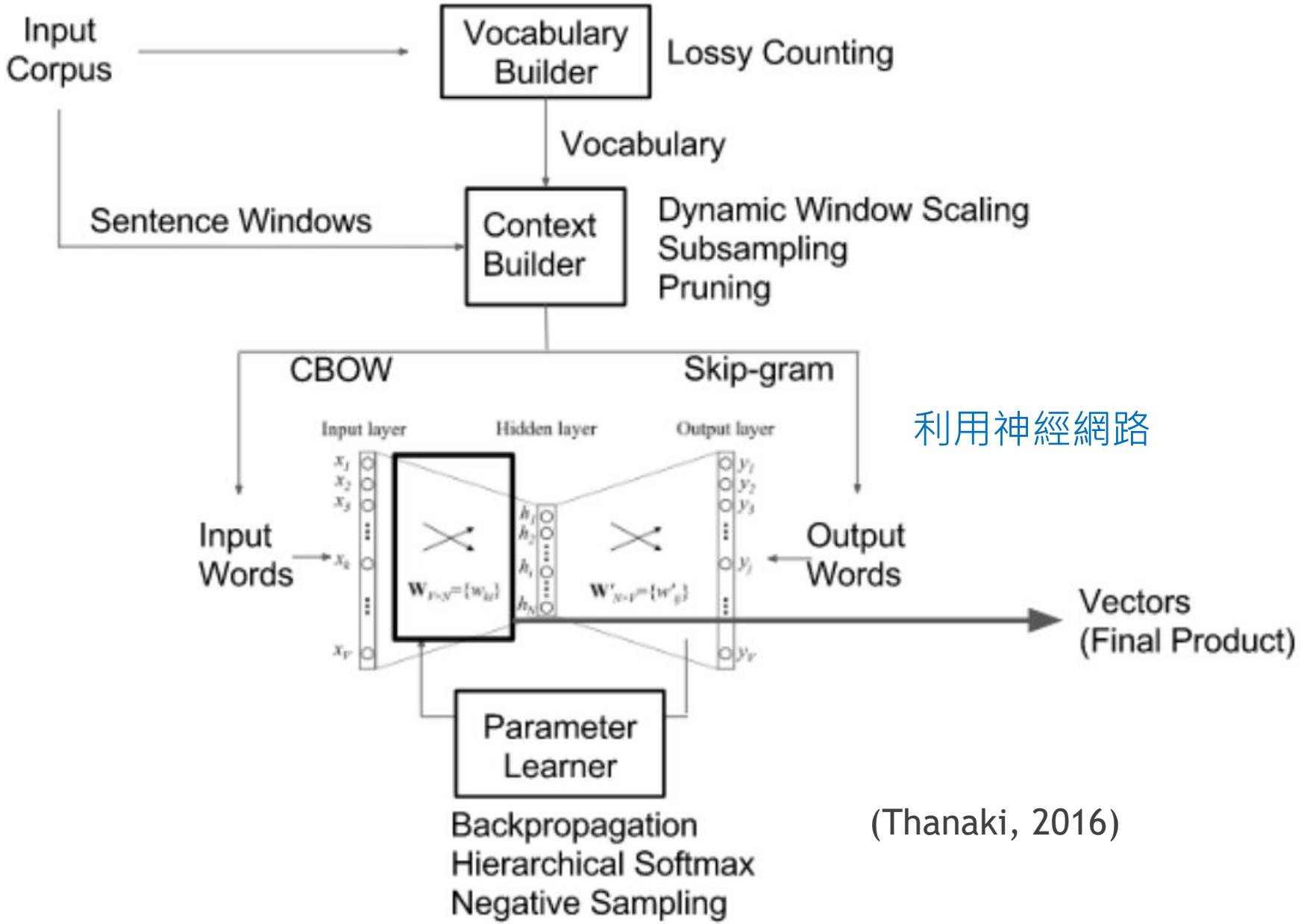
如何訓練內嵌詞向量(Embedding Word Vector)



(Thanaki, 2016)

- 以下3種詞向量表示法皆提供預先訓練好的詞向量供下載使用
- Google Word2vec: Skip Gram (SG) or Continuous Bag of Words (CBoW)
- Stanford Glove
- Facebook Fasttext

Word2Vec訓練法

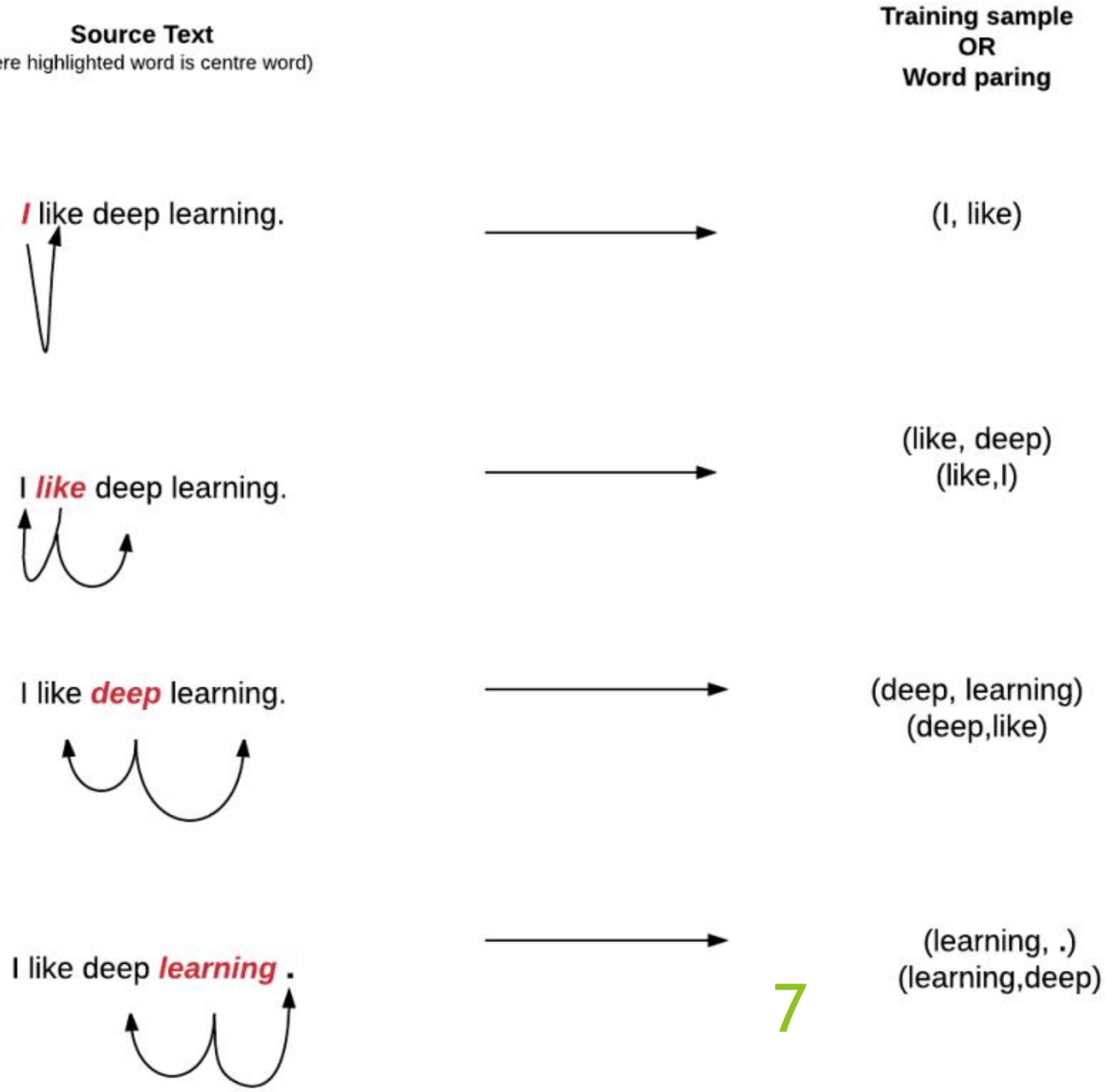


利用神經網路

(Thanaki, 2016)

