《人工神经网络》大作业中期报告

周正平
计算机科学与技术系
清华大学
zhouzp15@mails.tsinghua.edu.cn

张钰晖
计算机科学与技术系
清华大学
yuhui-zh15@mails.tsinghua.edu.cn

1 引言

基于知识库的问答系统(Knowledge Based Question Answering, KBQA)致力于解决这样一类问题:

给定问题q，通过以合理的方式查询结构化知识库$\mathcal{K}$，求得答案A。

依具体数据集，A可能由单个或多个答案组成，如下表所示。

QA类型	问题q示例	答案A示例	来源
单个答案	*when was walmart founded?*	*1962*	FREE917
多个答案	*who inspired obama?*	*{Saul Alinsky, Nipsey Russell}*	WEBQUESTIONS

我们的总体计划分为如下几个阶段:

- **阶段1：知识积累**
  通过动手实验对FREEBASE（包含44 million实体、2.9 billion事实）、SPARQL查询语言等基础知识进行系统、全面的了解；对本次实验的benchmark WEBQUESTIONS数据集进行统计分析。
- **阶段2：论文研读、确定选题**
  对SEMPRE、AQQU[1]、STAGG[2]、QUESTIONANSWERINGOVERFB[3]等KBQA系统的论文、代码、中间结果进行全面的分析与解读，并查看AQQU系统在网站上公开的error analysis[4]，提出可改进之处与创新点，确定选题。
- **阶段3：模型设计、基线实现**
  根据选题（基于文本信息的问句关系抽取），结合Memory Network、双向LSTM、Pairwise loss等技术，设计模型，并实现简单的基线。目前项目正进行到这一步，已经获得了初步的结果。
- **阶段4：完整实现、模型调试**
  将阶段3设计的模型完整实现，并完成模型的调参优化等工作。

[1]https://github.com/elmarhaussmann/aqqu
[2]https://github.com/scottyih/STAGG
[3]https://github.com/syxu828/QuestionAnsweringOverFB
[4]https://docs.google.com/spreadsheets/d/1Xou4Z2flX6etan0dhbrLrZ4F7bGHFWRadDJ8WUA-SpY/edit#gid=1282511496

- **阶段5：实验分析、收集数据**
  将阶段4实现的模型通过实验进行实证说明，并收集数据进行分析。
- **阶段6：总结收获、撰写论文**
  将本次工作及实验结果撰写为论文。

2 问题陈述

要对一个问题q进行正确的解答，其核心问题有2：

1. **实体链指（Entity Linking）**：q中涉及了哪些实体?
2. **关系抽取（Relation Extraction）**：q中涉及的实体与答案存在何种关系?

参考此前的state-of-the-art的KBQA系统AQQU、STAGG、QUESTIONANSWERINGOVERFB在WEBQUESTIONS数据集上的错误分析，可以发现，除去数据集中固有的噪声[5]之外，关系抽取乃是目前许多KBQA系统的核心瓶颈。

KBQA系统	平均F1	错误分析
AQQU（Bast et al., 2015）	49.4%	Entity Linking：9.82% **Relation Extraction：41.96%** 其他（包括数据噪声等）：48.22%
STAGG（Yih et al., 2015）	52.5%	Entity Linking：8% **Relation Extraction：35%** 其他（包括数据噪声等）：57%
QUESTIONANSWERINGOVERFB（Xu et al., 2016）	53.3%	Entity Linking：15% **Relation Extraction：50%** 其他（包括数据噪声等）：35%

因此，我们将本次作业的选题定为：**基于文本信息的问句关系抽取**。我们希望借助非结构化的文本数据（如WIKIPEDIA），进一步对KBQA中的问句关系抽取这一环节加以完善。

形式化地，将该问题定义如下：

给定问题q与知识库$\mathcal{K}$，首先根据已有的*Entity Linking*工具找出其中所有可能的实体，构成实体集合E。定义问题q的候选解析集合$C = \{(s, r, o)|s, o \in E \wedge (s, r, o) \in \mathcal{K}\}$。对于任一可能的解析$c \in C$，给出函数$f$，使得$f(q, c)$为$q$与$c$之间的匹配程度。以$argmax_{c \in C} f(q, c)$作为问题$q$的最终解析，并以之查询知识库$\mathcal{K}$，获得最终答案$A$。

