Natural Language Processing Summer 2021 #4

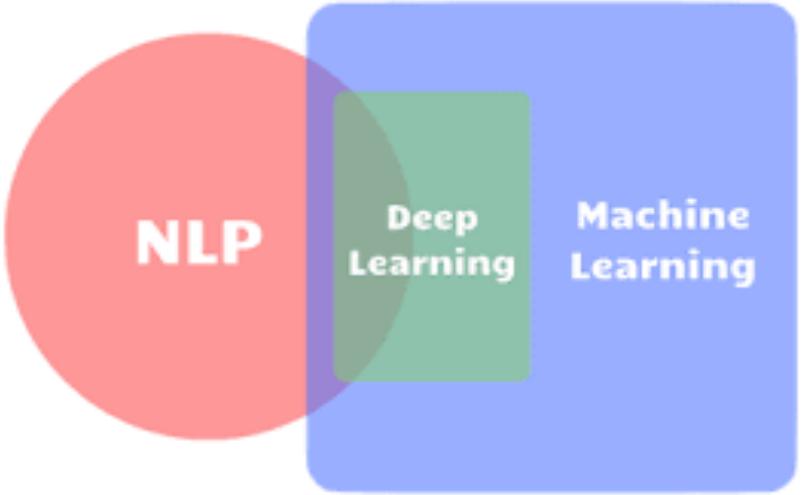
Chi-Jen Wu

Topics

- An introduction to NLP
- Language modeling
- Representation learning
- Web crawling and indexes
- Word Embeddings
- Text classification
- Sequence modeling
- Machine learning models
- Deep learning models

Two Approaches to NLP

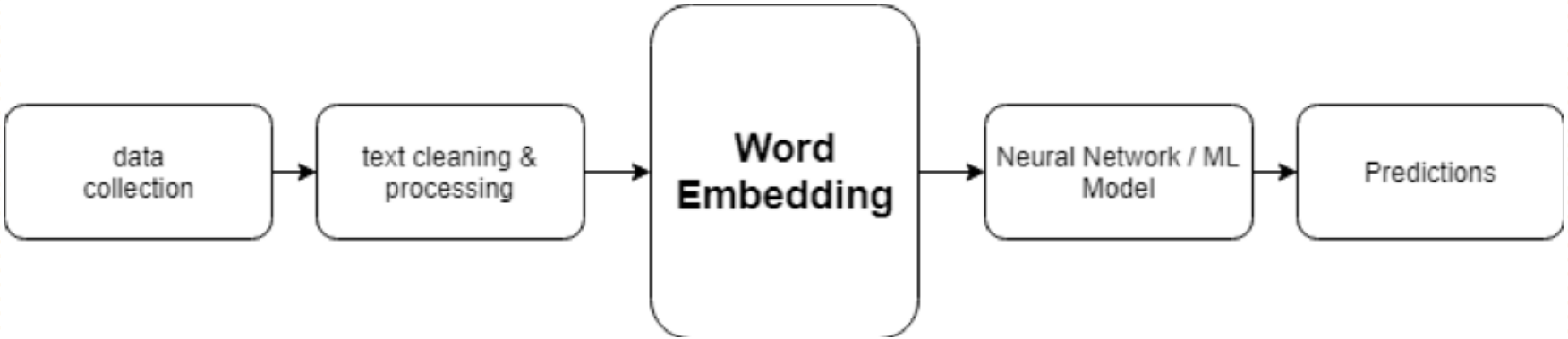
- Machine Learning
  - 處理語料
  - 特徵工程
  - 分類器
- Deep Learning
  - 處理語料
  - 設計模型
  - 模型訓練



Word Embeddings

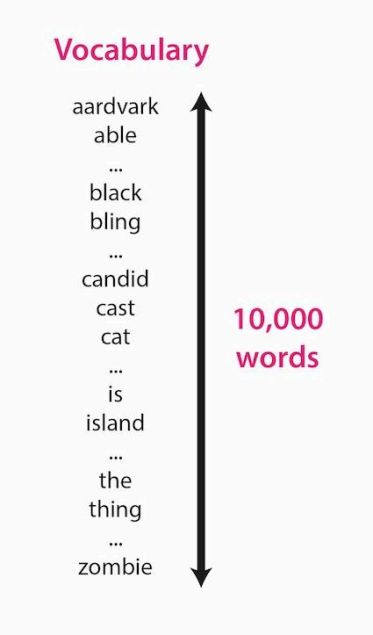
- 將字詞看作文本的最小單元
- 一個維數為所有詞的數量的高維空間映射到一個較低維數的連續向量空間中
- Embedding (映射 嵌入)
  - the representation of words
- 目前主流的方式 都以Neural Network (NN)為主
  - 之前介紹得 n-gram 就慢慢比較少人用
  - TF-IDF
- 大概分為兩大類 embedding 方法
  - Frequency based embedding
  - **Predicted Based Embedding**