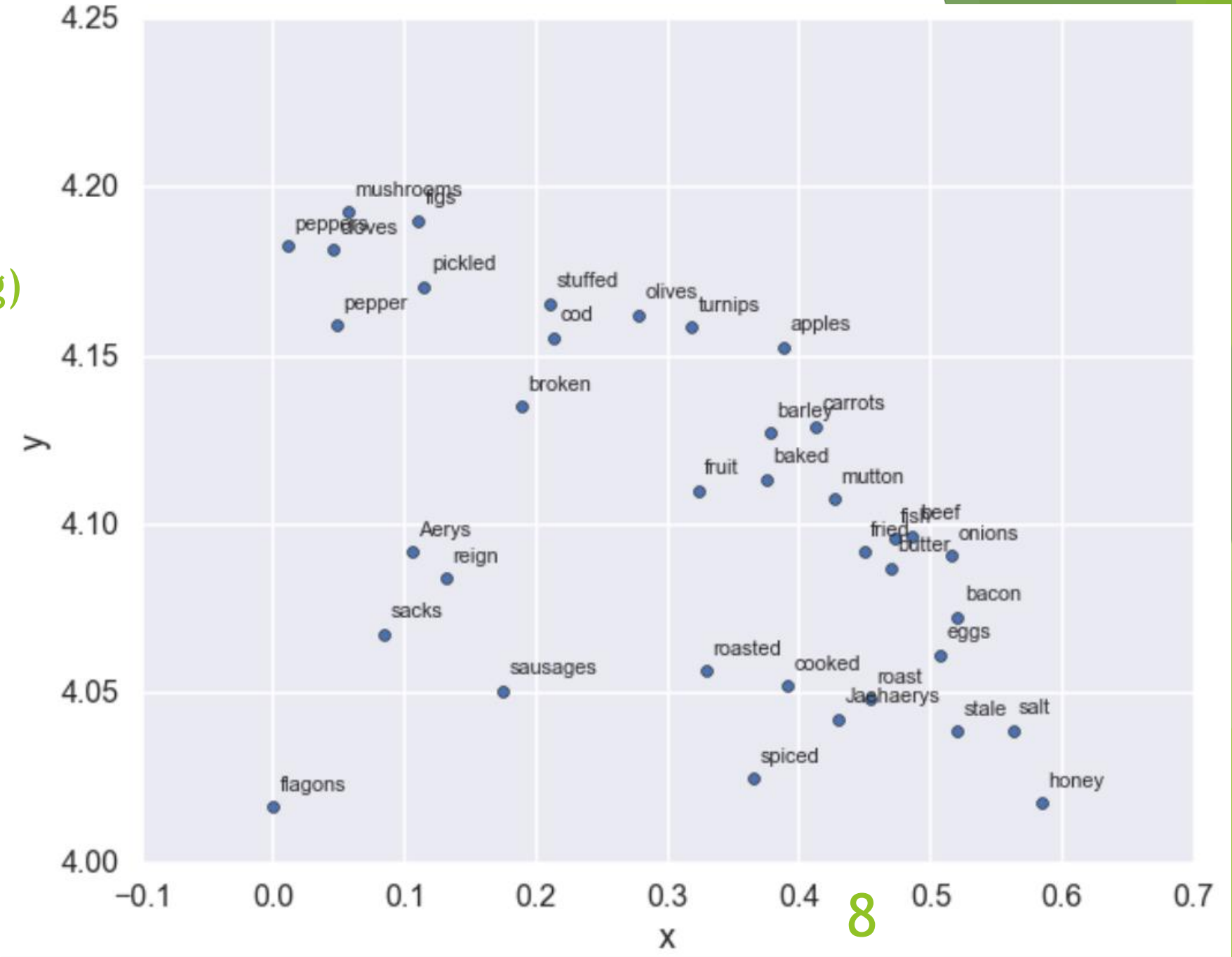
Word2Vec訓練資料

假設視窗大小為前後1個字



(Thanaki, 2016)

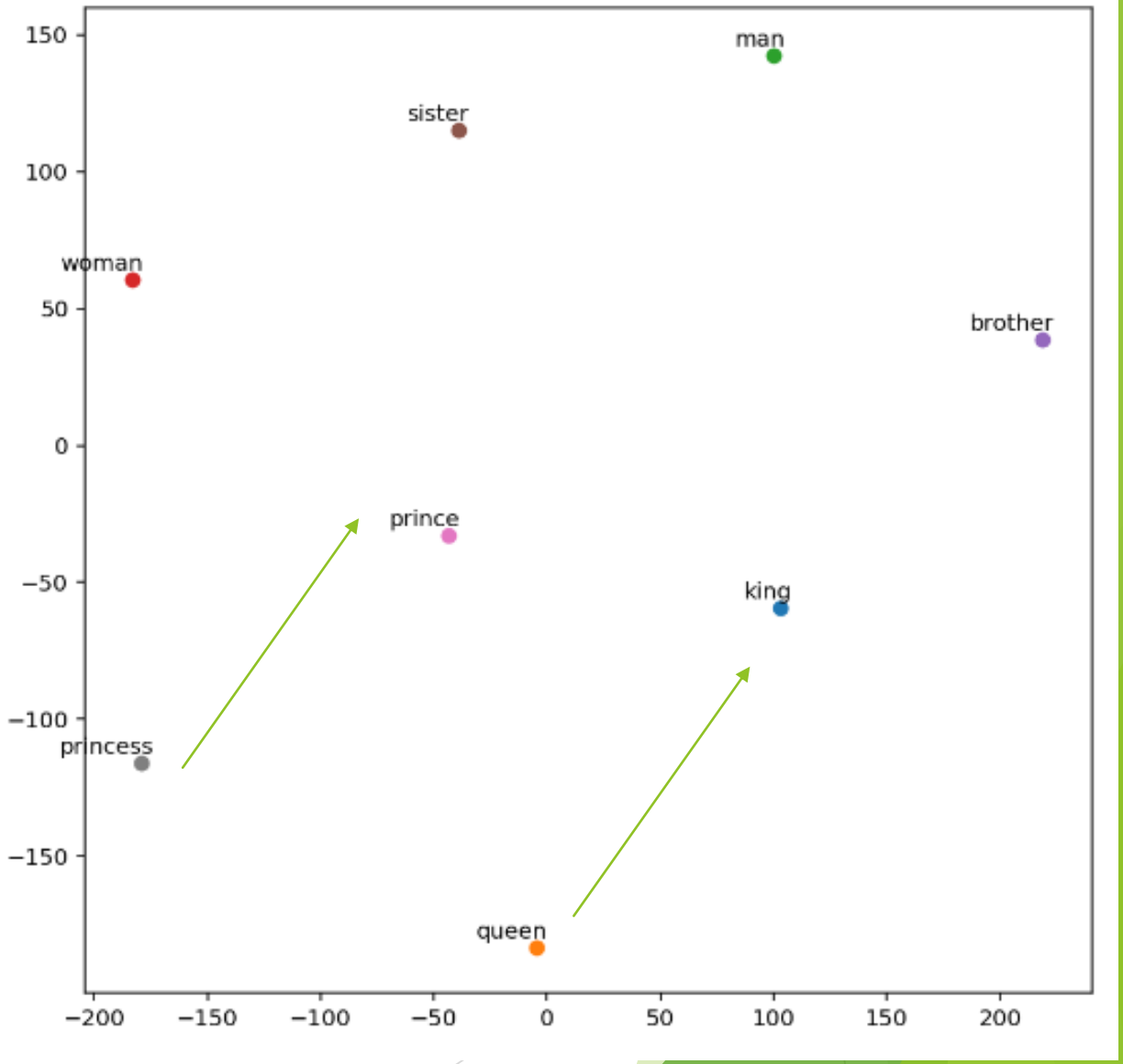
詞向量訓練結果1: 群聚性(Clustering)



