卷积深度神经网络在基于文档的自动问答任务中的应用与改进

傅 健
(复旦大学计算机学院 上海 200433)

摘 要 基于文档的自动问答,尤其是语义匹配,其目标是计算两个文本之间的相似度。这是自然语言处理中的典型任务,并且用以衡量对自然语言的理解程度。深度学习方法得益于可以自动化地学习到给定任务的最优特征表示,在许多研究中取得成功,也包括文本匹配。针对基于文档的自动问答,提出一个基于卷积深度神经网络的语义匹配模型,以便对每一对问题和文档提取特征,并据此计算它们的得分。通过问题和文档之间的交互计算,利用重叠词等文本特征,在中文开放域上的自动问答任务中取得的实际效果证明了该模型的有效性。

关键词 卷积神经网络 自动问答 深度学习 语义匹配 自然语言处理

中图分类号 TP391 **文献标识码** A DOI:10.3969/j.issn.1000-386x.2019.08.031

APPLICATION AND IMPROVEMENT OF CONVOLUTIONAL DEPTH NEURAL NETWORK IN DOCUMENT-BASED QUESTION ANSWERING TASK

Fu Jian
(*School of Computer Science, Fudan University, Shanghai 200433, China*)

Abstract Document-based automatic question answering, especially semantic matching, aims to compute the similarity or relevance between two documents. It is a typical task in NLP and considered as a touch-stone of natural language understanding. Deep learning has been successful in many studies, including text matching, because it can automatically learn the optimal feature representation of a given task. Aiming at document-based question answering, I proposed a structure based on convolution depth neural network to extract features from each pair of questions and documents and calculate their scores. The effectiveness of the model was proved by the interactive computation between questions and documents, as well as the use of overlapping words and other text features in the automatic question answering task in the Chinese open domain.

Keywords Convolutional neural network Question answering Deep learning Semantic match Natural language processing

0 引 言

语义匹配是自然语言处理中非常重要的一个任务。它旨在将多个句子进行建模并计算它们的相似性或相关度,是许多具体应用中非常中心的环节,包括自动问答[1]、答案句子选择[2]、信息检索、释义识别和文本蕴涵[3-4]等。

在自动问答(Question Answering)任务中,对自然语句的建模和理解是非常基本和重要的,其难点通常在于自然语句由于时序和层次关系所带来的复杂结构。在许多神经网络模型中,我们可以用长短时记忆网络 LSTM[5](Long Short-Term Memory),一种循环神经网络 RNN 的改进版本,亦或者是本文使用的卷积神经网络 CNN[6](Convolutional Neural Network),去对句子和句对作建模。此外,通常也会使用双向 LSTM[7]。CNN 对局部的计算选择(local selecting)能力[8],使得它拥有提取输入的有效特征和抽象特征。同时考虑到其便于修改的优点[9-10],所以本文选择使用 CNN 去做句子建模。

收稿日期:2019-01-17。**傅健**,硕士生,主研领域:自然语言处理。

一个好的句子匹配算法,除了需要对自然语句的内部结构进行建模之外,还需要考虑它们之间的交互。我们可以利用这些句对之间丰富的匹配模式信息,来获得更好的匹配得分。所以,在基本的如 ARC-I[11] 这样的句子匹配模型的基础上,提出一种将两个句子进行词和词的对齐的方法(如 ARC-II[11])。除了对齐和共同使用这些句对之外,我们还可以通过相似度匹配[12]达到这样的目的。具体地,我们计算出一个相似度单元,并代入到后续的计算中去。这里需要补充说明的是,对于相似度的计算,可以放在神经网络的上游,也可以放在下游,亦或者上下游同时计算[13]。

在上述基础上,我们也可以增加一些无需使用外部知识的特征,从而配合神经网络,进一步提高任务的表现(见实验),例如 TF-IDF、重叠词指示器[2]、带逆文本频率指数加权(IDF-weighted)的重叠词特征等。

