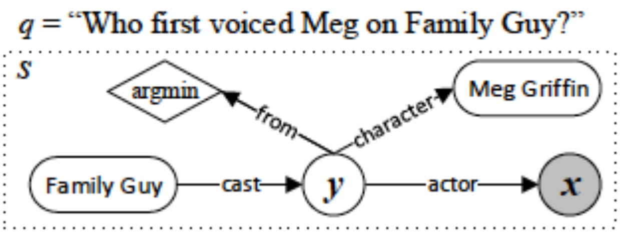
Question Answering with Knowledge Base （KBQA）

Semantic Parsing via Staged Query Graph Generation: Question Answering with Knowledge Base （ACL2015）



(1) EntityLinkingScore(FamilyGuy, "Family Guy") = 0.9
(2) PatChain("who first voiced meg on <e>", cast-actor) = 0.7
(3) QuesEP(q, "family guy cast-actor") = 0.6
(4) ClueWeb("who first voiced meg on <e>", cast-actor) = 0.2
(5) ConstraintEntityWord("Meg Griffin", q) = 0.5
(6) ConstraintEntityInQ("Meg Griffin", q) = 1
(7) AggregationKeyword(argmin, q) = 1
(8) NumNodes(s) = 5
(9) NumAns(s) = 1

Method	Prec.	Rec.	F_1
PatChain	48.8	59.3	49.6
+QuesEP	50.7	60.6	50.9
+ClueWeb	51.3	62.6	51.8

方法：

1、Entity Liking :

S-MART: Novel tree-based structured learning algorithms applied to tweet entity linking. (ACL 2015)，文章证明linking效果对最终效果有明显影响。

2、PatChain：pattern以及关系链得分；QuesEP：question以及整体推断的匹配得分；ClueWeb：使用ClueWeb训练每个sentence两个entity以及可能存在的关系。

3、使用DSSM进行匹配。

4、使用lambda-rank对所有可能的推断进行排序。

数据集：

WEBQUESTIONS dataset

评测脚本：

from http://www-nlp.stanford.edu/software/sempre/.

Question Answering on Freebase via Relation Extraction and Textual Evidence
https://github.com/syxu828/QuestionAnsweringOverFB.

方法：

1、Entity Liking :

数据集：WEBQUESTIONS dataset

An End-to-End Model for Question Answering over Knowledge Base with Cross-Attention Combining Global Knowledge （ACL2017）

方法：

本文提出了Neural Cross-Attention Model旨在针对所有的答案进行排序。每一个候选答案包含这样几个部分：实体、关系、类型、上下文（Answer context is the 1-hop entities and predicates which connect to the answer entity along the answer path.）

Cross-Attention Model分为两个部分：

1、Answer-towards-question(A-Q) attention：Each answer aspect should focus on different words of the same question. 计算每个候选答案的不同部分与问题的得分时候加权query token

2、Question-towards-answer(Q-A) attention：在整合query 和 candidate answer时候对不同的部分得分加权。

数据集：WEBQUESTIONS dataset

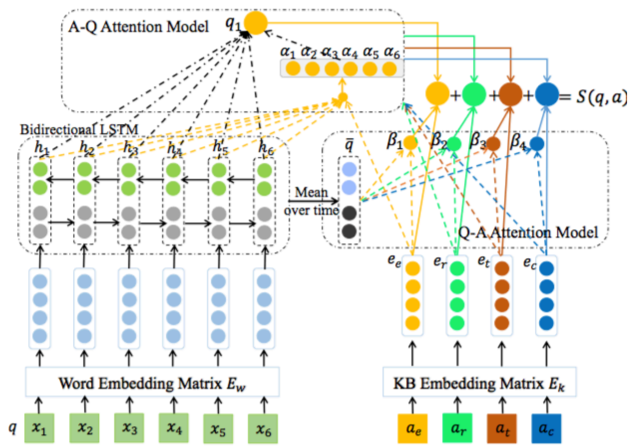


Figure 2: The architecture of the proposed cross-attention based neural network. Note that only one aspect(in orange color) is depicted for clarity. The other three aspects follow the same way.

Methods	Avg F_1
Bordes et al., 2014b	29.7
Bordes et al., 2014a	39.2
Yang et al., 2014	41.3
Dong et al., 2015	40.8
Bordes et al., 2015	42.2
our approach	**42.9**

Table 1: The evaluation results on WebQuestions.