(Thanaki, 2016)

詞向量訓練結果2:
類推性(Analogy)

- king – queen + princess = ?
- computer_programmer – man + woman = ?
- doctor – father + mother = ?
- 發現偏見與去除偏見 (Jurafsky, 2018)



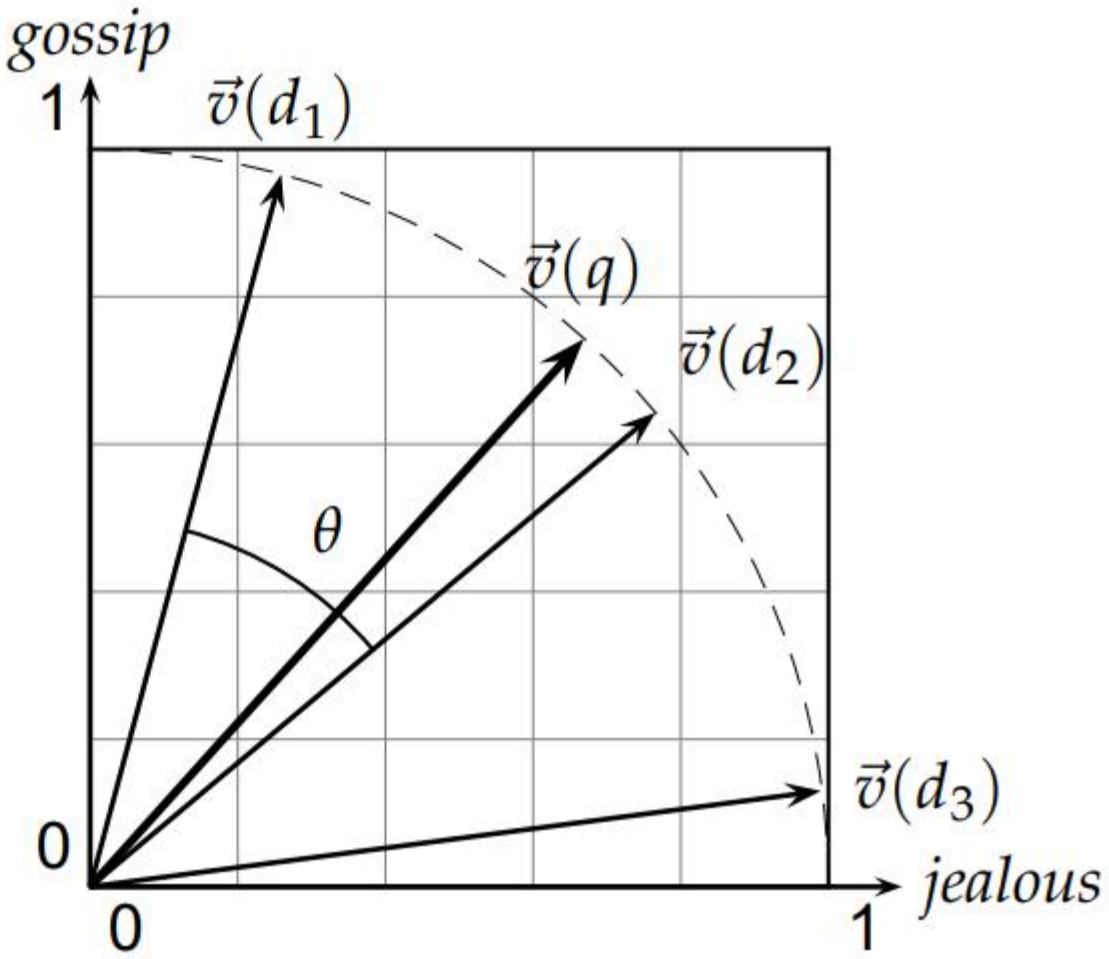
文字的比對法

詞向量或文件向量可利用向量內積，
計算向量相似度，又稱餘弦相似度

兩向量夾角$\theta\downarrow$，內積或餘弦值↑
表示兩向量相似度↑

$$\text{sim}(d_1, d_2) = \frac{\vec{V}(d_1) \cdot \vec{V}(d_2)}{|\vec{V}(d_1)||\vec{V}(d_2)|}$$

$$= \cos(\theta)$$



(Manning, 2008)

► **Figure 6.10** Cosine similarity illustrated. $\text{sim}(d_1, d_2) = \cos\theta$.

布林向量表示法

1表示詞出現
0表示詞未出現

- 文章向量
- 查詢向量
- 內積相似度當作排名指標

(Witten, 1999)

Table 4.7 Vectors for inner product calculation: (a) document vectors; (b) query vectors.

		Document vectors $\langle w_{d,t} \rangle$									
(a)	*d*	*col*	*day*	*eat*	*hot*	*lot*	*nin*	*old*	*pea*	*por*	*pot*
	1	1	0	0	1	0	0	0	1	1	0
	2	0	0	0	0	0	0	0	1	1	1
	3	0	1	0	0	0	1	1	0	0	0
	4	1	0	0	1	0	0	0	0	0	1
	5	0	0	0	0	0	0	0	1	1	0
	6	0	0	1	0	1	0	0	0	0	0
(b)	*eat*	0	0	1	0	0	0	0	0	0	0
	hot porridge	0	0	0	1	0	0	0	0	1	0

where the operation $\cdot$ is inner product multiplication. The *inner product* of two n-vectors $X = \langle x_i \rangle$ and $Y = \langle y_i \rangle$ is defined to be

$$X \cdot Y = \sum_{i=1}^{n} x_i y_i.$$

For example,

$$M(\textit{hot porridge}, D_1) = (0,0,0,1,0,0,0,0,1,0) \cdot (1,0,0,1,0,0,0,1,1,0) = 2.$$

TF-IDF向量表示法

TF文件內詞頻，表示詞代表文件能力

IDF詞出現文件數倒數，表示詞區別文件能力

兩值相乘愈大表示

詞愈能在資料集區別該文件

- 文章向量
- 查詢向量
- 內積相似度當作排名指標

(Witten, 1999)

Table 4.8 Application of the cosine measure: (a) term frequencies $f_{d,t}$ and document weights; (b) cosine similarities for queries.