综上所述,我们提出应用了基于卷积深度神经网络的语义匹配模型,不仅通过多层的卷积和池化,利用了句子各自内部独立的结构信息,而且能够捕捉每一对问题和答案之间丰富的模式信息。本文还将神经网络模型应用在了基于文档的中文自动问答任务(NLPCC DBQA Task)上,取得了不错的效果,并且能够配合额外的词重叠等构建的无外部知识特征进一步提升效果。

1 基于卷积的句子模型

受许多卷积网络模型的启发[9],本文基于卷积的句子模型 ConvNet 如图 1 所示,其目的是对每一对问题 query 和文档 document 都能学习出有效的中间特征表示,并用于后续的语义匹配。模型将输入的句子序列,在词嵌入处理之后,再用多层(或一层)卷积层和最大池化层进行信息提取,最终得到一个定长的向量表示,并以此作为特征表示。需要额外说明的是,卷积时可以使用多个过滤器,其数量作为超参进行调节。

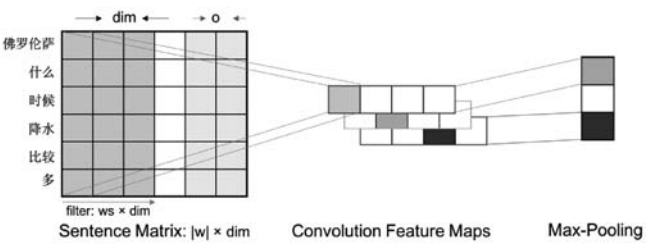

图 1 基于卷积的句子模型结构,将输入句子映射为中间的特征表示

我们的句子模型按顺序由词嵌入层、卷积层、非线性层和最大池化层组成,以下将依次对这 4 个神经网络层作详细介绍。

1.1 词嵌入层

词嵌入层的目的是将原先输入的字或词(可以分词的话)映射到一定维度的表示空间中,从而赋予字或词更丰富的上下文信息或同级信息。常用的词嵌入方法主要为两种,一是 word2vec[14],基于字或词和上下文周边文本的相互预测;二是 GloVe,基于字或词在所有文本中前后出现次数。总而言之,词嵌入都利用的是字或词在上下文文本中丰富的统计信息。具体地,在本文的自动问答任务中,我们首先对问题和答案这两个输入序列先作中文分词之后(利用分词工具,如 jieba 等),再用 word2vec 对序列中的每个词作分布式表示,详细步骤如下:

首先,整个神经网络的输入看作一串词的序列:$[w_1, w_2, \cdots, w_l]$,且每个词均来自词表 $\boldsymbol{V}$。每个词会依据预先训练好的 word2vec 词嵌入,从已有的嵌入矩阵 $\boldsymbol{W} \in R^{|v| \times d}$ 中获得对应的分布式向量 $\boldsymbol{w} \in R^d$。

除了词嵌入表示之外,我们还对句子中的每一个词增加了重叠指示器特征,表示当前词是否在另一个句子中出现。这将作为额外的输入,并参与后续匹配打分的计算(见图 1 模型结构)。最终,我们对问题和文档,都将分别得到句子矩阵 $\boldsymbol{S} \in R^{L \times (d+1)}$。

1.2 卷积层

卷积层即对当前的输入以卷积作为基本操作,进行进一步处理。与全连接(fully connected)网络相比,它们的计算方式相同,只不过卷积的输入为其中的一段定长的窗口大小。通常卷积神经网络的输入为图片,具体地,图片输入可看作是长度、宽度、深度(depth 或 channel)的三围张量,卷积时则用一个或多个拥有长度、宽度的 filter 在图片的每个 channel 上计算。那么应用到自然语言处理任务时,我们可以把输入的 channel 看作 1,而卷积时 filter 的 size 设定为词的 embedding。前面介绍过,卷积的作用可以有效提取局部特征,进一步地,我们可以设置多个卷积窗口大小,从而在局部特征外,得到尽可能更好的全局特征。下面进行详细步骤介绍。