Improved Neural Relation Detection for Knowledge Base Question Answering （ACL 2017）

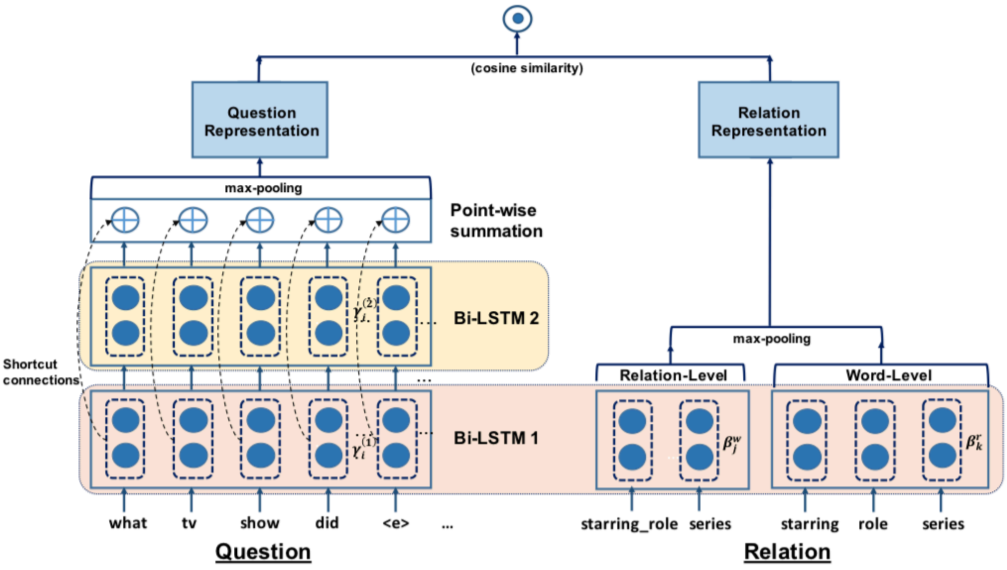
方法：

1、本篇文章首先利用关系对实体进行reranking。

2、其次，采用如下图所示的方法匹配相应关系，在这里topic entity只保留一种。

3、最后结合上述两个得分得到最后的结果。

感觉超参数较多，没有之前ranking 的方法更好；创新的地方在于relation的表示，以及entity的ranking等。



数据集：SimpleQuestions and WebQSP目的是更加偏重于检测relation extraction 的效果。

Model	Relation Input Views	Accuracy	
		SimpleQuestions	WebQSP
AMPCNN (Yin et al., 2016)	words	91.3	-
BiCNN (Yih et al., 2015)	char-3-gram	90.0	77.74
BiLSTM w/ words	words	91.2	79.32
BiLSTM w/ relation names	rel_names	88.9	78.96
Hier-Res-BiLSTM (HR-BiLSTM)	words + rel_names	**93.3**	**82.53**
w/o rel_name	words	91.3	81.69
w/o rel_words	rel_names	88.8	79.68
w/o residual learning (weighted sum on two layers)	words + rel_names	92.5	80.65
replacing residual with attention (Parikh et al., 2016)	words + rel_names	92.6	81.38
single-layer BiLSTM question encoder	words + rel_names	92.8	78.41
replacing BiLSTM with CNN (HR-CNN)	words + rel_names	92.9	79.08

Reading Comprehension Style Question Answering

MACHINE COMPREHENSION USING MATCH-LSTM AND ANSWER POINTER （ICLR 2017）
https://github.com/shuohangwang/SeqMatchSeq