(a)	d	Document vectors $\langle w_{d,t} \rangle$										W_d
		col	*day*	*eat*	*hot*	*lot*	*nin*	*old*	*pea*	*por*	*pot*	
	1	1.0	0.0	0.0	1.0	0.0	0.0	0.0	1.7	1.7	0.0	2.78
	2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.73
	3	0.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	1.73
	4	1.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	1.7	2.21
	5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.7	1.7	0.0	2.39
	6	0.0	0.0	1.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	1.41
	f_t	2	1	1	2	1	1	1	3	3	2	
	w_t	1.39	1.95	1.95	1.39	1.95	1.95	1.95	1.10	1.10	1.39	

(b)	d	Query			
		eat	*porridge*	*hot porridge*	*eat nine day old porridge*
		$W_q = 1.95$	$W_q = 1.10$	$W_q = 1.77$	$W_q = 3.55$
	1	0.00	0.61	0.66	0.19
	2	0.00	0.58	0.36	0.18
	3	0.00	0.00	0.00	0.63
	4	0.00	0.00	0.36	0.00
	5	0.00	0.71	0.44	0.22
	6	0.71	0.00	0.00	0.39
	Top	6	5	1	3

評估指標

- 召回率=回傳相關文件數/相關文件數
- 精確率=回傳相關文件數/回傳文件數
- 召回率-精確率曲線圖

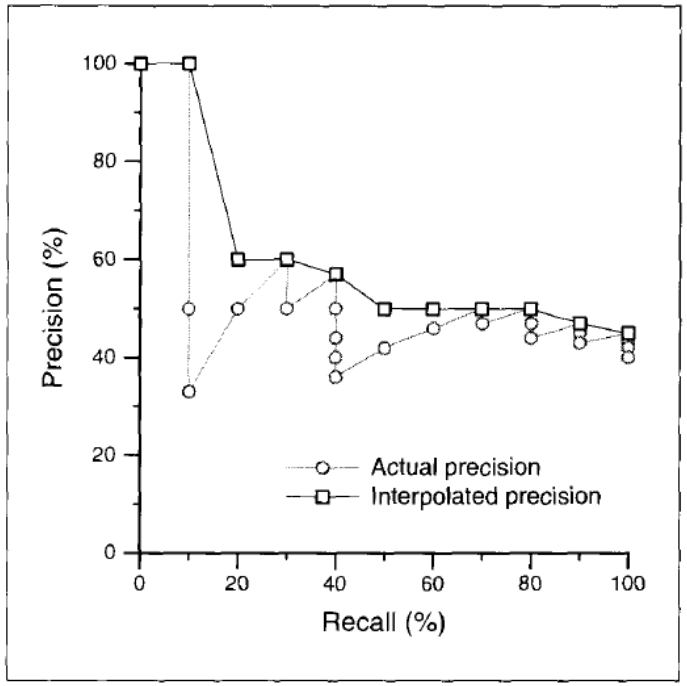


Figure 4.10 Recall-precision curve for ranking of Table 4.9.

Table 4.9 Example showing calculation of recall and precision: (a) rank order; (b) calculating effectiveness.

(a) r		R_r	P_r
1	R	10%	100%
2	—	10%	50%
3	—	10%	33%
4	R	20%	50%
5	R	30%	60%
6	—	30%	50%
7	R	40%	57%
8	—	40%	50%
9	—	40%	44%
10	—	40%	40%
11	—	40%	36%
12	R	50%	42%
13	R	60%	46%
14	R	70%	50%
15	—	70%	47%
16	R	80%	50%
17	—	80%	47%
18	—	80%	44%
19	R	90%	47%
20	—	90%	45%
21	—	90%	43%
22	R	100%	45%
23	—	100%	43%
24	—	100%	42%
25	—	100%	40%

(b) Recall	Precision	Interpolated precision
0%	—	100%
10%	100%	100%
20%	50%	60%
30%	60%	60%
40%	57%	57%
50%	42%	50%
60%	46%	50%
70%	50%	50%
80%	50%	50%
90%	47%	47%
100%	45%	45%
3-point average		53%
11-point average		61%

(Witten, 1999)

文字檢索的技術與應用

- Text Retrieval 文字檢索
  - Tokenization 斷詞
  - Indexing 索引
  - Vector Representation 向量表示法
    - Document Vector v.s. Query Vector
    - Binary v.s. TF-IDF
  - Similarity Metric 相似度指標
    - Cosine Measure 餘弦值
  - Document Ranking 文件排名
  - Evaluation Metric 評估指標
    - Precision, Recall, BLEU

- Cf: Manning et al. (2008), Introduction to Information Retrieval

文字探勘的技術與應用

- 文件摘要: 依句子重要度及多樣性決定留下哪些句子
- 文字雲: 依詞重要度及多樣性決定留下哪些詞及其顯示的大小及位置
- 文件主題分析: 依文件內不同詞的出現頻度自動為文件作主題成份分析
- 商品推薦: 累積客戶喜好，和商品計算相似度
- 垃圾分析/情感分析/輿情分析:
  - 利用人工或機器學習判定正負面評價或類別
    - Input: 文件向量
    - Model: NaiveBayes/SVM/XGBoost/RandomForest/…
    - Output: 正負面評價或類別

產生文字雲之前的斷詞範例

```
documents = tokenizedDocument(textData);
documentsRaw = documents;

documents(1:10)
```

(Matlab, 2018)

```
ans =
  10×1 tokenizedDocument:

     5 tokens: 吾輩 は 猫 で ある
     2 tokens: 夏目 漱石
     0 tokens:
     1 tokens: —
    11 tokens: 吾輩 は 猫 で ある 。 名前 は まだ 無い 。
   264 tokens: どこ で 生れ た か とんと 見当 が つか ぬ 。 何 でも 薄暗い じめじめ し た 所 で ニャーニャー 泣い て いた
   100 tokens: この 書生 の 掌 の 裏 で しばらく は よい 心持 に 坐っ て おっ た が 、 しばらく する と 非常 な 速力 で 運
    92 tokens: ふと 気 が 付い て 見る と 書生 は い ない 。 たくさん おっ た 兄弟 が 一疋 も 見え ぬ 。 肝心 の 母親 さえ
   693 tokens: ようやく の 思い で 笹原 を 這い出す と 向う に 大きな 池 が ある 。 吾輩 は 池 の 前 に 坐っ て どう した
   276 tokens: 吾輩 の 主人 は 滅多 に 吾輩 と 顔 を 合せる 事 が ない 。 職業 は 教師 だ そう だ 。 学校 から 帰る と 終
```