卷积层的目标是提取特征或模式信息,具体地,在给定输入(词级别)序列 $q^{\text{emb}} = r^{w_1}, r^{w_2}, \cdots, r^{w_l}$ 之后,我们定义矩阵 $\boldsymbol{Z}_q = [z_1, z_2, \cdots, z_l]$,每一列包含一个向量 $z_i \in R^{\text{dws}}$,代表序列中 w_s 的词的嵌入表示。在我们对问题 q,用 c 个(窗口大小不同的)filter 进行卷积操作后,得到:

$$\boldsymbol{Q} = W\boldsymbol{Z}_q + b \tag{1}$$

式中:$\boldsymbol{Q} \in R^{l \times c}$ 中的每一行 m 包含了以 q 中的第 m 个词为中心的文本窗口提取出来的特征,W 和 b 为神经

网络训练中待学习的参数。需要说明的是,卷积过滤器的数量为 c,以及词级别文本窗口大小 w_s 均为超参,需要进行手动选择。

接着,我们用相同的方式计算出答案 a 的输出矩阵 $\boldsymbol{A}$。在这里,神经网络可以选择共享参数或相互独立。

1.3　非线性层

为了使得神经网络能够学习到非线性决策边际,从而更好地提取出特征的表示,我们需要在线性的卷积层之后增加一层非线性激活函数 $\alpha()$。需要说明的是,这将会应用在之前卷积处理之后结果的每一位元素上。通常,非线性激活函数有多种选择,如 relu(常见于图像任务重)、sigmoid、tanh 等。在本文中,非线性激活函数默认使用的是双曲正切函数 tanh,它会将原先的输出重新映射到[-1,1]之间。

1.4　最大池化层

在过了卷积层与非线性函数计算之后,我们还需要池化层来将信息进行融合,并且同时将表示进行压缩。在卷积中提取特征时,通常采用最大池化或者平均池化。在本文中,我们选择使用最大池化,即在每一个 filter 中都选取最大值作为输出,以便为后续的语义匹配提供有效信息。

2　匹配模型

整个句对匹配模型的结构如图 2 所示。我们基于卷积的句子模型(前面所述),能够学习将输入的句对各自表示为中间向量表示,进而我们可以继续利用它们去计算相似度。与传统做法类似[8,15]:在我们获得两个句子的矩阵表示之后,将其作为一个多层感知器[16](multi-layer perceptron,MLP)的输入,并得到(分类或分数)输出。

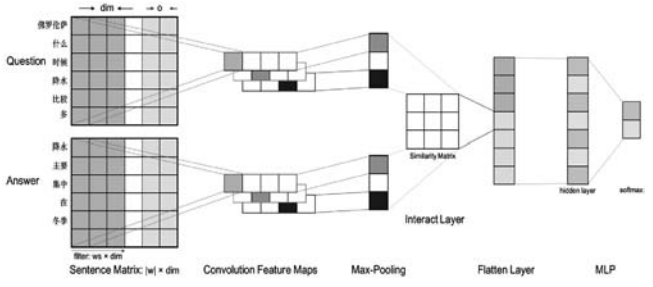

图 2　整体卷积深度神经网络结构图,用以语义匹配任务

但是,这里与之前工作不同的是,我们还可以进一步地提高两个句子之间的交互性,例如计算额外的问题 q 与文档 d 的相似度得分、用 attention 方法对原来的结果进行重新权重上的表示等。

接下来,我们将描述如何使用之前得到的中间特征表示,计算出相似度得分(或称为相似度单元),以及介绍其余的神经网络层。

2.1　交互层

在得到输入句对各自的中间向量表示之后,最常见的做法是将其进行拼接操作后作为输入一起带到后续的网络层(如多层感知器)中计算。但这种做法的一个弊端在于没有充分地提取出它们作为匹配的丰富模式信息,故需要加上交互层,用以将句对间的匹配信息表示出来。具体地,我们可以选择将句对作为输入,额外计算一些表征相似度的中间结果,并一同带入后续网络层。下面进行详细的步骤介绍。