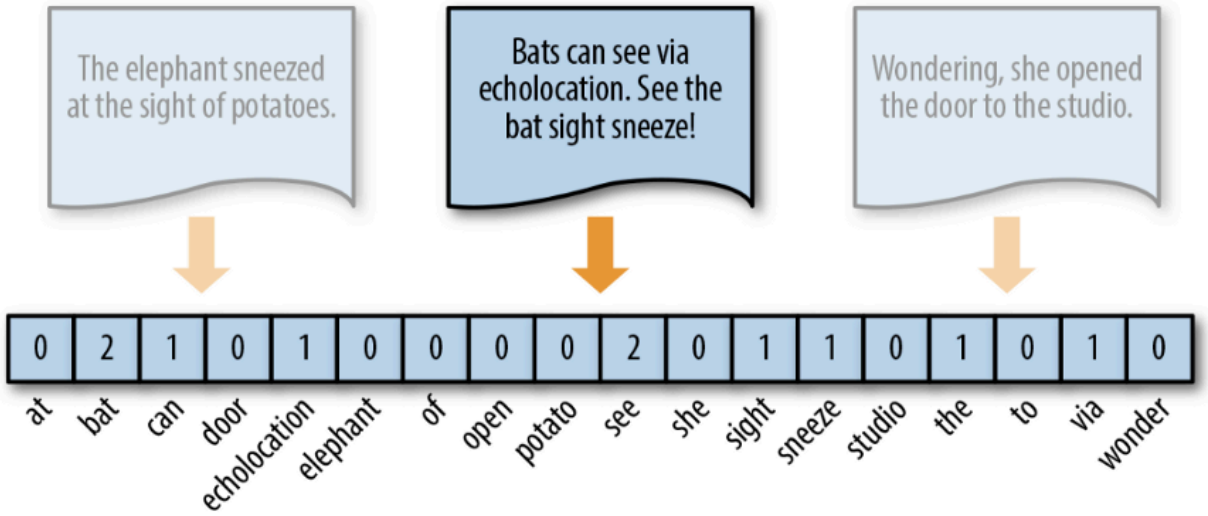
NLP flow


data collection
text cleaning & processing
Word Embedding
Neural Network / ML Model
Predictions

One Hot Encoding

- Traditional Context-Free Representations





Frequency based Embeddings

- TF-IDF Vector
  - 字詞的重要性與它在檔案中出現的次數成正比，同時與它在語料庫中出現的頻率成反比

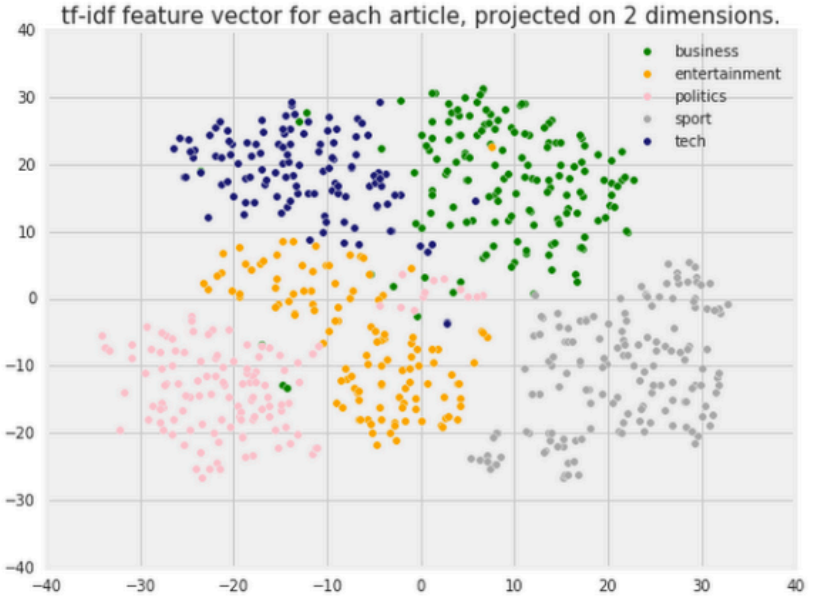
文件	the	red	dog	cat	eats	food
1. the red dog →	0.2	0.2	0.2	0	0	0
2. cat eats dog →	0	0	0.2	0.2	0.2	0
3. dog eats food →	0	0	0.2	0	0.2	0.2
4. red cat eats →	0	0.2	0	0.2	0.2	0

字詞

Sparse matrix

文件分類 by TF-IDF

- 利用文件的高tf-idf的字詞分群
  - Topic Modeling
  - 每個文件的主題
  - 利用主題關鍵字詞分群
  - Unsupervised learning
    - K-means algo
  - Supervised learning
    - SVM, KNN



Unsupervised learning

- K-means algorithm (算是最簡單的NLP算法之一)
  - 物以類聚
  - k是分成幾群, means就是每一群群心
  - 群心初始位置 (亂給, 所以群心初始位置會影響結果)
  - 時間複雜度 O(NKT) ， N 是數據數量， K 是群集數量， T 是重複次數

Supervised learning

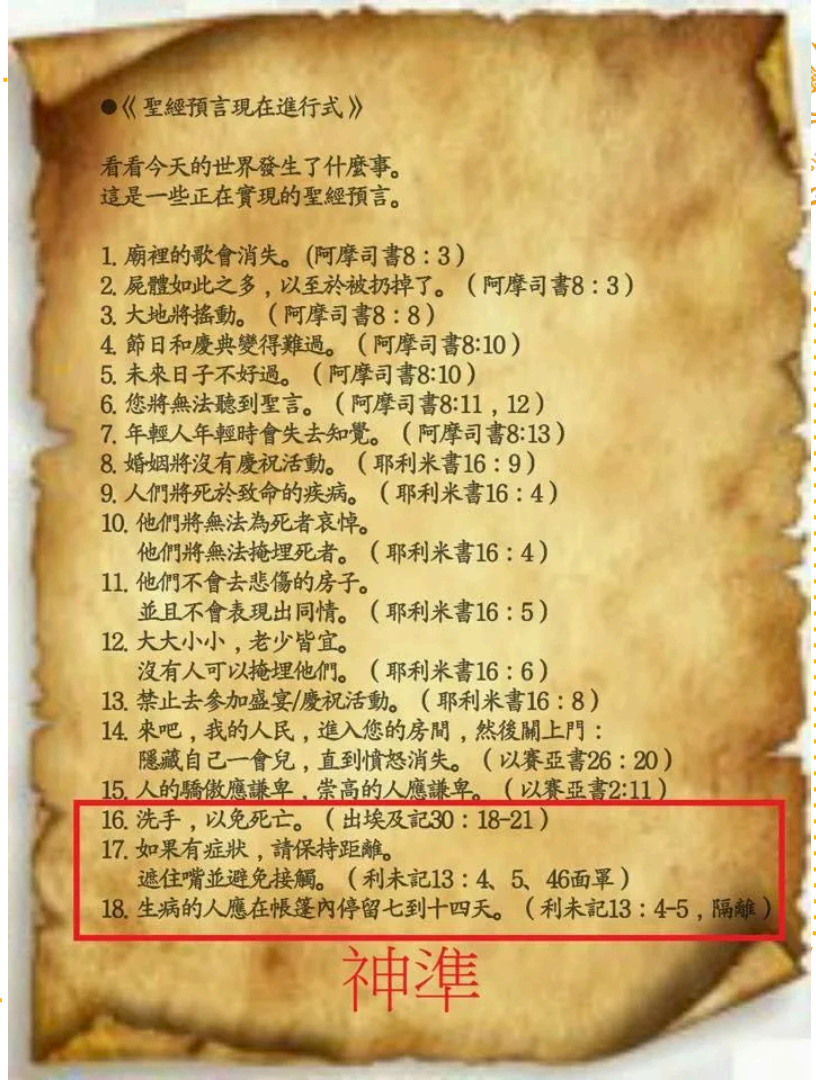
- kNN algorithm (也是最簡單的NLP算法之一)
  - k-nearest neighbors algorithm
  - 也是物以類聚,
  - **一群已經分好類別的資料** (像是yahoo電影分類)
  - **再加入未分類的資料**
  - 計算資料的"距離", 可能是座標距離或是字詞重疊程度等等 (要自行決定)

