Homework 3

60547047S 羅天宏

一、方法概述：

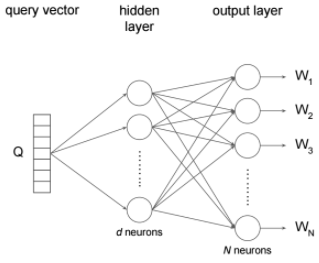
　　這次實作的方法為Relevance base word embedding。在論文中，講者重新敘述了feedforward neural network本身代表的含意，藉由定義的objective function以及network structure來學習IR中relevance model的兩個問題─distribution和classification。

　　DNN本身就是一個multi-layer perception，可以自動將輸入與輸出的資料中，學習到如何representation，優點是在學習的過程中，網路本身就會學習到這些特徵該如何抽取，抽取的方式可以想像成是weight的更新與修改，而最後一層的activation則是可以看成依據不同任務而設計。

　　本篇論文的作者說，如果output layer的activation為softmax，那麼可以視為在學習一個distribution。反之，如果最後一層的activation是sigmoid，則可以視為binary classification。此外，Back-propagation則是類似Maximum Likelihood Estimation的過程。這次實作是著重於如何學習一個distribution，因此為前者。

二、實驗：

設計一個feedforward neural network，共三層，一層為input layer，一層為hidden layer，最後一層即為output layer，activation為softmax。



Train Set: 800 long query

Test Set: 16 short query

實驗設計：

**Loss Function：**categorical crossentropy 、 kullback leibler divergence

**Bach Size：**16

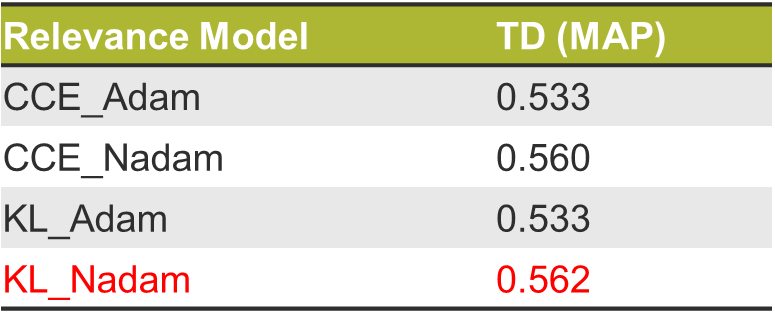
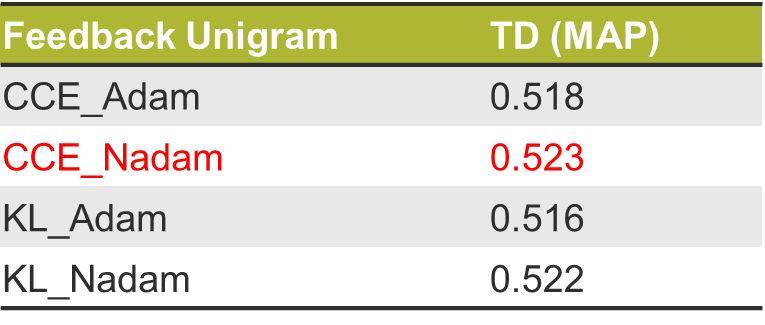
**Epoch：**55

**Objective Function：**Feedback unigram、Relevance model

**網路架構：**

|  |  |  |  |
| --- | --- | --- | --- |
| Layer | dimension | formula | activation |
| Input layer | 51253 |  | none |
| Hidden layer | 350 |  | none |
| Output layer | 51253 |  | softmax |

實驗結果：



討論：

　　因為這次的目的是學習機率分佈，因此loss function主要是設計為cross entropy，搭配上不同的Optimizer。最後發現，網路本身能夠將這樣的機率分佈學習得很好，且因為short query 和long query兩者十分相似，因此應用在short query上面的效果一樣不錯。比較的對象主要是不同的objective function，分別是feedback unigram和relevance model，以後者表現較佳，最好可達到0.562。

共有三個問題，第一個是Training set和Test set兩者太相近，第二個是能不能應用在另一個網路架構，最後一個是能不能超過自己的Teacher?

　　第一個問題我認為這代表這個網路本身沒有太多延展性，只能根據特定的任務或corpus來學習，若這兩種其中一個改變的話，整個網路又要重新設計，不夠general。

　　第二個問題可以從另一個實驗中看得出來，另一個實驗設計了LSTM和CNN與word embedding、embedding layer的組合，只是比較的方式不公平，因此沒有比較的基礎。在同樣的條件下，如同樣的input、objective，用更expressive的neural network，能不能提升效能？這是以後有機會可以試試看的部分。

　　第三個問題是網路學習後的結果，能不能超過原來unigram、relevance的表現？這代表網路的推論能力，而不是目前統計裡找到相關性的能力，我認為根本的原因或許在於loss function的設計。以上是這次實驗的簡單心得。