在我们用基于卷积的神经网络句子模型分别对问题 query 和文档 document 进行计算后,得到了中间向量的特征表示 x_{q} 和 x_{d}。通常 x_{q} 和 x_{d} 均为二维张量,第二维维度 d 相同(也可以不同,不同时需改变后面的参数矩阵大小)。据此,我们可以进一步计算句对的相似度得分(或称为相似度单元)。借鉴已有的方法[17],我们可以通过如下方式计算出相似度单元:

$$sim\_score(x_{\mathrm{q}},x_{\mathrm{d}}) = x_{\mathrm{q}}\boldsymbol{M}x_{\mathrm{d}}^{\mathrm{T}} \tag{2}$$

式中:$\boldsymbol{M}\in R^{d\times d}$ 是一个相似度矩阵,作为参数,随着训练进行优化、更新。需要说明的是,在这里可以选择同时计算一个或多个这样的相似度单元(相应地,会用到一个或多个参数矩阵 $\boldsymbol{M}$),从而得到提取到更多的有效信息,带入后续的计算中。

2.2　多层感知器

MLP 是一个常见的基本的网络结构。它基于前向反馈网络,含有多层网络层,包括输入层、中间的隐层、输出层等,每一层均为全连接层,即下一层的输出都依赖上一层的所有神经元。整个多层感知器,最终会得到一个分类或分数输出,之后根据选择的目标函数进行反向传播,更新前面的参数以使得相应的目标函数降低(向梯度下降的方向)。在本文中,对于自动问答任务,我们简单地将其看作是一个 0 -1(二)分类任务,即正确答案看作 1,错误答案看作 0(多个正确语句的前后顺序不影响评价指标的高低)。

具体地,本文使用的 MLP 除了输入层外,还由一层隐藏层,即一层线性全连接层,和一个逻辑回归组成(softmax 层),最终得到一个二值分类的输出结果。隐藏层的计算如下所示:

$$f(\boldsymbol{w}_{\mathrm{h}}x+b) \tag{3}$$

式中:$\boldsymbol{w}_{\mathrm{h}}$ 和 b 为隐藏层的参数向量,$\boldsymbol{w}_{\mathrm{h}}$ 为隐层的权重向量,b 为激活阈值。另外,f 是一个非线性激活函数(默认为 tanh 函数)。

接下里类似地进行一层输出层的计算,但在线性计算后无需跟一个非线性函数。而是对计算的结果进行逻辑回归计算(以得到二值分类),softmax 函数会计算出所有可能标签(是否为正确答案)的概率分布,如下所示:

$$p(y=j|x)=e^{yj}/\sum e^{yk} \quad (4)$$

2.3 训 练

在得到本文自动问答任务中是否为正确答案的概率分布表示之后,需要为模型定义一个训练目标函数或者损失函数。根据具体任务的不同,损失函数可以选择交叉熵损失函数、均方差误差损失函数、最大边际损失、其他距离衡量等。而在选取最终的目标函数之后,我们模型的所有网络层的参数才能进行相应的更新,如使用批量的随机梯度下降(SGD)等优化方法,最终使得整个任务的 performance 提高。

在本文的自动问答任务中,我们将输出看作了 0 -1 二值分类任务,并选择随机梯度下降进行模型参数更新。具体地,整个模型的训练目标为降低交叉熵损失函数为:

$$(y,o)=-(1/N)\sum y_n \log o_n \quad n \in N \quad (5)$$

式中:y 为正确标签答案,o 为预测得分,N 为类别(两类)。模型对每一轮输入的 batch-size 的数据进行如上更新,并进行多轮迭代,最终以 early-stop 的结果得到最优的模型。

3 实验结果与讨论

接下来,我们讨论将上述基于卷积的语义匹配模型应用到具体的自然语言处理任务中去,并加以改进。数据集上,我们选取了中文的自动问答任务,其数据集来源于 NLPCC 的 DBQA 任务(document-based QA Task)。配合其他如词重叠等特征,模型取得了实际有效的结果。

3.1 数据集

我们在开放域上的基于文档的中文自动问答数据集上进行了实验,数据集如表 1 所示。其中,dbqa-train 和 dbqa-test 的 QA-pairs 个数分别为 181 882 和 122 531。经统计,在中文分词后,大多数问题语句的长度在不到 20 个词左右,而大多数答案语句的长度在不到 40 个词左右。