Frequency based embedding 缺點

- 沒有考慮文章上下文的關係
  - Co-Occurence Vector (2013)
    - 共現向量和共現矩陣
    - 提供另一種想法 (idea from image processing)
- 慢慢淡出NLP
  - Deep learning
  - Predicted Based Embedding太好了

什麼是上下文

- 原出處是出埃及記30:18-21



上下文和斷章取義

- 原出處是出埃及記 30:18-21
- 「你要用銅做洗濯盆和盆座，以便洗濯。要將盆放在會幕和壇的中間，在盆裡盛水。亞倫和他的兒子要在這盆裡洗手洗腳。他們進會幕，或是就近壇前供職給耶和華獻火祭的時候，**必用水洗濯，免得死亡**。他們洗手洗腳就免得死亡。這要作亞倫和他後裔世世代代永遠的定例。」

共現矩陣想法和計算

- He is not lazy. He is intelligent. He is smart.
  - Context Window = 2

He	is	not	lazy	He	is	intelligient	He	is	smart
He	is	not	lazy	He	is	intelligient	He	is	smart
He	is	not	lazy	He	is	intelligient	He	is	smart
He	is	not	lazy	He	is	intelligient	He	is	smart

	He	is	not	lazy	intelligent	smart
He	0	4	2	1	2	1
is	4	0	1	2	2	1
not	2	1	0	1	0	0
lazy	1	2	1	0	0	0
intelligent	2	2	0	0	0	0
smart	1	1	0	0	0	0

共現矩陣

語文學角度

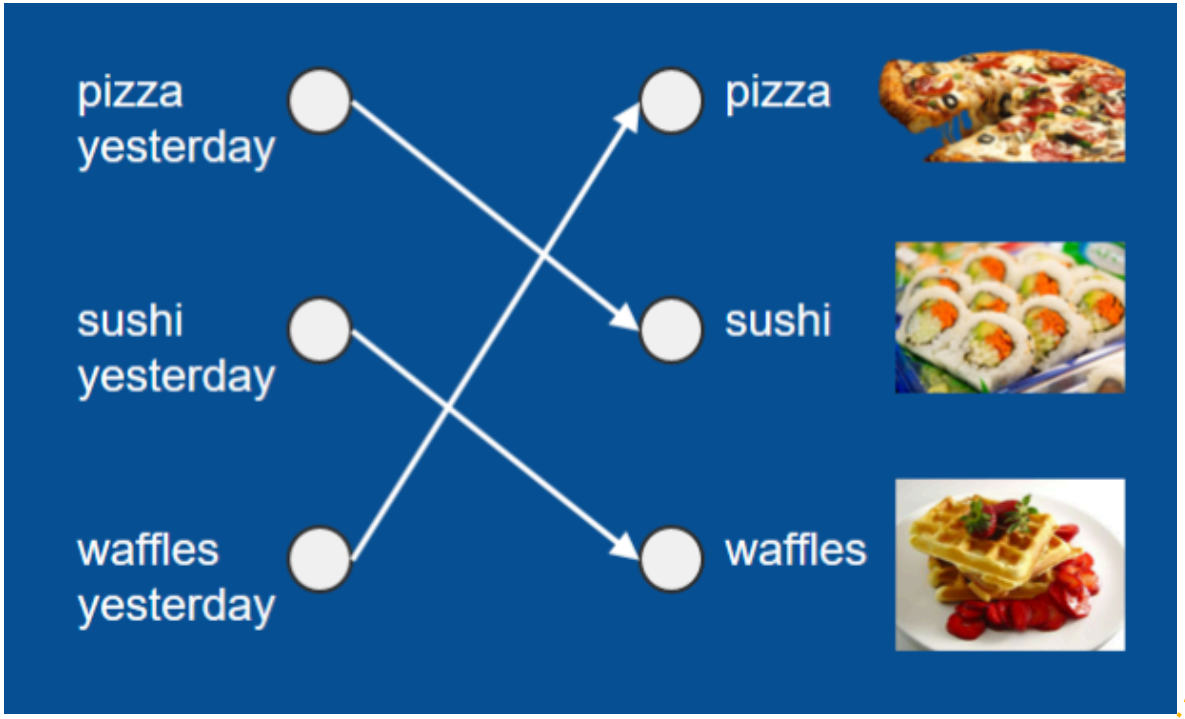
- **要理解一個詞彙的語意，首先要先理解它的上下文資訊**
  - 人類 = 男人 + 女人 (大概)
  - 如果要表示**人類**
    - 需要的**維度**很大
    - 好幾個向量
- 需要一個更好的表示方法 (embedding)
- **Sequence modeling**