方法：

1、使用Match-LSTM（利用attention将query中words加权），将问题作为前提，将文章作为假设。最终得到文章中所有词语的隐状态。

2、预测：1）选择单词的位置序列（sequence model）2）开始结束的位置（boundary model）

数据集：
SQuAD

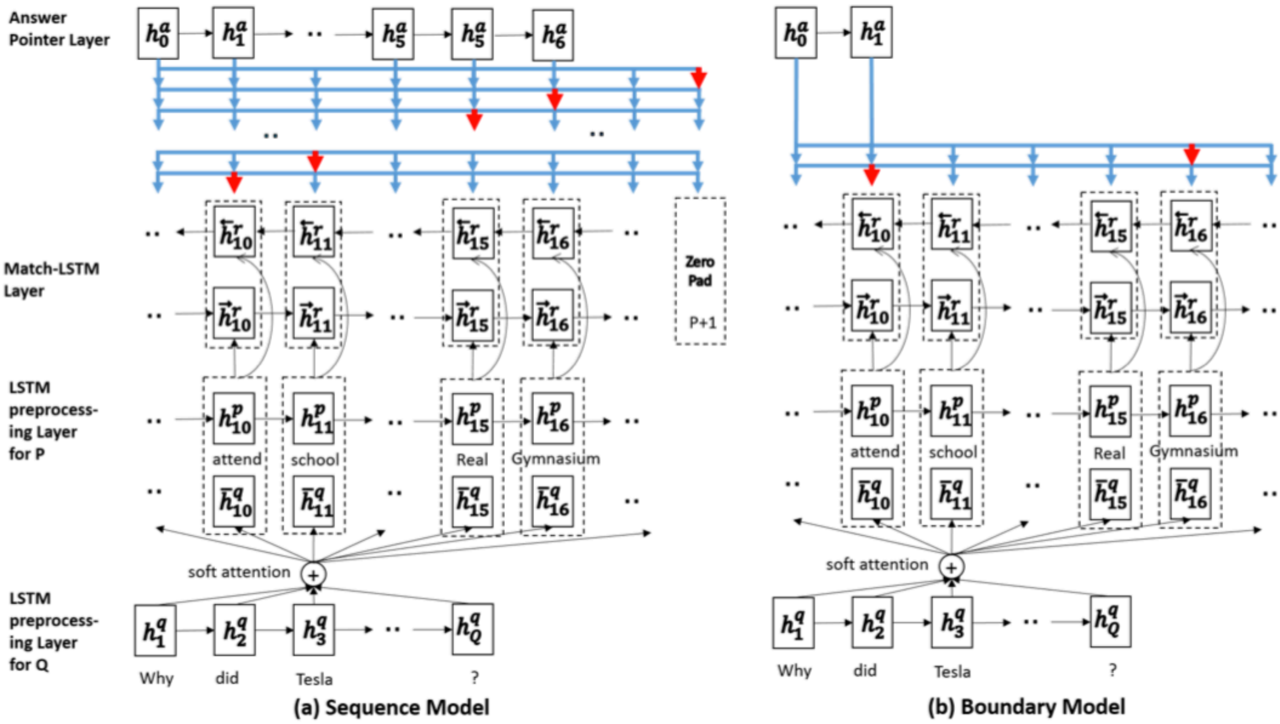


Figure 1: An overview of our two models. Both models consist of an LSTM preprocessing layer, a match-LSTM layer and an Answer Pointer layer. For each match-LSTM in a particular direction, $\bar{h}_i^{\mathrm{q}}$, which is defined as $\mathbf{H}^{\mathrm{q}}\alpha_i^{\mathsf{T}}$, is computed using the α in the corresponding direction, as described in either Eqn. (2) or Eqn. (5).

	l	$\|\theta\|$	Exact Match		F1	
			Dev	Test	Dev	Test
Random Guess	-	0	1.1	1.3	4.1	4.3
Logistic Regression	-	-	40.0	40.4	51.0	51.0
DCR	-	-	62.5	62.5	71.2	71.0
Match-LSTM with Ans-Ptr (Sequence)	150	882K	54.4	-	68.2	-
Match-LSTM with Ans-Ptr (Boundary)	150	882K	61.1	-	71.2	-
Match-LSTM with Ans-Ptr (Boundary+Search)	150	882K	63.0	-	72.7	-
Match-LSTM with Ans-Ptr (Boundary+Search)	300	3.2M	63.1	-	72.7	-
Match-LSTM with Ans-Ptr (Boundary+Search+b)	150	1.1M	63.4	-	73.0	-
Match-LSTM with Bi-Ans-Ptr (Boundary+Search+b)	150	1.4M	**64.1**	**64.7**	**73.9**	**73.7**
Match-LSTM with Ans-Ptr (Boundary+Search+en)	150	882K	**67.6**	**67.9**	**76.8**	**77.0**

Table 2: Experiment Results. Here "Search" refers to globally searching the spans with no more than 15 tokens, "b" refers to using bi-directional pre-processing LSTM, and "en" refers to ensemble method.

BI-DIRECTIONAL ATTENTION FLOW FOR MACHINE COMPREHENSION （ICLR 2017）
BIDAF
allenai.github.io/bi-att-flow/