MeCab斷詞範例

```
tdetails = tokenDetails(documents);
size(tdetails)
```

```
ans = 1×2
       80472           7
```

```
head(tdetails)
```

(Matlab, 2018)

```
ans = 8×7 table
```

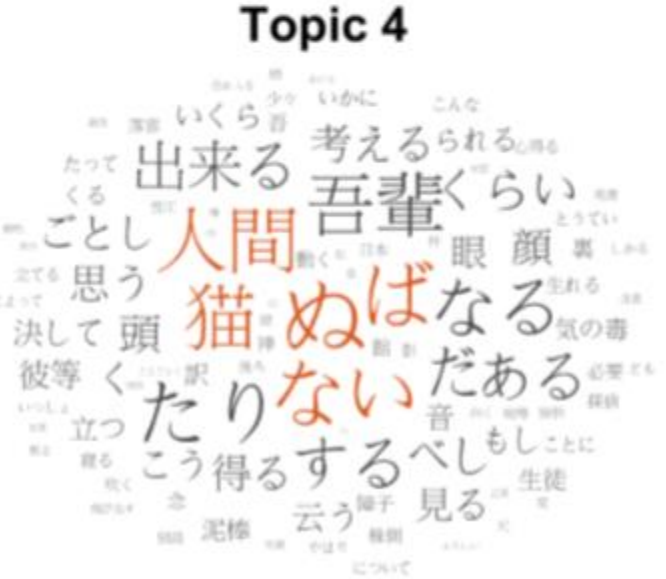
...

	Token	DocumentNumber	LineNumber	Type	Language	PartOfSpeech
1	"吾輩"	1	1	letters	ja	pronoun
2	"猫"	1	1	letters	ja	noun
3	"夏目"	2	1	letters	ja	proper-noun
4	"漱石"	2	1	letters	ja	proper-noun
5	"吾輩"	3	1	letters	ja	pronoun
6	"猫"	3	1	letters	ja	noun
7	"まだ"	3	1	letters	ja	adverb
8	"無い"	3	1	letters	ja	adjective

文字雲範例

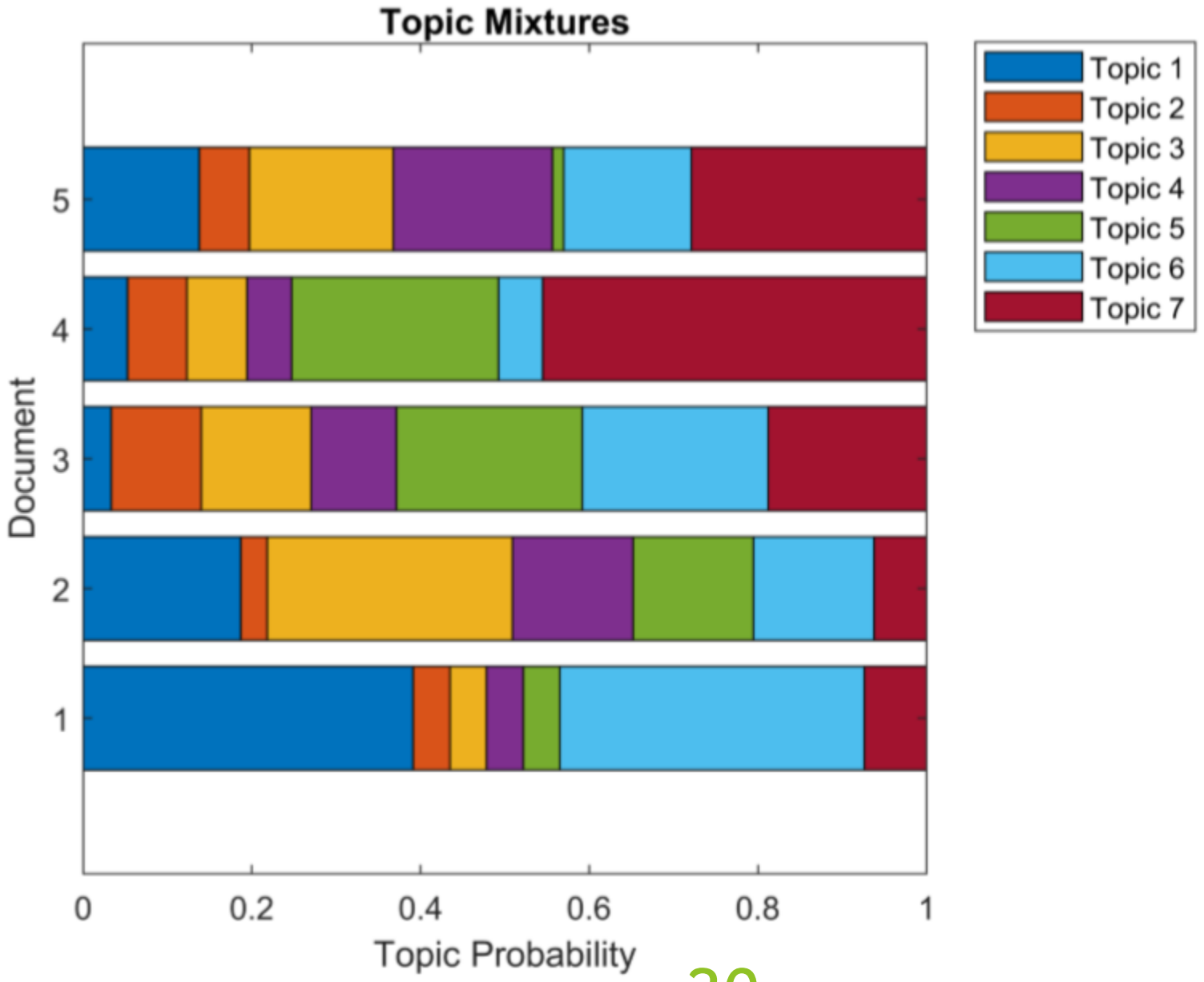
(Matlab, 2018)

LDA
(latent Dirichlet allocation)
文件主題分析
範例結果1



(Matlab, 2018)

LDA
(latent Dirichlet allocation)
文件主題分析
範例結果2



(Matlab, 2018)