本次作业所选用的数据集、评价方案及预期结果如下：

- **数据集**：WEBQUESTIONS，共包含5810组Q-A对，每个问题的答案可能是一个或多个属性或实体的名称。该数据集为上述多个KBQA系统所采用，是公认度较高的benchmark。
- **评价方案**：以系统输出的答案A与WEBQUESTIONS数据集提供的真实答案G[2]者之间的F1值作为系统性能的评价指标。
- **预期结果**：WEBQUESTIONS数据集上的KBQA baseline为SEMPRE系统（$F1 = 35.7\%$），之后陆续得到AQQU（$F1 = 49.4\%$）、STAGG（$F1 = 52.5\%$）、QUESTIONANSWERINGOVERFB（$F1 = 53.3\%$）等多个系统的完善。本次作业希望能结合以上系统的已有结果，在其基础上加以改善，取得F1值的提高。

3 方法

从问题的定义可以看出，Entity Linking这一步可以直接由较为鲁棒的NLP工具加以解决，故而问题的关键在于，**给定问题q和关系r，如何判定q和r的匹配程度**。

[5]该数据集受众筹质量所限，噪声较大，答案不全、错误的情况较为严重。然而，由于其规模较大且贴近实际应用，仍为学术界广泛采用的benchmark。

为什么模型很难将它们准确匹配？因为这里的“关系（Relation）”在概念上太单薄了。我们只能从FREEBASE中获知其名称，但无法推知其更多的语义信息，以及其常见的表达方式。引入自然语言便丰富了关系r中蕴含的语义信息，从而可以更准确地将无法简单匹配的q和r关联起来。

我们定义概念支持语句（Support Sentence）如下：给定事实三元组$c = (s, r, o)$，如果一个在WIKIPEDIA中出现的句子x中同时包含了s和o，则将x加入c对应的支持语句集合中。每个三元组至少有一个支持语句，即由自身的s, r, o中包含的字符拼接而成的序列。如果o是一种属性（如数值、日期等），则所有包含s的语句都被认为是c的支持语句。

当然，并不是所有的支持语句都在相同的程度上体现了三元组$c = (s, r, o)$包含的语义信息。因此，需要利用Attention机制对它们进行软筛选。最终模型设计如下：