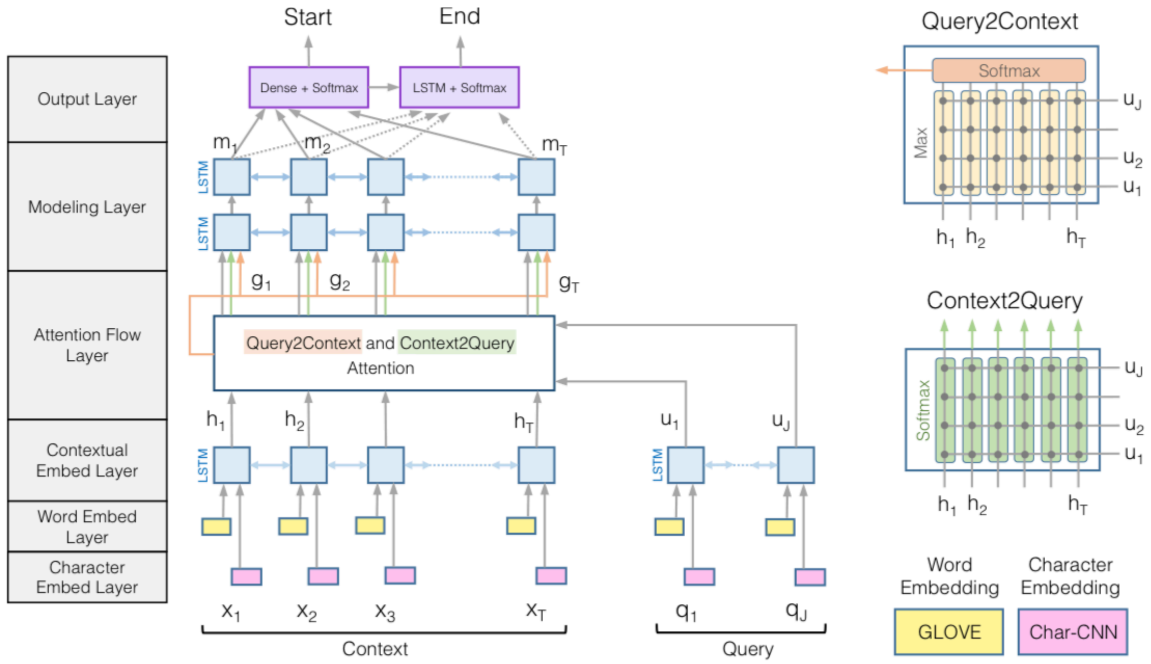


Figure 1: BiDirectional Attention Flow Model *(best viewed in color)*

方法：

1、Character Embedding Layer

2、Word Embedding Layer

3、Contextual Embedding Layer：使用LSTM encode

4、Attention Flow Layer：不同于match-LSTM这里包含两种attention。

1）Context-to-query Attention.类似于match-LSTM

2）Query-to-context Attention.对similar matrix按query维度取max后过softmax

5、Modeling Layer：使用LSTM encode上述信息。

	Single Model		Ensemble	
	EM	F1	EM	F1
Logistic Regression Baseline[a]	40.4	51.0	-	-
Dynamic Chunk Reader[b]	62.5	71.0	-	-
Fine-Grained Gating[c]	62.5	73.3	-	-
Match-LSTM[d]	64.7	73.7	67.9	77.0
Multi-Perspective Matching[e]	65.5	75.1	68.2	77.2
Dynamic Coattention Networks[f]	66.2	75.9	71.6	80.4
R-Net[g]	**68.4**	**77.5**	72.1	79.7
BiDAF (Ours)	68.0	77.3	**73.3**	**81.1**

(a) Results on the SQuAD test set

	EM	F1
No char embedding	65.0	75.4
No word embedding	55.5	66.8
No C2Q attention	57.2	67.7
No Q2C attention	63.6	73.7
Dynamic attention	63.5	73.6
BiDAF (single)	67.7	77.3
BiDAF (ensemble)	72.6	80.7

(b) Ablations on the SQuAD dev set

Table 1: (1a) The performance of our model BiDAF and competing approaches by Rajpurkar et al. (2016)[a], Yu et al. (2016)[b], Yang et al. (2016)[c], Wang & Jiang (2016)[d], IBM Watson[e] (unpublished), Xiong et al. (2016b)[f], and Microsoft Research Asia[g] (unpublished) on the SQuAD test set. Results shown here reflect the SQuAD leaderboard (`stanford-qa.com`) as of 6 Dec 2016, 12pm PST. (1b) The performance of our model and its ablations on the SQuAD dev set. Ablation results are presented only for single runs.

6、Output Layer：通过Modeling Layer（M）首先预测出起始位置，然后放入Bi-LSTM中预测结束位置。

数据集：SQuAD

Gated Self-Matching Networks for Reading Comprehension and Question Answering （ACL 2017）

方法：

1、相比较之前两种方法加入门控机制，类似match-LSTM。

2、Gated Attention-based Recurrent Networks：在match-LSTM输入前加入sigmoid决定passage中各部分的重要程度。

3、Self-Matching Attention：根据上下文推断答案，因此这里就是passage中一个词语与context vector关系。

数据集：SQuAD

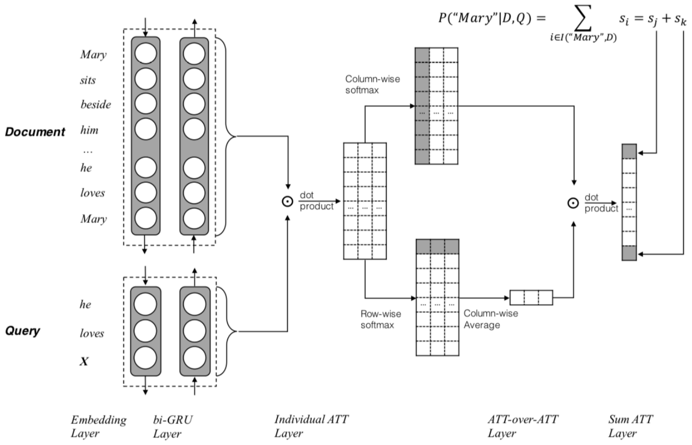
	Dev Set	**Test Set**
Single model	**EM / F1**	**EM / F1**
LR Baseline (Rajpurkar et al., 2016)	40.0 / 51.0	40.4 / 51.0
Dynamic Chunk Reader (Yu et al., 2016)	62.5 / 71.2	62.5 / 71.0
Match-LSTM with Ans-Ptr (Wang and Jiang, 2016b)	64.1 / 73.9	64.7 / 73.7
Dynamic Coattention Networks (Xiong et al., 2016)	65.4 / 75.6	66.2 / 75.9
RaSoR (Lee et al., 2016)	66.4 / 74.9	- / -
BiDAF (Seo et al., 2016)	68.0 / 77.3	68.0 / 77.3
jNet (Zhang et al., 2017)	- / -	68.7 / 77.4
Multi-Perspective Matching (Wang et al., 2016)	- / -	68.9 / 77.8
FastQA (Weissenborn et al., 2017)	- / -	68.4 / 77.1
FastQAExt (Weissenborn et al., 2017)	- / -	70.8 / 78.9
R-NET	**71.1 / 79.5**	**71.3 / 79.7**
Ensemble model		
Fine-Grained Gating (Yang et al., 2016)	62.4 / 73.4	62.5 / 73.3
Match-LSTM with Ans-Ptr (Wang and Jiang, 2016b)	67.6 / 76.8	67.9 / 77.0
RaSoR (Lee et al., 2016)	68.2 / 76.7	- / -
Dynamic Coattention Networks (Xiong et al., 2016)	70.3 / 79.4	71.6 / 80.4
BiDAF (Seo et al., 2016)	73.3 / 81.1	73.3 / 81.1
Multi-Perspective Matching (Wang et al., 2016)	- / -	73.8 / 81.3
R-NET	**75.6 / 82.8**	**75.9 / 82.9**
Human Performance (Rajpurkar et al., 2016)	80.3 / 90.5	77.0 / 86.8