表 1 NLPCC DBQA 任务数据集

数据集	QA-pairs 个数
DBQA-train	181 882
DBQA-test	122 531

3.2 评价指标

关于任务的评价指标,考虑到 DBQA 任务实际上定义为一个排序任务,因此我们选择使用更合适的 Mean Reciprocal Rank(MRR)、Mean Average Precision(MAP)这两种常见的搜索排序指标进行评价,而不采用正确率。其中,以 MRR 指标结果为主。

3.3 词嵌入数据

在词嵌入上,我们用 word2vec 方法在中文 wiki 百科语料上训练了 word embedding,其中包含了超过 230 000 的中文文章。并且值得一提的是,预训练的 word embedding 对任务效果具有很大的帮助。

3.4 结果与讨论

实验结果具体如表 2 所示。整个表格依次展示了:(1) 基于 CNN 的神经网络基准线的效果;(2) 额外增加的计算相似度单元所带来的效果提升;(3) 额外增加特征(词重叠特征)所带来的效果体征;(4) 基于模型并配合额外特征(词重叠特征)所共同带来的效果提升。可以看到,相似度单元和词重叠特征都分别起到了很大的作用。并且,基于卷积的语义匹配模型可以与额外的特征共同配合,得到更好的任务效果。需要说明的是,这几种神经网络的超参并未做过多的调整,后续还有待做更多的探究。

表 2 NLPCC DBQA 数据集上的实验结果

模型	MAP	MRR
基于 CNN 的基本匹配模型	36.41	36.42
+ 相似度单元	59.14	59.21
+ 词重叠特征	77.17	77.24
+ 相似度单元 + 词重叠特征	84.90	84.96

4 结 语

在本文中,我们提出并运用了适用于语义匹配的基于卷积的深度神经网络结构,它不仅考虑了对每个独立句子作层次化建模,同时还提取出它们的匹配模式特征。并且,在开放域上的基于文档的中文自动问答任务中,配合词重叠等其他特征,本模型取得了实际有效的结果。后续仍有待作进一步的研究。

参 考 文 献

[1] Berger A, Caruana R, Cohn D, et al. Bridging the Lexical Chasm: Statistical Approaches to Answer-Finding[C]//Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2000:192 - 199.

(下转第 219 页)

区的瓦斯突出预测。

应用实际的瓦斯突出影响因子数据进行多次重复试验,确定多层 DAE-LSSVM 预测模型的结构参数,与不同网络结构的单层 DAE-LSSVM 相比,证明多层 DAE-LSSVM 模型有较好的降维效果。同 PCA-LSSVM、LLE-LSSVM、LSSVM、BP 神经网络瓦斯突出预测模型对比,证明所提模型预测准确率高,鲁棒性强。

参 考 文 献

[1] 窦林名,何学秋, Ren T,等. 动静载叠加诱发煤岩瓦斯动力灾害原理及防治技术[J]. 中国矿业大学学报,2018,47(1):48 – 59.

[2] 荣海,张宏伟,梁冰,等. 煤岩动力系统失稳机理[J]. 煤炭学报,2017,42(7):1663 – 1671.

[3] 李冬,彭苏萍,杜文凤,等. 煤层瓦斯突出危险区综合预测方法[J]. 煤炭学报,2018,43(2):466 – 472.

[4] 姜福兴,尹永明,朱权洁,等. 基于掘进面应力和瓦斯浓度动态变化的煤与瓦斯突出预警试验研究[J]. 岩石力学与工程学报,2014,33(S2):3581 – 3588.

[5] 李楠,王恩元,Ge M C. 微震监测技术及其在煤矿的应用现状与展望[J]. 煤炭学报,2017,42(S1):83 – 96.

[6] 付华,李海霞,卢万杰,等. 一种改进的极限学习机煤与瓦斯突出预测模型[J]. 传感技术学报,2016,29(1):69 – 74.