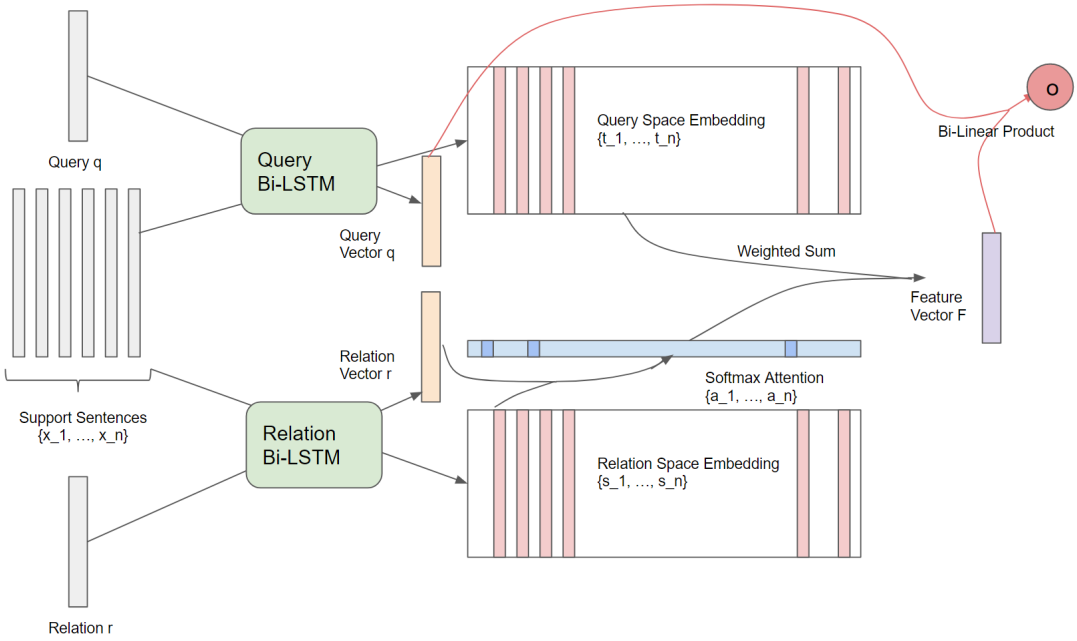

Figure 1: 模型概览

模型的输入为问题q，三元组$c = (s, r, o)$，以及c对应的支持语句集合SupportSentences = $\{x_1, x_2, ..., x_n\}$。输出为一个实数，代表q和c的匹配程度。

1. 使用word embedding将$q, r, \{x_1, x_2, ..., x_n\}$中的所有单词向量化；
2. 使用BiLSTM/LSTM/GRU的最后一个hidden state将r、各个Support Sentence向量化，得到Relation Space中的向量$\vec{r}, \{\vec{s_1}, \vec{s_2}, ..., \vec{s_n}\}$
3. 将各个Support Sentence顺序填入Memory中；
4. 计算r对各个Support Sentence的Softmax Attention $a_i = \frac{\exp(\vec{r}^T A_r \vec{s_i})}{\sum_{k=1}^{n} \exp(\vec{r}^T A_r \vec{s_k})}$；
5. 使用另一个BiLSTM/LSTM/GRU的最后一个hidden state将q、各个Support Sentence向量化，得到Query Space中的向量$\vec{q}, \{\vec{t_1}, \vec{t_2}, ..., \vec{t_n}\}$；
6. 计算r的特征向量$\vec{F} = \sum_{i=1}^{n} a_i \cdot \vec{t_i}$；
7. 计算q和r的相似度$o = \vec{q}^T A_q \vec{F}$
8. 使用F1值作为supervise，采用Pairwise loss $loss = \max(0, 1 - o_{q,r^+} + o_{q,r^-})$进行训练。

4 初步结果

在对数据集与此前的论文进行充分的调研了解后，我们首先采用未加入支持的版本作为基线进行训练。也即该模型的输入仅为q与三元组$c = (s, r, o)$，而不考虑支持语句。

模型中采用双向LSTM对问句q与三元组c中的s, r进行了编码，得到特征向量y_q与y_c，并简单地令$f(q, c) = \cos(y_q, y_c)$作为q与c的匹配程度。模型在WEBQUESTIONS测试集上取得了38.1%的平均F1，与该数据集上的基线SEMPRE系统（$F1 = 35.7\%$）相近，但仍低于其他使用了更为精细复杂的词法学Feature的KBQA系统。