Attention-over-Attention Neural Networks for Reading Comprehension （ACL 2017）

方法：使用attention over attention的方法进行加权，但是稍有所不同。文章利用该种方法对baseline产生的结果进行重排序，预测时候也是预测词语的概率之和。
数据集：CNN news datasets and CBTest NE/CN datasets.

Teaching machines to read and comprehend.

The goldilocks principle: Reading children’s books with explicit memory representaions.



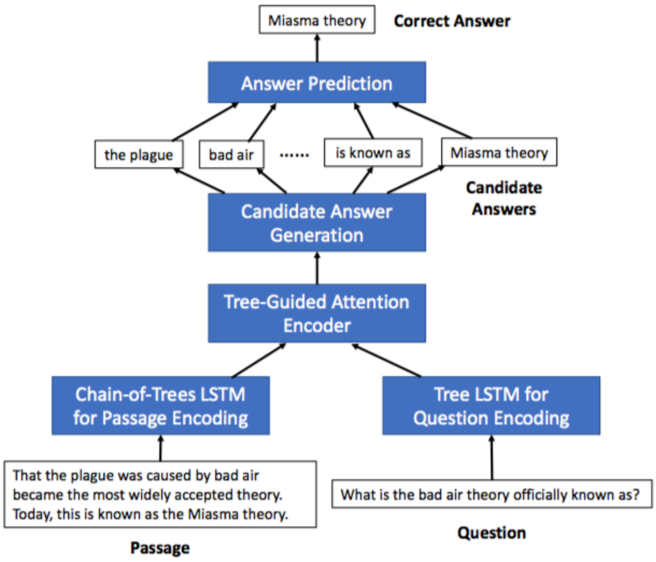
	CNN News		CBTest NE		CBTest CN	
	Valid	Test	Valid	Test	Valid	Test
Deep LSTM Reader (Hermann et al., 2015)	55.0	57.0	-	-	-	-
Attentive Reader (Hermann et al., 2015)	61.6	63.0	-	-	-	-
Human (context+query) (Hill et al., 2015)	-	-	-	*81.6*	-	*81.6*
MemNN (window + self-sup.) (Hill et al., 2015)	63.4	66.8	70.4	66.6	64.2	63.0
AS Reader (Kadlec et al., 2016)	68.6	69.5	73.8	68.6	68.8	63.4
CAS Reader (Cui et al., 2016)	68.2	70.0	74.2	69.2	68.2	65.7
Stanford AR (Chen et al., 2016)	72.4	72.4	-	-	-	-
GA Reader (Dhingra et al., 2016)	73.0	73.8	74.9	69.0	69.0	63.9
Iterative Attention (Sordoni et al., 2016)	72.6	73.3	75.2	68.6	72.1	69.2
EpiReader (Trischler et al., 2016)	**73.4**	74.0	75.3	69.7	71.5	67.4
AoA Reader	73.1	**74.4**	**77.8**	**72.0**	**72.2**	**69.4**
AoA Reader + Reranking	-	-	**79.6**	**74.0**	**75.7**	**73.1**
MemNN (Ensemble)	66.2	69.4	-	-	-	-
AS Reader (Ensemble)	73.9	75.4	74.5	70.6	71.1	68.9
GA Reader (Ensemble)	76.4	77.4	75.5	71.9	72.1	69.4
EpiReader (Ensemble)	-	-	76.6	71.8	73.6	70.6
Iterative Attention (Ensemble)	74.5	75.7	76.9	72.0	74.1	**71.0**
AoA Reader (Ensemble)	-	-	**78.9**	**74.5**	**74.7**	70.8
AoA Reader (Ensemble + Reranking)	-	-	**80.3**	**75.6**	**77.0**	**74.1**