[7] 朱志洁,张宏伟,韩军,等. 基于 PCA-BP 神经网络的煤与瓦斯突出预测研究[J]. 中国安全科学学报,2013,23(4):46 – 51.

[8] 付华,代巍. 基于 LLE 与 BA-Elman 的瓦斯涌出量动态预测研究[J]. 传感技术学报,2016,29(9):1383 – 1388.

[9] 谢国民,谢鸿,付华,等. 煤与瓦斯突出预测的 NN-SVM 模型[J]. 传感技术学报,2016,29(5):733 – 738.

[10] Vincent P, Larochelle H, Bengio Y, et al. Extracting and composing robust features with denoising autoencoders[C]// International Conference on Machine Learning. ACM, 2008: 1096 – 1103.

[11] Wang J, Wen Y, Gou Y, et al. Fractional-order gradient descent learning of BP neural networks with Caputo derivative[J]. Neural Networks, 2017, 89:19 – 30.

[12] 闫江伟,张小兵,张子敏. 煤与瓦斯突出地质控制机理探讨[J]. 煤炭学报, 2013,38(7):1174 – 1178.

[13] 王世超,潘凤龙,申健. 煤与瓦斯突出预测敏感指标确定及应用[J]. 煤炭科学技术,2013,41(5):82 – 85.

[14] 唐巨鹏,杨森林,王亚林,等. 地应力和瓦斯压力作用下深部煤与瓦斯突出试验[J]. 岩土力学,2014,35(10):2769 – 2774.

[15] 付华,王馨蕊,王志军,等. 基于 PCA 和 PSO-ELM 的煤与瓦斯突出软测量研究[J]. 传感技术学报,2014,27(12):1710 – 1715.

(上接第 180 页)

[2] Yu L, Hermann K M, Blunsom P, et al. Deep learning for answer sentence selection[EB]. arXiv preprint arXiv:1412.1632, 2014.

[3] Liu P, Qiu X, Chen J, et al. Deep fusion LSTMs for text semantic matching[C]//Proceedings of Annual Meeting of the Association for Computational Linguistics, 2016.

[4] Liu P, Qiu X, Huang X. Modelling interaction of sentence pair with coupled-lstms[EB]. arXiv preprint arXiv:1605.05573, 2016.

[5] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997,9(8):1735 – 1780.

[6] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998,86(11):2278 – 2324.

[7] Santos C, Tan M, Xiang B, et al. Attentive pooling networks[EB]. arXiv preprint arXiv:1602.03609, 2016.

[8] LeCun Y, Bengio Y. Convolutional networks for images, speech, and time series[M]//The handbook of brain theory and neural networks. MIT Press, 1998: 255 – 258.

[9] Kalchbrenner N, Grefenstette E, Blunsom P. A convolutional neural network for modelling sentences[EB]. arXiv preprint arXiv:1404.2188, 2014.

[10] Wang Z, Mi H, Ittycheriah A. Sentence similarity learning by lexical decomposition and composition[EB]. arXiv preprint arXiv:1602.07019, 2016.

[11] Hu B, Lu Z, Li H, et al. Convolutional neural network architectures for matching natural language sentences[C]// Proceedings of the 27th International Conference on Neural Information Processing Systems—Volume 2. MIT Press,2014: 2042 – 2050.

[12] Severyn A, Moschitti A. Learning to rank short text pairs with convolutional deep neural networks[C]//Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2015: 373 – 382.

[13] Yin W, Schütze H, Xiang B, et al. ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs[EB]. arXiv preprint arXiv:1512.05193, 2015.

[14] Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and their Compositionality [J]. Advances in Neural Information Processing Systems, 2013, 26:3111 – 3119.

[15] Bordes A, Glorot X, Weston J, et al. A semantic matching energy function for learning with multi-relational data[J]. Machine Learning, 2014, 94(2):233 – 259.

[16] Bengio Y. Learning Deep Architectures for AI[J]. Foundations and trends in Machine Learning, 2009,2(1):1 – 127.

[17] Bordes A, Weston J, Usunier N. Open question answering with weakly supervised embedding models[C]//Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, 2014:165 – 180.