Sequence modeling

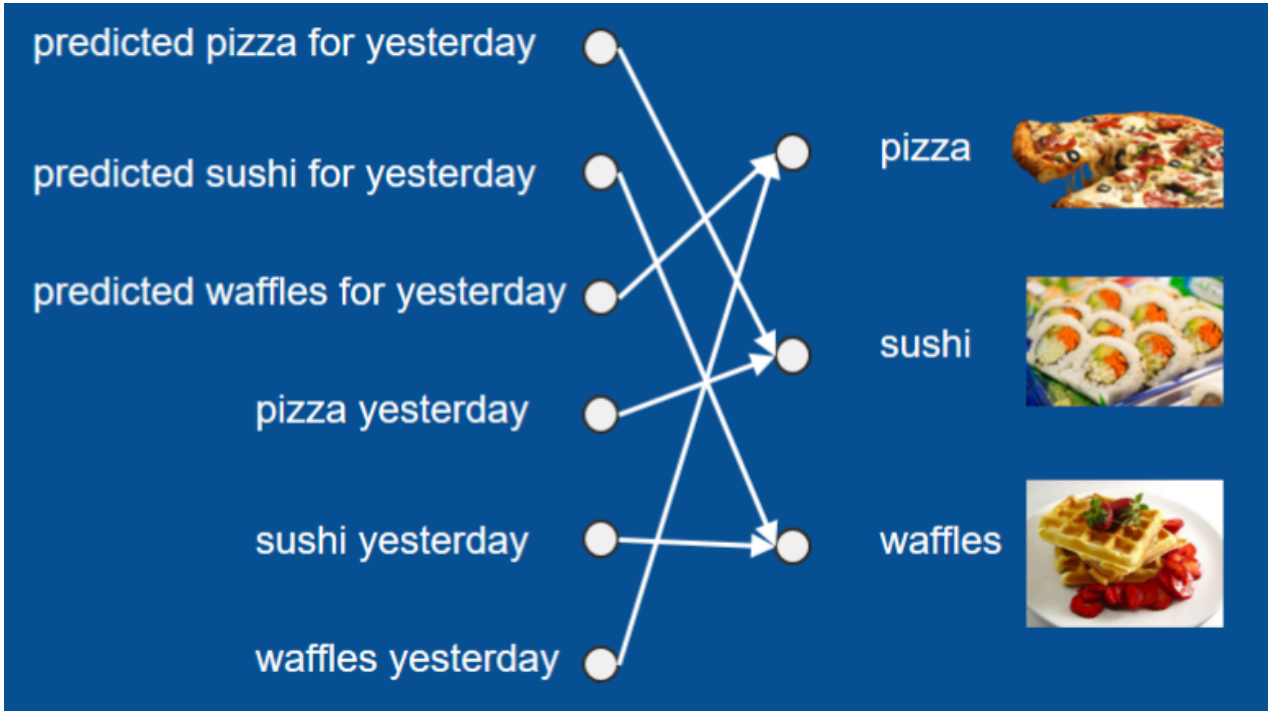
- 所以文字資料是連續的
  - 字詞的表達 (representation) 也要能連續的
  - Recurrent Neural Networks (RNN)
  - Long Short-Term Memory (LSTM)
- 很多資料都是連續的或是前後有相關的
  - 股票 (**可能有關?**)
  - 吃什麼晚餐 等等
    - 跟你前幾天吃的晚餐是有關的 (基本上)