A Constituent-Centric Neural Architecture for Reading Comprehension （ACL 2017）
方法：

1、基于短语匹配，即短语句法树。

不同于现有的线形模型，加入句法信息，缺点难以复现，改进空间不大。

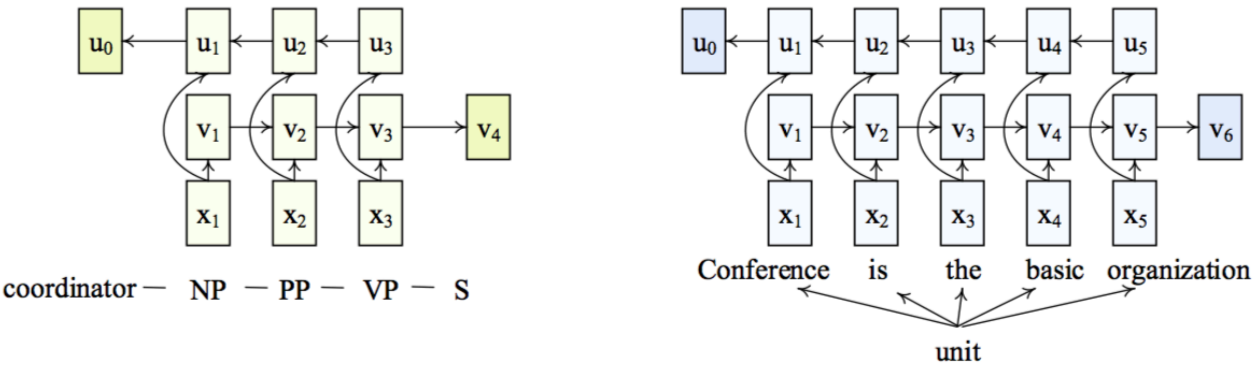
数据集：SQuAD



	Exact Match (EM,%)	F1 (%)
Single model		
Logistic Regression (Rajpurkar et al., 2016)	40.0	51.0
Fine Grained Gating (Yang et al., 2016)	60.0	71.3
Dynamic Chunk Reader (Yu et al., 2016)	62.5	71.2
Match-LSTM with Answer Pointer (Wang and Jiang, 2016)	64.1	73.9
Dynamic Coattention Network (Xiong et al., 2016)	65.4	75.6
Multi-Perspective Context Matching (Wang et al., 2016)	66.1	75.8
Recurrent Span Representations (Lee et al., 2016)	66.4	74.9
Bi-Directional Attention Flow (Seo et al., 2016)	68.0	77.3
Ensemble		
Fine Grained Gating (Yang et al., 2016)	62.4	73.4
Match-LSTM with Answer Pointer (Wang and Jiang, 2016)	67.6	76.8
Recurrent Span Representations (Lee et al., 2016)	68.2	76.7
Multi-Perspective Context Matching (Wang et al., 2016)	69.4	78.6
Dynamic Coattention Network (Xiong et al., 2016)	70.3	79.4
Bi-Directional Attention Flow (Seo et al., 2016)	73.3	81.1
CCNN Ablation (single model)		
Replacing tree LSTM with chain LSTM	63.5	73.9
Replacing chain-of-trees LSTM with independent tree LSTMs	64.8	75.2
Removing the attention layer	63.9	74.3
Replacing tree-guided attention with flat attention	65.6	75.9
CCNN (single model)	**69.3**	**78.5**
CCNN (ensemble)	**74.1**	**82.6**

Structural Embedding of Syntactic Trees for Machine Comprehension （2017 EMNLP）

相比较上面一篇结果稍差，思想类似。



(a) A SECT example

(b) A SEDT example

Two-Stage Synthesis Networks for Transfer Learning in Machine Comprehension （2017 EMNLP）

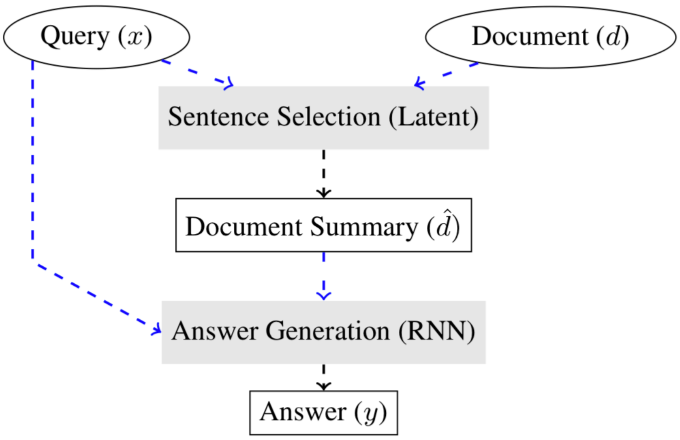
https://github.com/ davidgolub/QuestionGeneration

