

r06922115

鄭皓謙

HW1 report

Task 3

task3與task2作法相似，差異在如何利用時間的資訊。

首先產生fake link，我猜測在testing set完全沒有link的情況下，產生一些fake link可以幫助node2vec，產生fake link的方式則因為時間資訊與task2有些不同。

接著根據title abstract產生document embedding。

最後將兩者並且加上時間當作classifier的輸入，做binary classification。

不同於doc2vec的做法，這邊產生doc embedding的方式不是pre-train，而是直接跟link prediction訓練，所以embedding可能不具有generalization的特性

task3我嘗試了node2vec作為graph embedding，再用NN處理word embedding，同時利用三個feature用nn做classifier。

Classifier

每個node會有word embedding跟graph embedding作為feature以及論文時間資訊，將三種feature接起來後，link prediction則用兩個node的feature接起來，配合BCE loss，用NN做binary classification。

Doc Embedding

與task2相同，為了利用title與abstract的資訊，我參考Google的論文，Attention is all you need，裡面transformer的架構，嘗試將title+abstract轉成一個向量，為每個node得到doc embedding。會選擇如此原因是我曾實作過這部分，減少實作時間。

原本transformer model的目標是用來訓練seq2seq model做機器翻譯，翻譯的原文序列為K, key，而翻譯後的文本序列為Q, query，Q K分別會做self-attention，然後再用Q對K做attention，象徵對每個input sequence的attention weight。每個node，title與abstract會接在一起，將Q作為citation link的destination node，K作為source node。

由於原先transformer是seq2seq model，所以輸出是sequence，我增加了一個trainable weight再對輸出的sequence做一次attention，當作是summary，如此一來便對每個link pair有了text embedding。使用glove當作pre-trained word embedding。

Graph Embedding

與task2相同，首先根據Fake link產生的graph，使用node2vec的feature當作graph embedding。

Negative sampling

根據task1同學的分享，藉由計算adjacent matrix, $A^2 = A * A$ ，特別挑選那些兩步之間有連線的node，而一步卻沒有link的node pair。0.8機率sample上述情況，剩下0.2機率則uniform sample training node set。

Fake link

由於testing set完全沒有資料，這次要做test node的fake link是計算training set的out-degree的mean與std，然後為每個test node從normal distribution sample一個out-degree d，決定好的testing node的out-degree後，要產生d個fake link。

為了不產生時間跨度較大的fake link，根據training set裡面含有的link，計算每個link之前的時間差距，計算出時間差距的mean以及std，當要產生d個fake link的時候，首先對training set node 做 uniform sampling，選定citation node之後檢查與時間差距分佈的z-score，當z-score大於2的時候，重新sample一個citation node。

嘗試與心得

與task2作法很相似，但是如果完全不利用時間資訊的話，連all zeros都過不了，令我非常驚訝，而加入時間資訊以後問題就變簡單一點了，一樣為了讓graph embedding

有些效果，這次我依然嘗試產生fake link，除了不產生過去cite未來論文的link，也使link的時間跨度不要太大。我後來覺得由於每個點的out-degree很低，所以node2vec很吃運氣，所以未來可能可以嘗試加入weight，利用額外的pre-trained model計算weight，或者是多sample幾個點，但是給較低的weight可能比較好。

實驗圖 accuracy

藍色部分是training set 以及 negative sampling的link

而紅色為validation set，不帶有negative sampling，全部都是positive sample

分別為accuracy 與 loss的變化，我解釋為training 跟validation的曲線相似，沒有相異的趨勢，所以沒有overfit，而訓練中止則是因為時限的緣故。