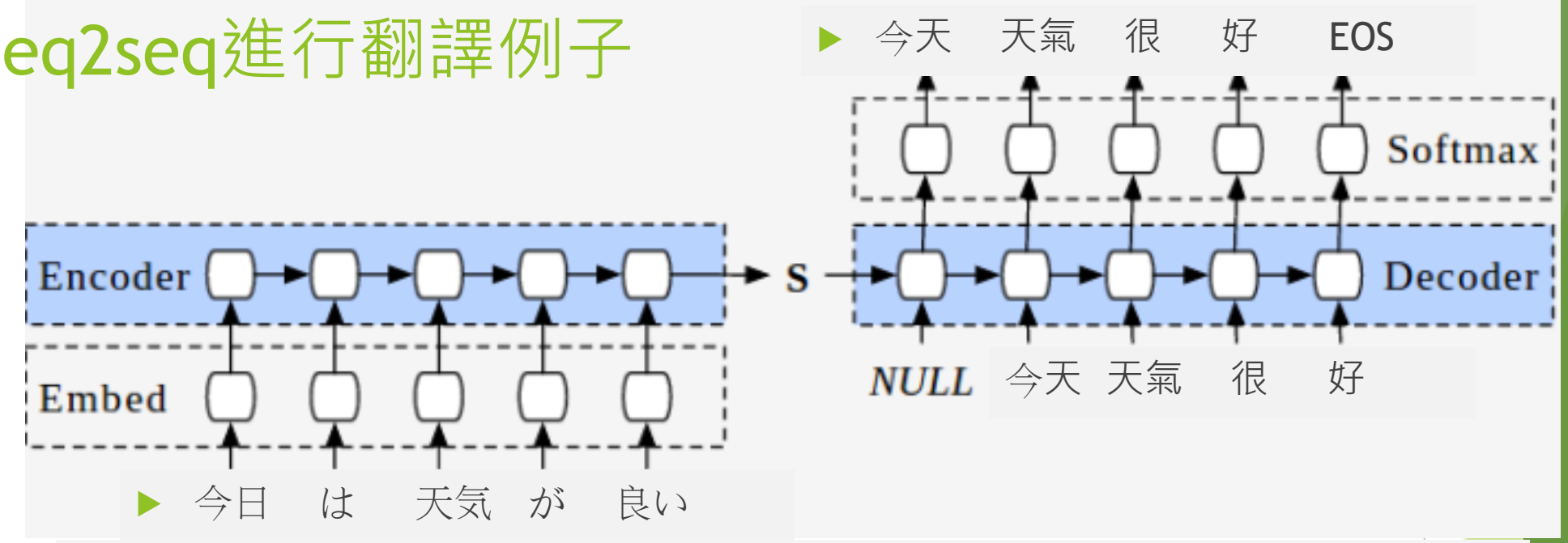
文字生成的技術與應用

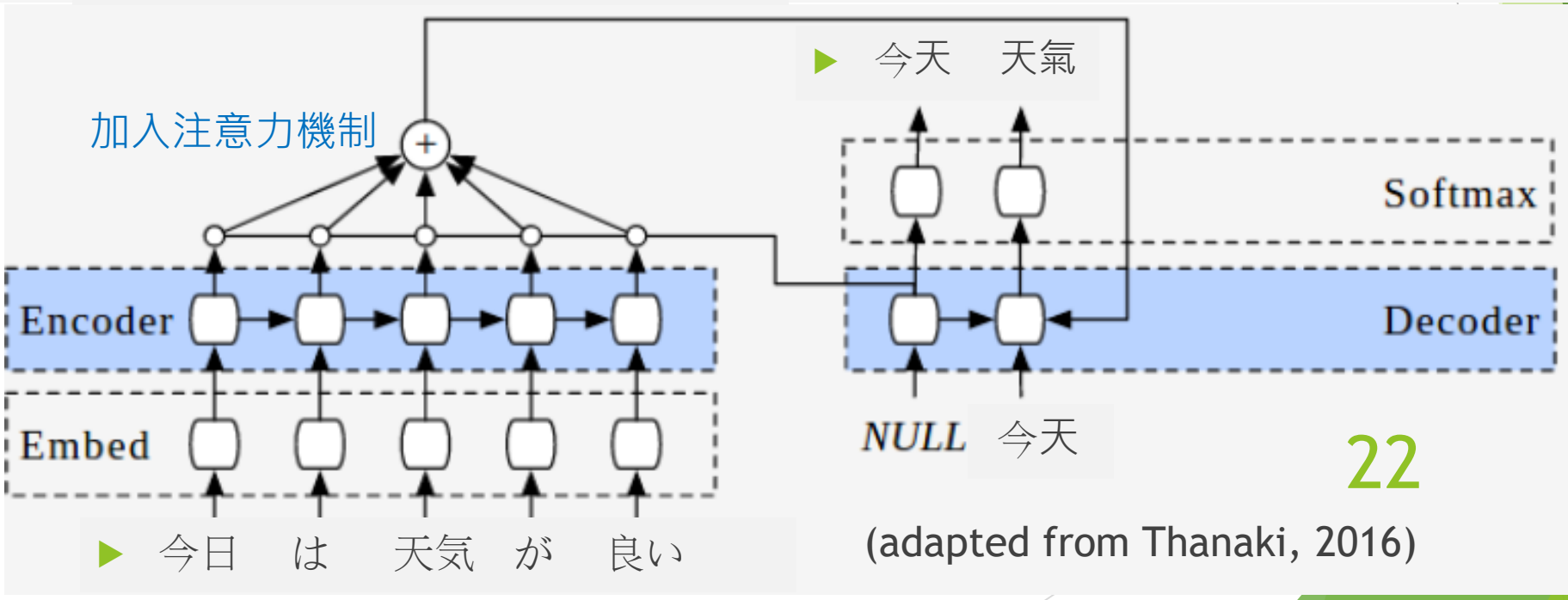
- 文字生成可視為馬可夫模型(Markov Model)的應用，如何由前文推出下一個詞

  $Pr(w_i \mid w_1, w_2, w_3, \ldots, w_{i-1})$

- 神經網路(OpenNMT) vs 統計式(Moses) vs 規則式(SYSTRAN)
- 深度學習
  - Recurrent Neural Network (RNN)
    - Long and Short Term Memory (LSTM)
    - Gated Recurrent Unit (GRU)
  - Convolutional Neural Network (CNN)
  - Encoder/Decoder-based Sequence to Sequence (seq2seq) Generator
- 代表性: Google Translate
  - https://www.youtube.com/watch?v=M8ZfR4iEwig