WIKIREADING: A Novel Large-scale Language Understanding Task over Wikipedia （2016 ACL）

Coarse-to-Fine Question Answering for Long Documents （2017 ACL）

方法：

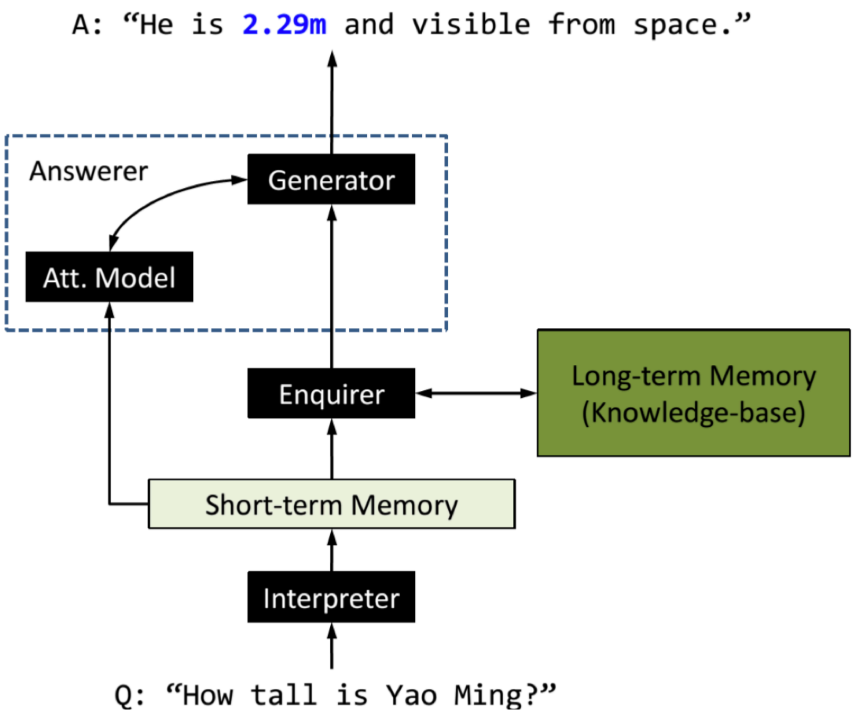
动态选择所需要的句子，并尝试使用增强学习方法训练。



Dataset	Learning	Accuracy
WikiReading Long	First	26.7
	Base	40.1
	Oracle	43.9
	Pipeline	36.8
	SoftAttend	38.3
	Reinforce (K=1)	40.1
	Reinforce (K=2)	**42.2**
Wiki suggest	First	44.0
	Base	**46.7**
	Oracle	60.0
	Pipeline	45.3
	SoftAttend	45.4
	Reinforce (K=1)	45.4
	Reinforce (K=2)	45.8
WikiReading	First	71.0
	Hewlett et al. (2016)	71.8
	Base	**75.6**
	Oracle	74.6
	SoftAttend	71.6
	Pipeline	72.4
	Reinforce (K=1)	73.0
	Reinforce (K=2)	73.9

Generative Question Answering

Neural Generative Question Answering (IJCAI 2016)
https://github.com/jxfeb/Generative_QA（数据）



方法：

模型分为Interpreter, Enquirer, Answerer 三个部分

1、Interpreter：RNN encoder

2、Enquirer：文章提出了两种方案，实际上可以抽象成检索问题。

3、Answerer：这个地方做了一个分类器，用于判断是从vocabulary中选择，还是从KB中选择。

数据（问问爬下来的）

Models	Single	Multi	Mixed
CopyNet	9.7	0.8	8.7
GenQA	47.2	28.9	45.1
COREQA	**58.4**	**42.7**	**56.6**

Table 4: The AE accuracies (%) on real world test data.

Models	Correctness	Fluency	Coherence
CopyNet	0	13.3	3.3
GenQA	26.7	33.3	20
COREQA	**46.7**	**50**	**60**

Table 5: The ME results (%) on sampled mixed test data.

Generating Natural Answers by Incorporating Copying and Retrieving Mechanisms in Sequence-to-Sequence Learning (ACL 2017)
http://www.nlpr.ia.ac.cn/cip/shizhuhe/publications.html