LSTM 大概說

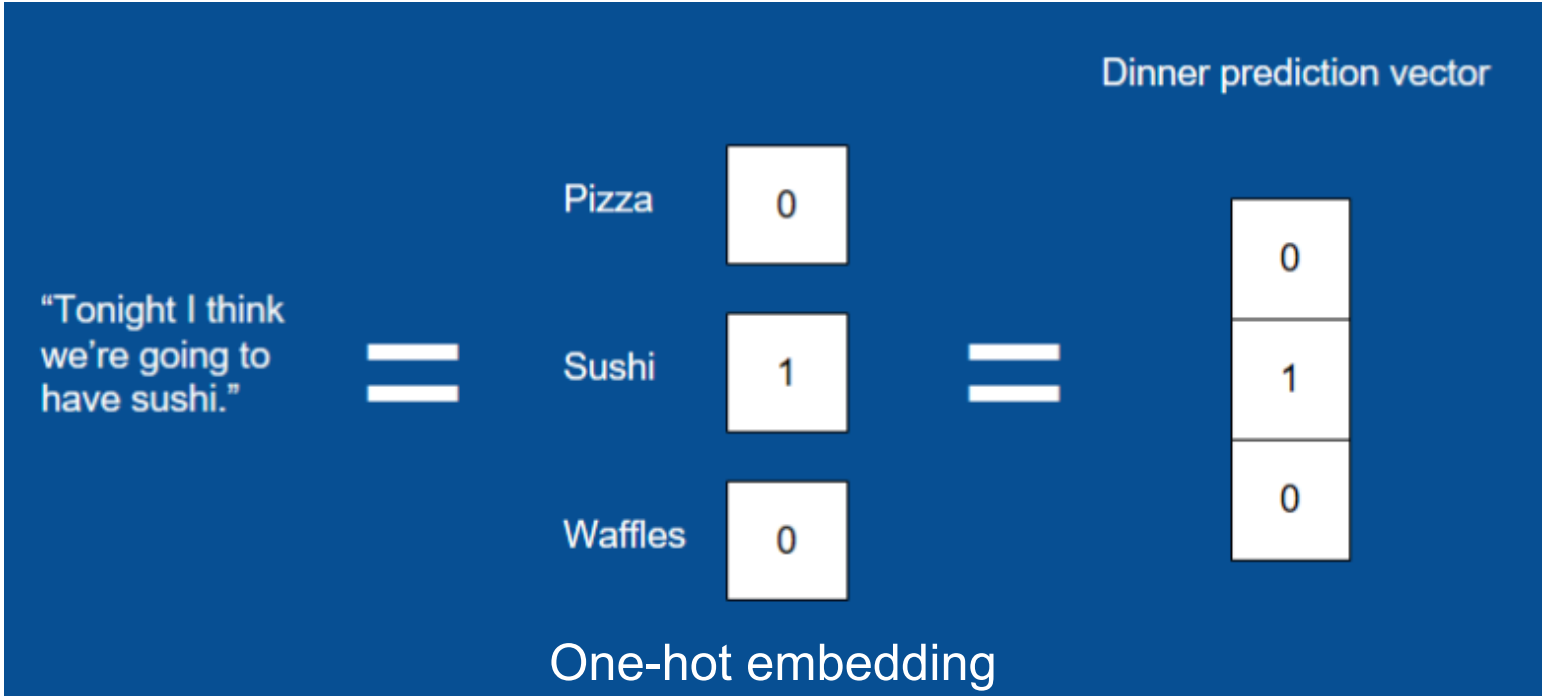
- 今晚我想來點



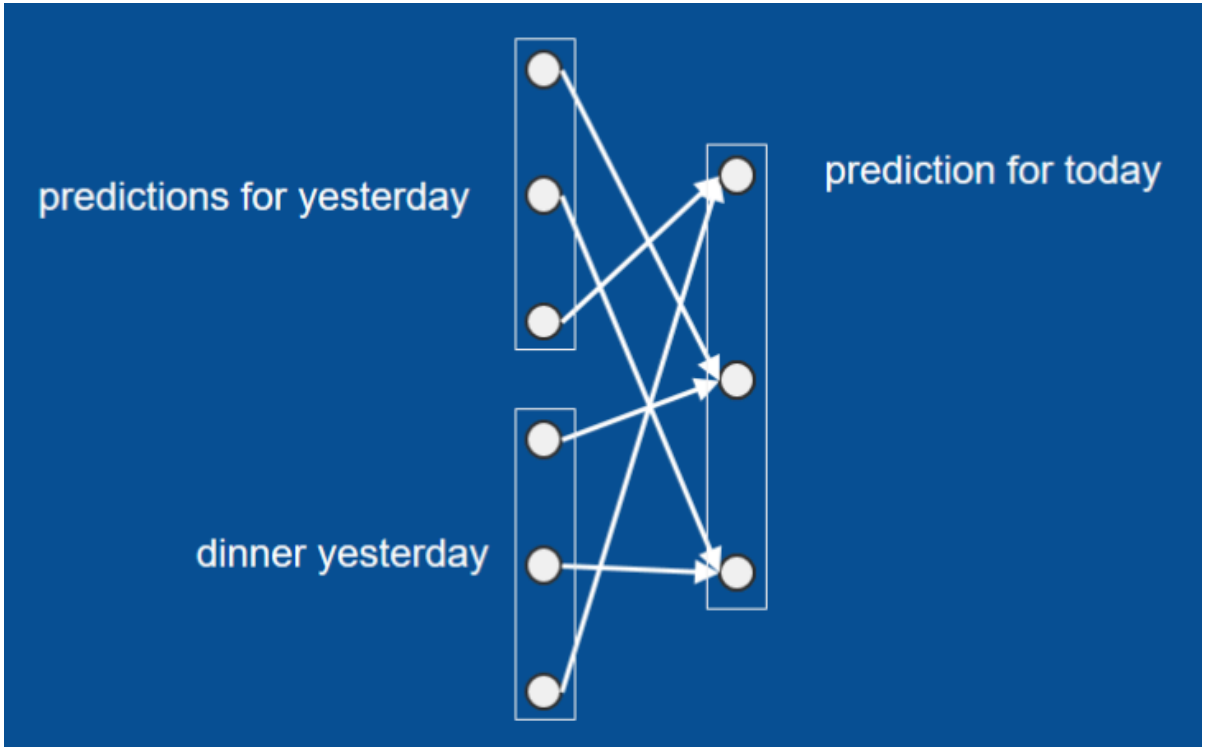
預測晚餐 LSTM模型


predicted pizza for yesterday
predicted sushi for yesterday
predicted waffles for yesterday
pizza yesterday
sushi yesterday
waffles yesterday
pizza
sushi
waffles

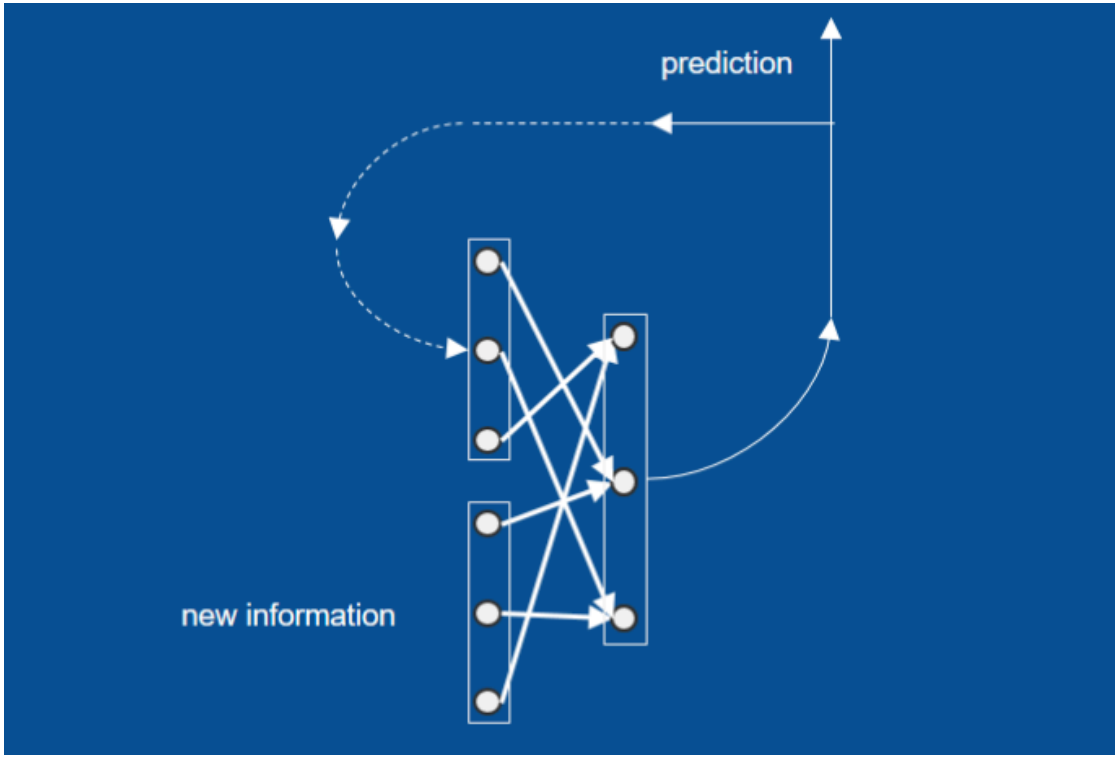
今晚我想來點壽司


Dinner prediction vector
"Tonight I think we're going to have sushi."
Pizza
0
Sushi
1
Waffles
0
0
1
0
One-hot embedding

