基于 CNN 卷积神经网络和 wordoverlap 模型的章节排序

摘要：本文为解决面向智能问答的章节排序任务，提出了一个结合 CNN 卷积神经网络和 wordoverlap 的模型，能够针对某个问题下的所有答案进行排序，使得其符合人工的标注准确度。本文的 CNN 模型首先将问题以及其下的一个答案使用 Word2vec 模型进行映射，并且放入 CNN 中提取特征，得到了关于这个答案以及其下的一个问题之间的匹配程度的分数。除此之外，我们设计了一个 wordoverlap 模型进行结合使用最后这个联合模型，构建答案和问题之间的近义词表，以及通过判断问题和答案之间相同词或者近义词的个数来判断应该使用 CNN 模型还是 wordoverlap 模型。我们在网上公布的测试集上进行测试，发现我们设计的模型的效果表现较好。

Abstract: This paper is aimed at solving the problem of the sorting of chapters. We propose an innovation motric which combines the model of CNN and wordoverlap. This metric can sort all the answers under a certain question and can reach the level of human's operation. The CNN model here use method of word2vec to map the words in a sentence to a matrix which can be processed by CNN and get the effective representation of the features. And we can get a score which can measure the level of matching from this neural network model. Addition, we also construct a statistic model which is wordoverlap. The metric can use the same words or synonym in the question-answer pair to get the score for measurement. In the actual implementation, we use the condition of the number of the same words and synonym to decide which model we should use. In other words, we combine these two models. We do experiments on the dataset of the public training set. The results demonstrates the effectiveness of our metric.

1. 引言

智能问答一直是自然语言处理和人工智能领域的一个前沿研究课题。其目标是对用户通过自然语言表述的问题直接提供精确答案。在本文中，针对 CCIR 与和搜狗搜索联合主办“面向智能问答的篇章排序”智能问答评测比赛上的任务，我们提出了一个基于联合 CNN 以及 wordoverlap 模型的章节排序方法。

鉴于 CNN 强大的特征提取能力以及 word2vec 的字符映射功能。我们构建了一个能够将问题和其对应的一个答案映射成一个分数的 CNN 模型。该模型产生的分数应该符合以下的特性：该分数应该符合人工标注的时候的趋势，即问题和答案之间的匹配程度越高，该分数应该越高。

传统的基于统计的自然语言处理模型具有很高的使用价值。在一般的问题和答案下，存在许多的相同词或者是近义词。这些特征是匹配的问题和答案之间十分重要的依据。wordoverlap 算法的本质在于使用问题和答案之间的近义词的相似度来衡量他们之间的匹配程度， 从而能够得到一个对应的分数。在这里我们使用余弦相似度首先来构建一个近义词表，以便可以更好地处理问题和答案之间的相似度并且得到较为准确的结果。

从上述的描述和分析中可以看出，wordoverlap 模型是基于统计的模型，具有较好的稳定性，但是在实际中由于分词的方法以及其他的特殊字符的出现，问题和其匹配的答案之间却可能存在很少的相同或者近义词。而 CNN 模型是基于算

法的一个高效模型，能够自动地很好提取问题和答案的语义特征，并且进行依据此进行匹配程度的分析。但是这个模型的效果收到我们训练的方式，以及训练数据的多少的影响。在很多情况下的鲁棒性不如 wordoverlap 模型。在这里我们提出了一个联合 CNN 和 wordoverlap 模型的方法。在问题和答案之间存在比较多的相同词汇或者近义词的时候，我们使用鲁棒性较好的 wordoverlap 模型；当问题和答案之间的相同词或者近义词的数目较少的时候，可以使用 CNN 模型来获得较好的效果。