方法：

不同于传统的seq2seq model，在生成答案的时候，来源于vocabulary，source question以及matched KB：

1、Predict-mode

2、Copy-mode

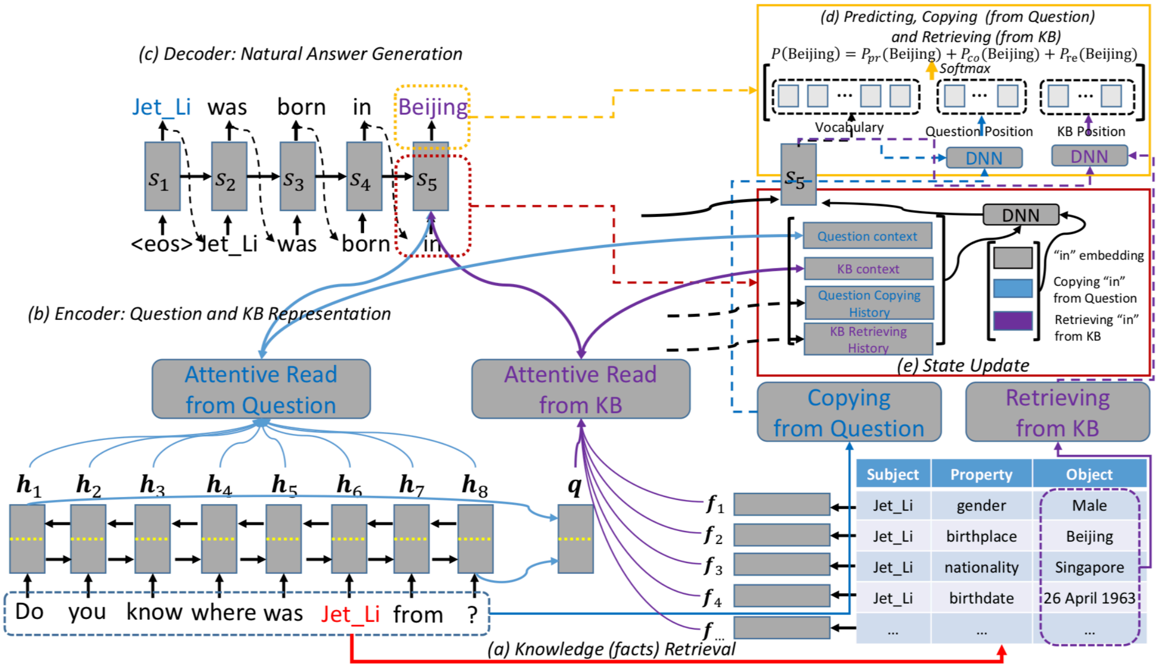


Figure 2: The overall diagram of COREQA.

3、Retrieve-mode

Query Revised

Question Generation for Question Answering（2017 EMNLP）

Learning to Paraphrase for Question Answering （2017 EMNLP）

提出三种改写*Question*的方法，感觉和*ASK THE RIGHT QUESTIONS-ACTIVE QUESTION REFORMULATION WITH REINFORCEMENT LEARNING*没有本质区别。

数据集：*WEBQUESTIONS*，*GRAPHQUESTIONS*，*WIKIQA*

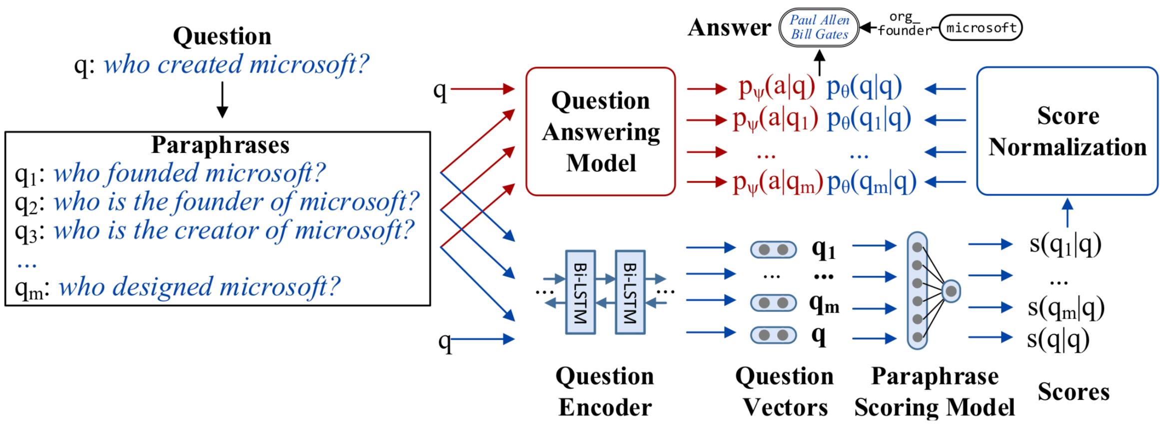


Figure 1: We use three different methods to generate candidate paraphrases for input q. The question and its paraphrases are fed into a neural model which scores how suitable they are. The scores are normalized and used to weight the results of the question answering model. The entire system is trained end-to-end using question-answer pairs as a supervision signal.

Open-domain Question Answering

Reading Wikipedia to Answer Open-Domain Questions (2017 ACL)

EVIDENCE AGGREGATION FOR ANSWER RE-RANKING IN OPEN-DOMAIN QUESTION ANSWERING （2018 ICLR）
方法：
对检索出来的文章中候选答案进行重排序。

Leveraging Knowledge Bases in LSTMs for Improving Machine Reading (2017 ACL)
方法：
使用facts增强QA的效果。

R3: Reinforced Ranker-Reader for Open-Domain Question Answering （2018 AAAI）
方法：
使用增强学习来指导文章的排序。