訓練一下LSTM

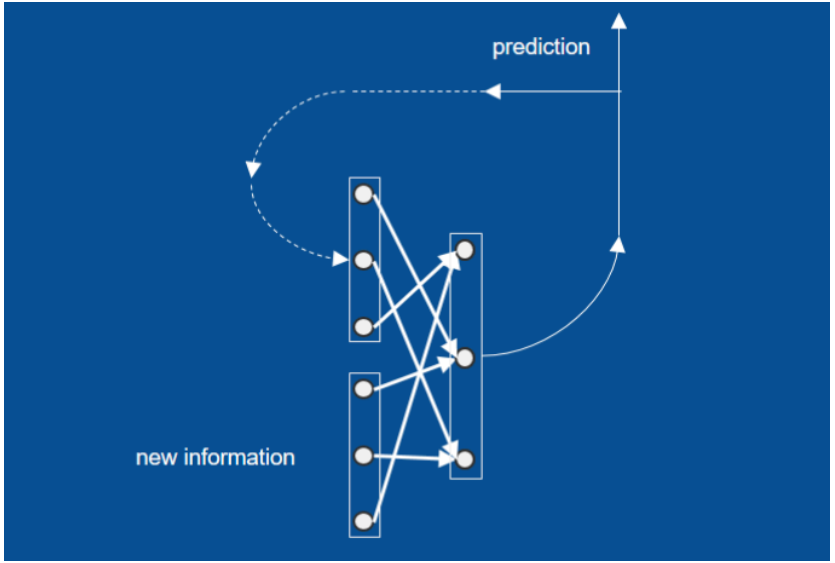

predictions for yesterday
dinner yesterday
prediction for today

LSTM testing



文字預測模型

- 把晚餐換成
  - 小明/大雄/小叮噹
  - 看/爬/吃
  - 電影/阿里山/麵包
- 給模型很多句子
  - 小明看電影
  - 大雄爬阿里山
  - 小叮噹吃麵包



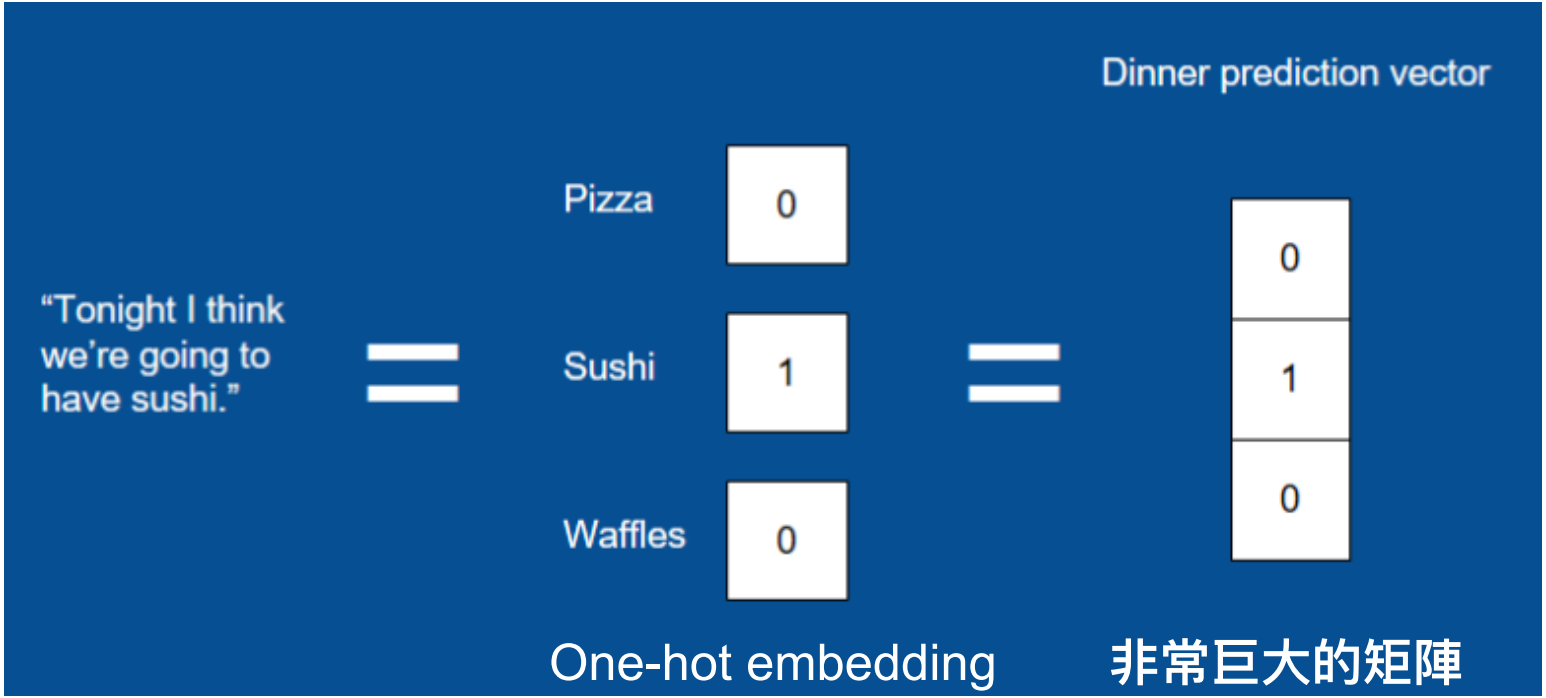
Sequence modeling 突飛猛進

- 大概都是近十年發明的演算法
- 內容非常多
- 可應用範圍也非常廣泛
  - 也因為這樣大家都AI
- 在AI課程會詳細介紹
- 有興趣的同學可以自行找國外大學的教學影片
- 也可以練習英聽能力

