D12 詞袋模型(Bag-of-words)

簡報閱讀 範例與作業 問題討論

重要知識點

- 了解 Bag-of-words 原理以及如何運用 Python 實現。

如何訓練文字模型

當我們想訓練一個 NLP 模型時，大家都會遇到下方問題：

- **我們如何將文字輸入 ML 模型？**
- **我們如何在電腦中表達不同文字的意義？**

Word Embedding

那在 NLP 領域，我們如何讓電腦了解文字？

- 最簡單的 word embedding 方式就是將**每一個單字以一個數值表示**(Label encoding)，再轉換為 One hot encoding 的方式呈現(因為不同文字之間，通常沒有強弱之分)。

Label Encoding

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

→ One Hot Encoding

Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

[Label Encoding v.s One Hot Encoding]

初學Python手記#3-資料前處理(Label encoding、 One hot encoding)

Pat Huang Follow
Apr 25, 2019 · 6 min read

這兩個編碼方式的目的是為了將類別 (categorical)或是文字(text)的資料轉換成數字，而讓程式能夠更好的去理解及運算。

Label encoding : 把每個類別 mapping 到某個整數，不會增加新欄位

One hot encoding : 為每個類別新增一個欄位，用 0/1 表示是否

Label Encoding

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

→ One Hot Encoding

Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

補充資料：Label Encoding & One Hot Encoding

如何表達一句話，或一篇文章？

現在我們已經將文字轉換為 One hot encoding 的格式，那表達一句話或一篇文章最簡單的方式，就是將這句話或這篇**文章中出現過的文字全部加起來**，這也就是我們的 Bag-of-words，可以想像成我們把所有的單詞放進一個袋子(詞袋)。

補充資料：Label Encoding & One Hot Encoding

步驟

假設今天我們的資料集包含正反面評價 1000 則

- 首先我們用資料集所有的單詞建造一個字典，也就是所有的單詞需要有對應的 index (沒有順序限制，但固定後就不可改變)，假設字典大小為 3000 (也就是 3000 個單詞)。
- 各別的評價可以看為一個袋子，所以我們要用一個向量表示這個評價。
- 我們先建造一個 3000 維，數值皆為 0 的向量 (ex.[0, 0, 0,.....])，再將這個評價內有出現的單詞取出，找到對應的 index，將向量中這個位置的值 +1。
- 舉個例，我們在一個評價中找到兩個 good，good 對應到的 index 為 5，所以我們就在向量 [5] 的位置 +2，變為 [0, 0, 0, 0 , 0, 2,.....]。

Bag-of-words

優點：

1. 直觀，操作容易，並且不需要任何預訓練模型，可套用在任何需要將文字轉向量的任務上。

缺點：

1. 無法表達前後語意關係。
2. 無法呈現單字含義：許多單字有多種不同含義，如我要買蘋果手機跟我要去菜市場買蘋果，兩句話中的蘋果意義不相同，但在 Bag-of-words 中無法呈現。
3. 形成稀疏矩陣，不利於部分模型訓練：假設我們訓練的 corpus 內有 100000 個單字，那要表達每一個單字就是(1,100000) 的 vector，其中絕大部分都是0的數值。

儘管 Bag-of-words 有諸多不足的地方，然而在較為簡單的情境下，其效果仍然相當不錯。

改善：

- Word Embedding 為 NLP 中相當重要的領域，在 Bag-of-words 之後有許多改進的方式陸續被提出，如 TFIDF、Word to Vector、GloVe、ElMo 到近期的 BERT，之後章節會陸續為各為學員介紹。

參考資料

網站：Different techniques to represent words as vectors (Word Embeddings)

Different techniques to represent words as vectors (Word Embeddings)

From Count Vectorizer to Word2Vec

Karan Bhanot Follow
Jun 7, 2019 · 7 min read ★

Currently, I'm working on a Twitter Sentiment Analysis project. While reading about how I could input text to my neural network, I identified that I had to convert the text of each tweet into a vector of a specified length. This would allow the neural network to train on the tweets and correctly learn sentiment classification.

Thus, I jot down to take a thorough analysis of the various approaches I can take to convert the text into vectors — popularly referred to as Word Embeddings.