下面将详细阐述 CNN 模型的设计，CNN 与 wordoverlap 联合模型的设计，以及我们的代码描述。

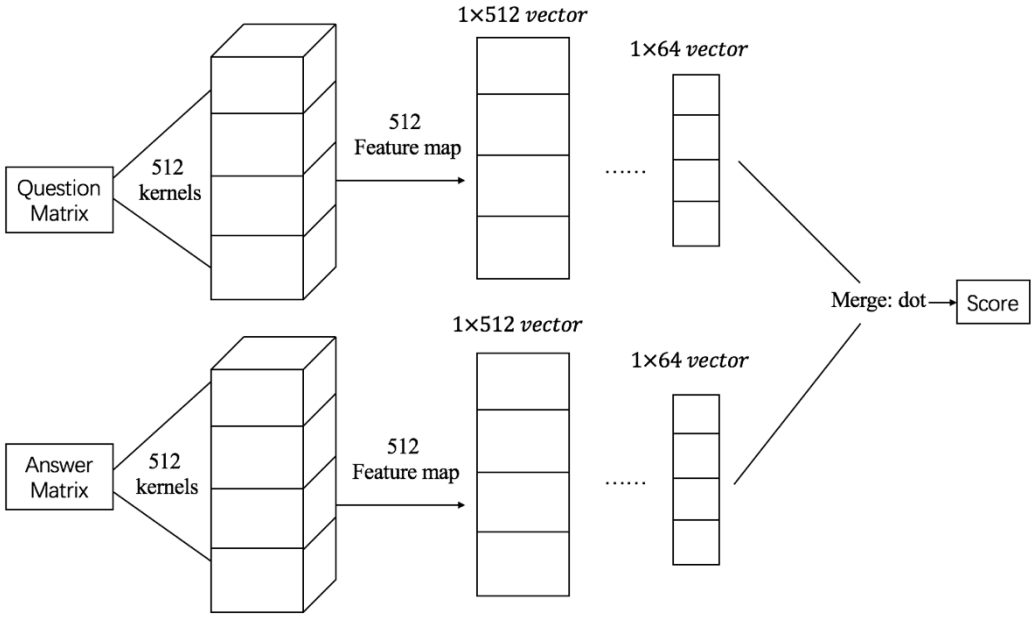
2. CNN 模型设计

1） Word2vec 的训练

首先我们需要将给出的训练数据集中的中文单词映射成为向量的形式，
也就是一个 Embedding 的过程。在这里我们使用的方法是首先使用 nlpir 的包进行分词，并且用分好的词来训练一个 word2vec 模型。这个模型的训练使用了维基百科的训练数据和本次比赛给出的训练数据以及测试数据。在训练完成之后，我们可以将分好词的单词映射成为相同维度的向量。而在自然地，问题和答案各自映射成为一个二维矩阵形式。针对本次的数据的特点，一般问题能够分出的单词较少，答案能够分出的单词数较多，所以这个答案矩阵的高度明显高于问题矩阵。我们认为这是合理的，因为问题对于其下的每一个答案来说都是一样的。那么其高度并不会很大程度上影响最后的匹配分数的计算。

2）CNN 模型的设计

在使用训练好的 word2vec 模型诸侯，我们已经可以得到一个问题矩阵和答案矩阵。在这种形势下可以放入 CNN 模型进行特征提取。问题矩阵和答案矩阵分别通过两路 CNN，分别输出一个向量，即我们的提取的关于问题和答案的特征。将两者的向量进行 merge，在这里是通过点积两个向量的方式。下图中说明了我们的 CNN 网络的基本设计框架：



其中在第一个卷积层我们输出了 512 个特征图，并且在此之后将其再次卷积，而且拼接称为一个 512 维的向量。该向量经过多层全连接层的神经网络之后，就得到了我们需要的 64 维的特征向量。两者的点积就成为最后的，衡量问题和答案之间相似度的分数。该分数应该符合人工标注的时候的趋势，即问题和答案之间的匹配程度越高，该分数应该越高。

3. CNN 和 wordoverlap 模型联合

Wordoverlap 模型是基于统计的自然语言处理模型。尤其是问题和答案之间存在较多的相同的词的时候，具有较好的表现。wordoverlap 算法的本质在于使用问题和答案之间的近义词的相似度来衡量他们之间的匹配程度， 从而能够得到一个对应的分数。CNN 模型和 wordoverlap 模型各自都有很好的效果。在这里我们考虑将他们两者联立。以期能够获得更好的效果。

我们处理的方法是这样的：首先我们需要构建近义词表，若对于测试数据的 query 及 answer，有在近义词表中的单词，则作为相同词对待。遍历所有测试数据中的问题及答案，若其中的相同词及近义词的对数超过一个阈值，则使用 wordoverlap 的方式产生对于该问题的问题答案对分数；若其中的相同词及近义词的对数少于一个阈值，则使用 CNN 模型的方法来产生对于该问题的问题答案对分数。在实际的最后的得分中，我们还有另外的一个机制：因为在分词的时候，有些单词的数目特别少，或者问题和答案之间实在太相近，我们模型得出的分数对于不同的问题相差很少的时候，按照一般的人工的思想，我们最后应该对其进行随机排序。以上的思想和方法经过我们对于实际的，已经公开的训练数据集上的测试之后，发现效果是十分好的。

4. 代码描述

对于训练及测试数据的解析编码部分，该部分是通过如下两个步骤完成：

1）通过 get_words_for_test_json.py 这份程序完成对于输入的训练及测试数据的读入、解析、分词功能。

2）通过 train_word2vec_model.py 这份程序完成对于分好的单词进行进行训练，训练出对应的 wor2vec 模型。

3）对于分好词的输入数据，通过 get_CNN_wordoverlap_score.py 进行处理，处理方法如下：

1>建近义词表，若对于测试数据的 query 及 answer，有在近义词表中的单词，则作为相同词对待。遍历所有测试数据中的问题及答案，若其中的相同词及近义词的对数超过一个阈值，则使用 wordoverlap 的方式产生对于该问题的问题答案对分数；若其中的相同词及近义词的对数少于一个阈值，则使用 CNN 模型的方法来产生对于该问题的问题答案对分数。

2> wordoverlap.py 完成对于相应问题采用 wordoverlap 的方式得到相应问题答案对应分数的功能。

3> 完成 CNN 的部分:CNN 的代码执行分为训练与测试两个部分：

- 训练：执行 Final_CNN_Model_train.py 进行 CNN 模型的训练，结果保存为 my_model_weights.h5 和 my_model_architecture.json
- 测试：执行 Final_CNN_Model_test.py 对测试集得到每一个问题对应的不同答案的分数。

4）对于上面的两种方法分别得到对应的问题的答案排序，并存入

结果文件中，完成比赛的要求任务

5. 结论

本文提出了一个基于 CNN 卷积神经网络和 wordoverlap 模型联合的章节排序方法。CNN 模型在近几年中已经表现出强大的特征提取能力，在这里能够很好地提取出问题和答案之间的语义关系。Wordoverlap 模型作为基于统计的自然语言处理模型，能够根据问题和答案中的相同词和近义词来衡量它们之间的匹配程度。在最后的联立模型中，我们通过统计问题和答案中的相同词和近义词个数来判断应该使用哪个模型。该联合模型最后可以完成我们的面向智能问答的章节排序系统。并且在我们使用公开的训练数据上进行测试的时候，效果表现较好。