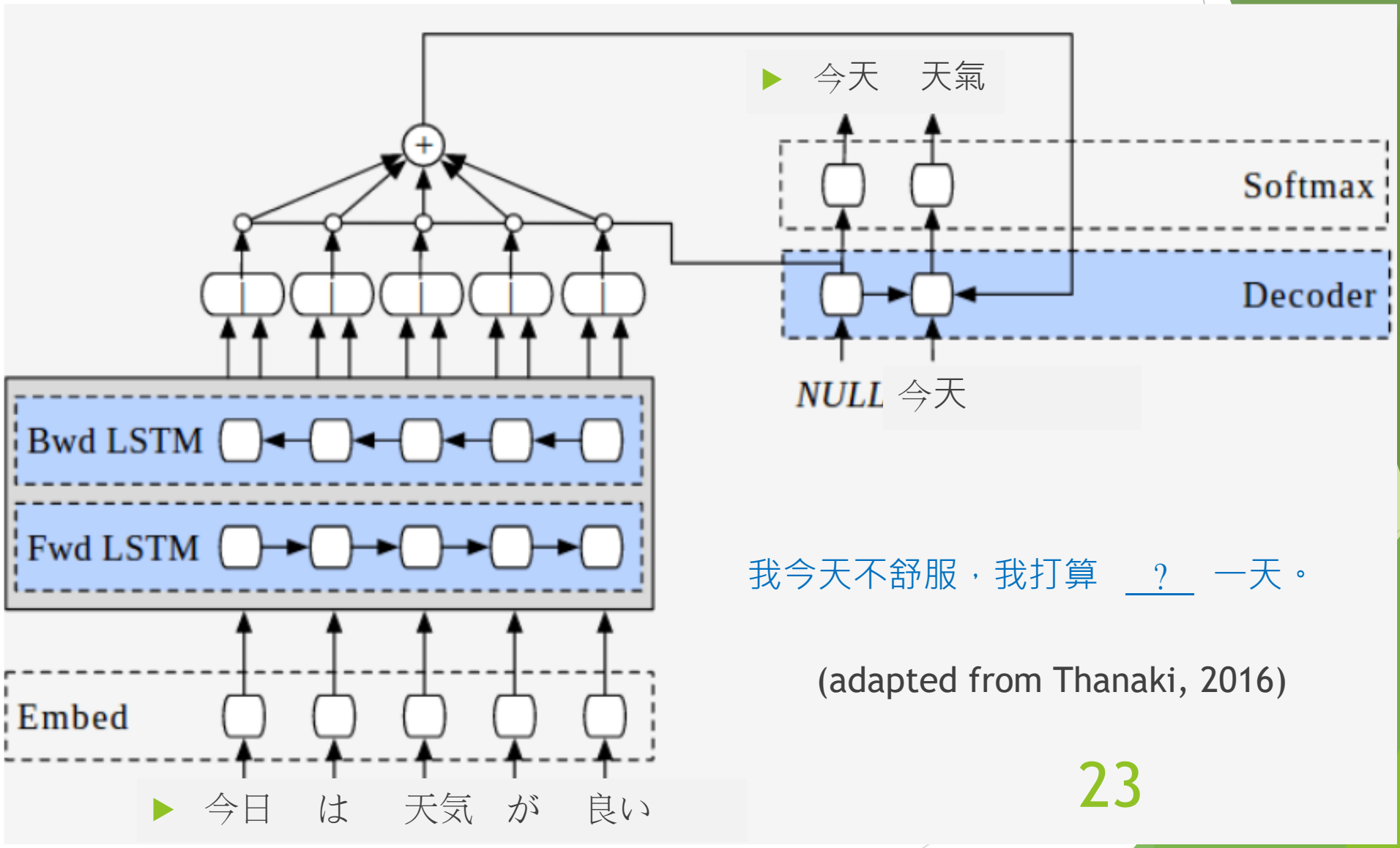
Seq2seq進行翻譯例子





(adapted from Thanaki, 2016)

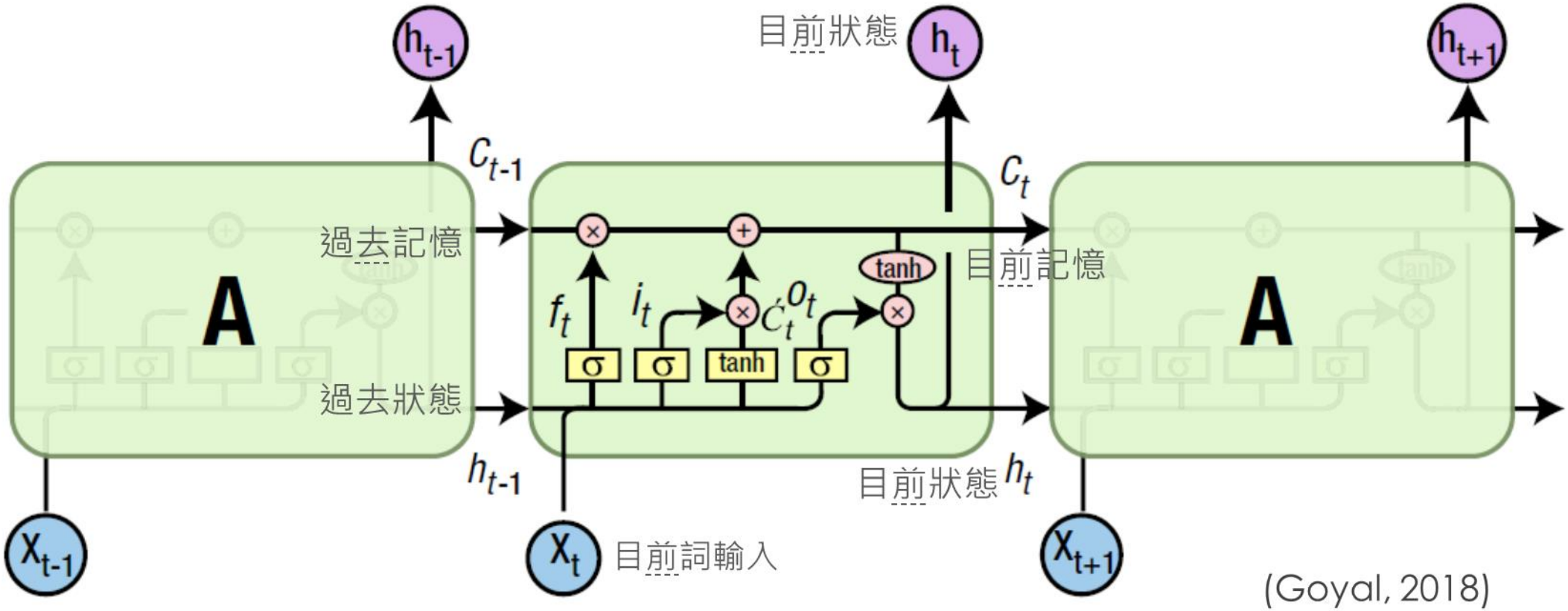
Seq2seq進行翻譯例子:加入雙向及注意力機制



我今天不舒服，我打算 ? 一天。

(adapted from Thanaki, 2016)

LSTM長短期記憶單元結構圖



(Goyal, 2018)