Predicted Based Embedding

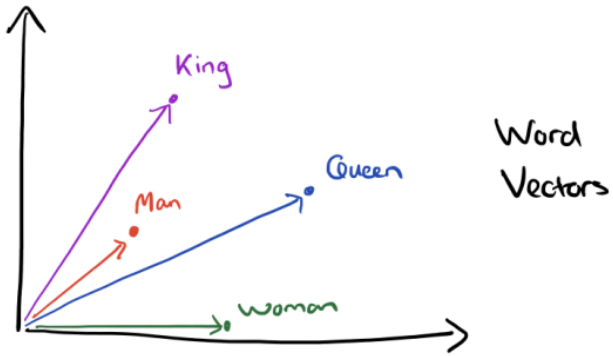
- word2vec (2013)
  - Google 的開源工具
- GloVe (2014)
  - Stanford, Global Vectors for Word Representation
- fastText (2016)
  - Facebook的開源工具
- Tencent AILab (2018)
  - 16G, 開源詞庫
- **大概兩三年就有新的技術出現**
- **大部分為 pre-trained word embeddings**

One-hot embedding 資料很多時



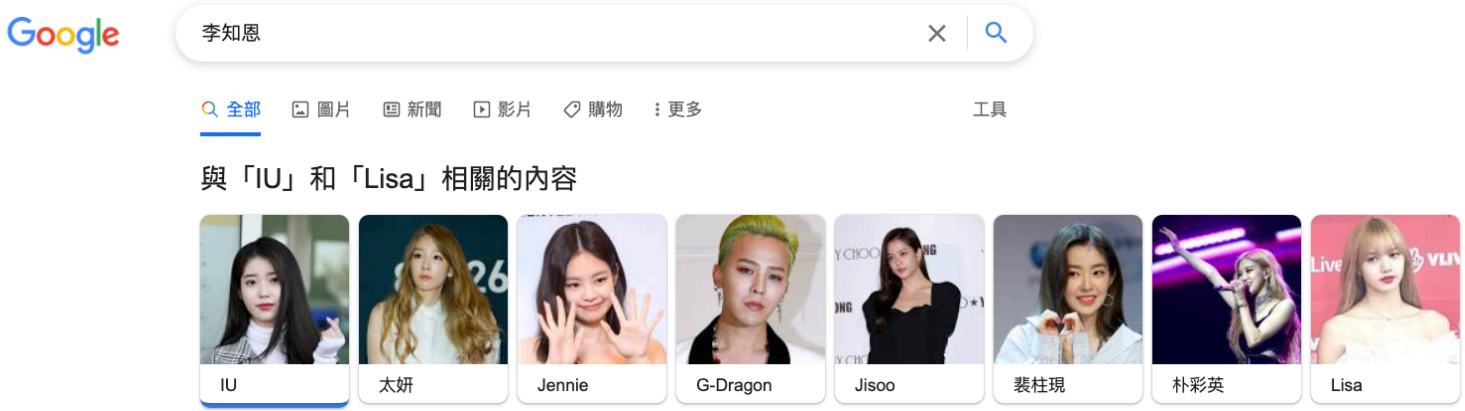
word2vec

- 根據輸入的「詞的集合」計算出詞與詞之間的距離
- 「字詞」轉換成「向量」形式
- 餘弦值 (cosine)
  - 計算距離範圍為 0–1 之間
  - 值越大兩個詞關聯度越高
- Poorly understood (目前)
  - 還在研究討論中



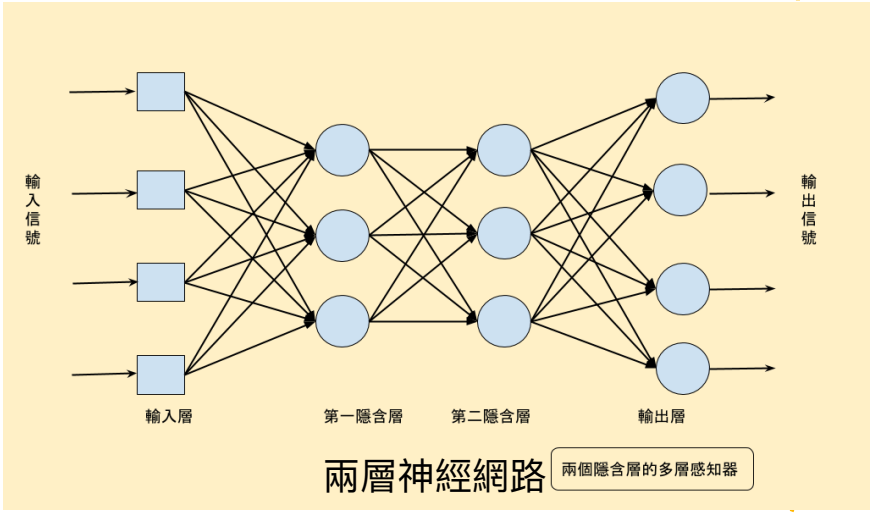
An example

- 輸入一個詞彙 “**李知恩**”，則模型訓練產生的結果
- 可能會預測在 “李知恩” 附近有較高機率出現的字是 “太妍”, “朴彩英”



word2vec 訓練

- 透過學習大量文本資料來訓練一個兩層神經網路
- 在給定一個字詞後
  - 其他字詞出現的機率
- 計算兩字詞的相關度
  - 餘弦值 (cosine)



word2vec 模型

- **CBOW**
  - 給定上下文，來預測輸入的字詞
    - 給定韓星/國民妹妹/台北/金曲獎
    - 李知恩
- **Skip-gram**
  - 給定輸入字詞後，來預測上下文
    - 給定李知恩
    - 韓星國民妹妹/李知恩/參加/台北/金曲獎/頒獎/典禮
- 缺點
  - 多義詞
  - 靜態向量

word2vec / fastText

- 訓練模型來最大化一個**目標函式**
  - 給定一個資料下去最大化所要的字詞機率
- fastText
  - 把n-gram考慮進來
  - 動態些 (新詞問題)
- 這些都是很概念的說
  - 需要計算許多機率模型
  - 我們不深入探討