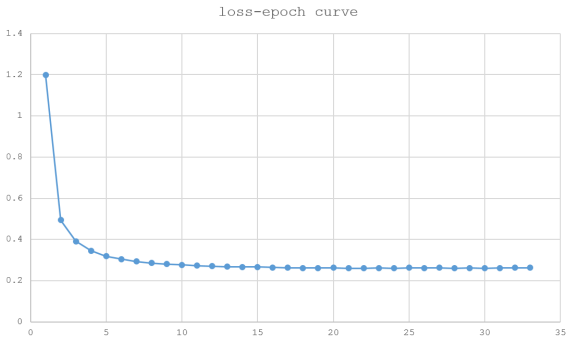

Figure 2: 训练集上loss-epoch曲线

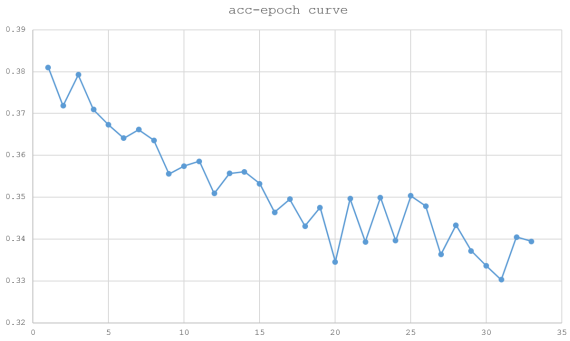

Figure 3: 测试集上acc-epoch曲线

可以发现，该方法存在较为严重的过拟合问题：第一个epoch得到的结果反而是全局最优的。我们会在之后考虑加入正则项，并引入支持语句对模型加以完善。

5 目前的困难

首先，在问题的定义过程中，一个关键的难点在于理清FREEBASE的schema（组织逻辑）。在数据处理过程中，遇到了不计其数的问题，包括服务器内存不足、数据庞大、逻辑晦涩等。其查询工具SPARQL语言亦需要自行学习。但这些问题最终都通过上网查阅教程及动手实验加以克服。详情可以查看这部分的学习笔记。

其次，在方法设计的过程中，我们发现许多三元组在WIKEPEDIA中没有对应的支持语句，而有的存在多个支持语句。前者我们通过规定每个三元组都至少以自己为支持语句完善定义，而后者则通过注意力机制得到解决。这种解决方法的合理性和有效性尚需要实验数据的支持。

最后，在初步实验的过程中，我们发现一个问题对应的三元组数量是极其庞大的，在我们的参数设置`batch_size = 16`的情况下，一个epoch需在GPU上训练45分钟上下。我们可能需要通过观察数据，对候选三元组进行剪枝操作。

参考文献

[1] Bast H, Haussmann E. More accurate question answering on freebase[C]//Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. ACM, 2015: 1431-1440.

[2] Li H, Xiong C, Callan J. Natural Language Supported Relation Matching for Question Answering with Knowledge Graphs[J].

[3] Xu K, Feng Y, Reddy S, et al. Enhancing freebase question answering using textual evidence[J]. CoRR abs/1603.00957, 2016.

[4] Yih S W, Chang M W, He X, et al. Semantic parsing via staged query graph generation: Question answering with knowledge base[J]. 2015.

[5] Xu K, Reddy S, Feng Y, et al. Question answering on freebase via relation extraction and textual evidence[J]. arXiv preprint arXiv:1603.00957, 2016.

[6] Yu M, Yin W, Hasan K S, et al. Improved Neural Relation Detection for Knowledge Base Question Answering[J]. arXiv preprint arXiv:1704.06194, 2017.

[7] Sukhbaatar S, Weston J, Fergus R. End-to-end memory networks[C]//Advances in neural information processing systems. 2015: 2440-2448.

[8] Bordes A, Usunier N, Chopra S, et al. Large-scale simple question answering with memory networks[J]. arXiv preprint arXiv:1506.02075, 2015.

[9] Anything A M. Dynamic Memory Networks for Natural Language Processing[J]. Kumar et al. arXiv Pre-Print, 2015.

[10] Severyn A, Moschitti A. Modeling relational information in question-answer pairs with convolutional neural networks[J]. arXiv preprint arXiv:1604.01178, 2016.

[11] Yih S W, Chang M W, He X, et al. Semantic parsing via staged query graph generation: Question answering with knowledge base[J]. 2015.

[12] Yu M, Yin W, Hasan K S, et al. Improved Neural Relation Detection for Knowledge Base Question Answering[J]. arXiv preprint arXiv:1704.06194, 2017.

[13] Chen D, Fisch A, Weston J, et al. Reading Wikipedia to Answer Open-Domain Questions[J]. arXiv preprint arXiv:1704.00051, 2017.

[14] Berant J, Chou A, Frostig R, et al. Semantic Parsing on Freebase from Question-Answer Pairs[C]//EMNLP. 2013, 2(5): 6.

[15] Yin W, Schütze H, Xiang B, et al. Abcnn: Attention-based convolutional neural network for modeling sentence pairs[J]. arXiv preprint arXiv:1512.05193, 2015.

[16] Golub D, He X. Character-level question answering with attention[J]. arXiv preprint arXiv:1604.00727, 2016.

[17] Berant J, Liang P. Semantic Parsing via Paraphrasing[C]//ACL (1). 2014: 1415-1425.