Figure 3-9. *LSTM module with four interacting layers*

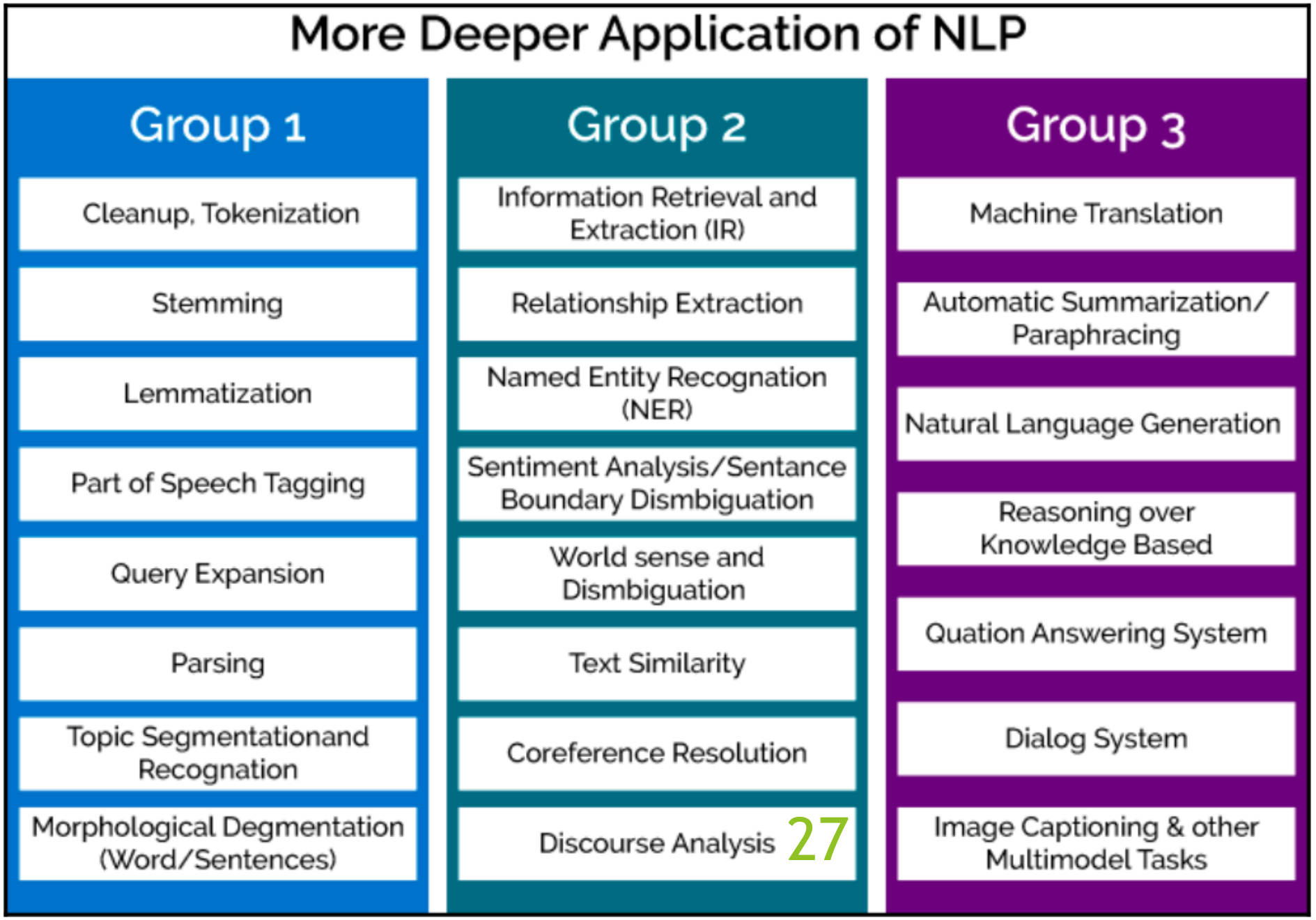
深度學習進入門檻

- 軟體
  - 程式語言: Python, Java, Matlab
    - 套件: numpy, scikit learn, pandas, matplotlib, …
  - 深度學習框架 (Gulli, 2017)
    - 上層: PyTorch, Keras
    - 下層: TensorFlow, Theano, CNTK, Caffe, Torch
- 硬體
  - GPU 圖形處理單元
  - TPU 張量處理單元
- Google Colab深度學習平台: 提供GPU/TPU每天最多12小時用量

現況與未來

- 文字探勘研討會及競賽
  - 亞洲NTCIR http://research.nii.ac.jp/ntcir/ntcir-14/
  - 美洲TREC https://trec.nist.gov/tracks.html
  - 歐洲CLEF http://clef2018.clef-initiative.eu/index.php?page=Pages/labs_info.html
- 語言資料集
  - 政府官方多語文件網站、 Wikipedia 、電影字幕網站、日漢/漢日辭典、碩博士論文網、 …
  - 公開NLP資料集 https://machinelearningmastery.com/datasets-natural-language-processing/
  - NTCIR 跨語言資料集 http://research.nii.ac.jp/ntcir/permission/ntcir-6/perm-en-CLQA.html
- 日中雙語應用
  - 學術面勘探應用: 日語相關近年研究主題的趨勢分析、文本偏見分析、小說分析、…
  - 語言學習輔助應用: 相近詞查詢、翻譯用詞建議、錯別字訂正、…
  - 其他創新應用: 跨語言檢索、雙語聊天機器人、以自然語言檢索日文系相關常見問答集、…

自然語言技術與應用

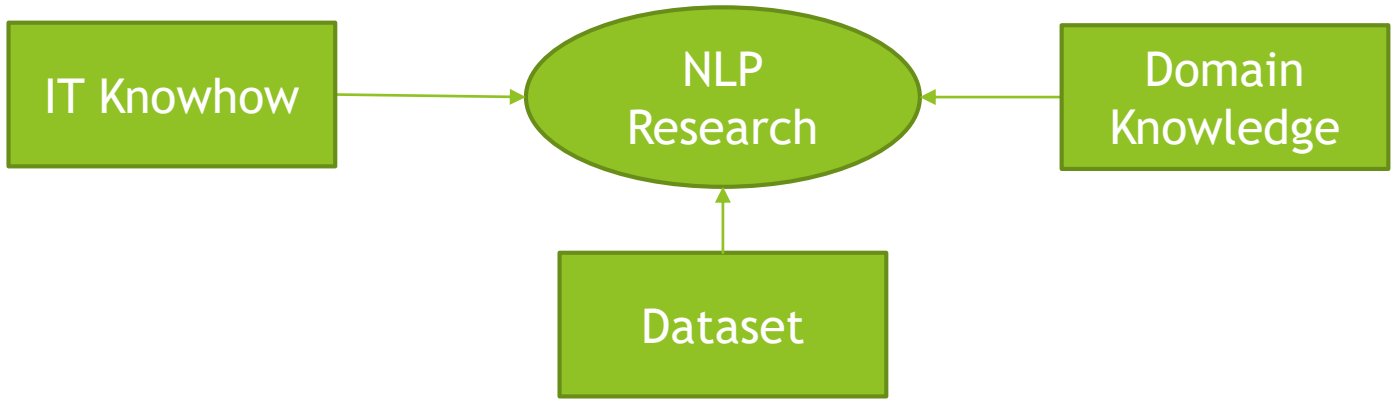


(Thanaki, 2016)

中日雙語系自然語言研究的機會

- 單語系自然語言處理已吸引足夠多關注
- 雙語系自然語言處理多和英文相關
- 中日雙語的自然語言處理須求大但少研究關注
- 自然語言處理研究的挑戰

1. 資料集(dataset)
2. 資訊處理能力(IT knowhow)
3. 領域知識(domain knowledge)



参考文献

- Goyal et al. (2018) Deep Learning for Natural Language Procesing- Creating Neural Networks with Python, Apress.
- Gulli et al. (2017) Deep Learning with Keras- Implement neural networks with Keras on Theano and TensorFlow, Packt.
- Jurafsky et al. (2018) Speech and Language Processing- An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, 3rd Ed., (Draft).
- Manning et al. (2008) Introduction to Information Retrieval, Cambridge.
- Matlab (2018) Analyze Japanese Text Data, https://jp.mathworks.com/help/textanalytics/ug/analyze-japanese-text.html
- Thanaki (2017) Python Natural Language Processing- Explore NLP with machine learning and deep learning techniques, Packt.
- Witten et al. (1999) Managing Gigabytes- Compressing and Indexing Documents and Iamges, 2nd Ed., Morgan Kaufmann.