HW#4 word2vec實作 相似關鍵詞延伸 (fastText也行)

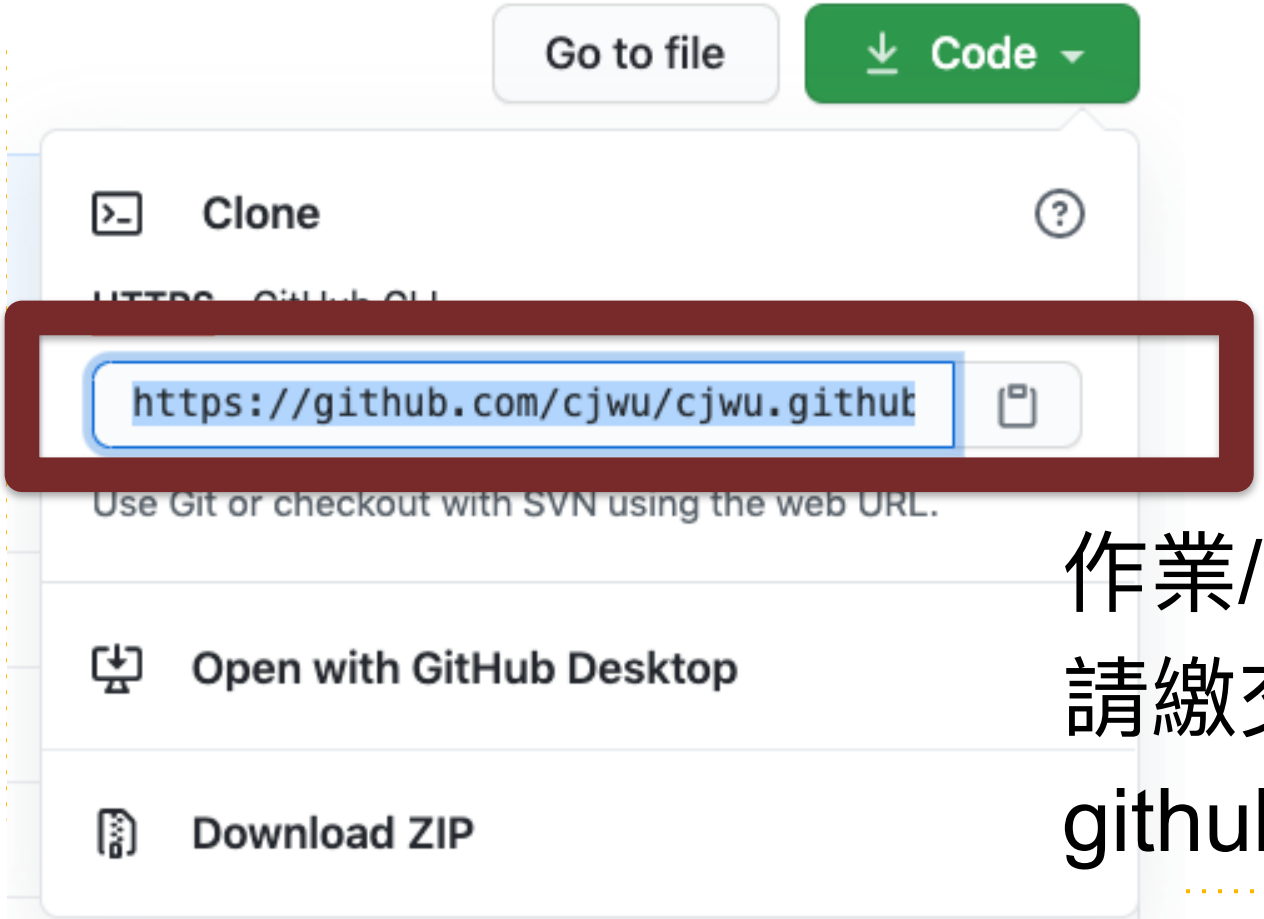
1. 訓練給定中文語料庫 (2G)

   https://drive.google.com/file/d/1EdHUZIDpgcBoSqbjlfNKJ3b1t0XIUjbt/view?usp=sharing

2. 分詞 訓練 word2vec (fastText), opencc 繁簡體轉換
   2.1 pip install word2vec
3. 得到相似關鍵詞延伸模型
   3.1 輸入 ex “李知恩”
   3.2 輸出**前二十個**相關詞
   3.3 自行調整參數去看看輸出結果
   3.4 分數以輸出詞的相關度評分

HW#4 word2vec實作

4. 請 Commit 到 github
5. 請記得要commit >= 5次
6. 不要改github branch name
7. 8/19 12:00 前 push github
8. 遲交或補交 扣20分
9. 一個function 不能超過50行 (超過一行扣一分)



作業/隨堂練習
請繳交你自己
github上的這個Link

論文名稱:	以word2vec擴展關鍵字詞應用於商品名稱自動化分類
指導教授:	
指導教授(外文):	
學位類別:	碩士
校院名稱:	國立
系所名稱:	資訊管理學系
學門:	電算機學門
學類:	電算機一般學類
論文種類:	學術論文
論文出版年:	2018
畢業學年度:	106
語文別:	中文
論文頁數:	50
中文關鍵詞:	文件分類、深度學習、詞向量、自動化分類

發展非常快速
2018還可以當碩士論文
2021已經是作業題目了

Thanks!

Open for any questions

CJ Wu

cjwu@mail.cgu.edu.tw