USTC at NTCIR-12 STC Task

Junbei Zhang, Junfeng Hou, Shiliang Zhang, Lirong Dai

National Engineering Laboratory for Speech and Language Information Processing
University of Science and Technology of China, Hefei, China

Abstract

- **Task: Short Text Conversation (STC)**
- **Team: USTC**
- **System architecture:**
  - Lexical features: Query-Response Similarity, Query-Post Similarity, Transition-p2c
  - Semantic features: EncDec-Forward model, EncDec-Reverse model, Joint-Train model
  - Ranking: linear RankingSVM
- **Results: 0.2867 on Mean nDCG@1, 0.4509 on Mean P+, and 0.4181 on Mean nERR@10**

System Architecture

- We model this task as learning-to-rank problem, and classify the features into two categories: lexical features and semantic features.

Lexical Features

- **Query-Response Similarity**
  - Map query and response to their own TF-IDF score vector
  - Calculate cosine similarity between query and response
- **Query-Post Similarity**
  - Calculate cosine similarity between query and post
- **Transition-p2c**
  - Model the transition probability between post words' vector and response words' vector
  - Example:
    - ☐ Query: Where do we eat?
    - ☐ Query-Response: We eat two apples.
    - ☐ Query-Post: Where do we go?->We go to the bank.
    - ☐ Transition-p2c: Restaurant. (Transition probability between "Where" and "Restaurant", "eat" and "Restaurant" is higher than normal.)
  - Algorithm:

Algorithm 1 Transition-p2c Train

Input: repos-post, repos-comment
Output: transition matrix $\mathbf{T}$

Word segmentation, and get words' vector of post and comment
Initialize: $\mathbf{T} = \text{zeros}(m, n)$, m = length of post vocabulary set, n = length of comment vocabulary set
IDF score of post set and comment set
for (p, c) in (post-word vector, comment-word vector) **do**
  Get tf − idf score vector : **p-tf-idf, c-tf-idf**
  $\mathbf{T} = \mathbf{T} + \mathbf{p\text{-}tf\text{-}idf} \cdot \mathbf{c\text{-}tf\text{-}idf}^T$
end for
Normalization: for i in [0, m], normalize $\mathbf{T}[i]$

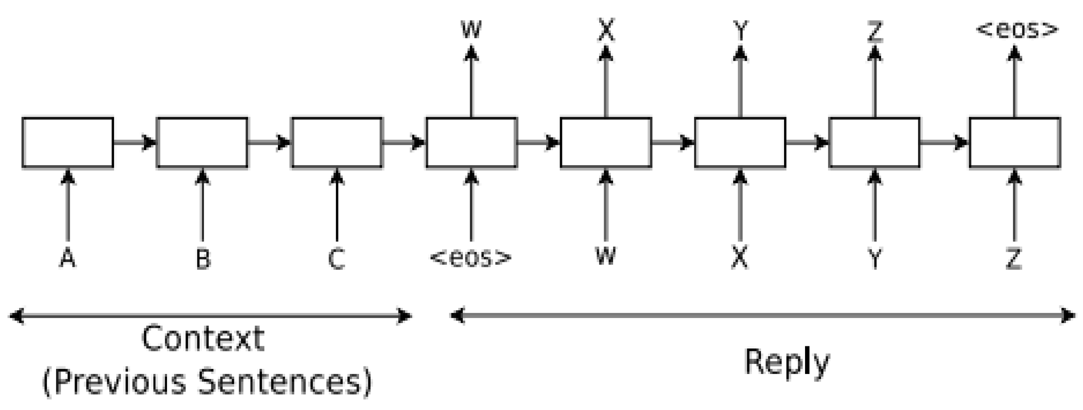
Algorithm 2 Transition-p2c Test

Input: test-query, repos-comment
Output: transition score

Initialize: score = 0, $\mathbf{K} = \text{zeros}(m, n)$
Get tf-idf score vector of test-query and comment
$\mathbf{K} = \mathbf{query\text{-}tf\text{-}idf} \cdot \mathbf{c\text{-}tf\text{-}idf}^T$
for (m, n) in **K**.shapes **do**
  $score = score + \mathbf{K}[m][n] * \mathbf{T}[m][n]$
end for

Semantic Features

- **EncDec-Forward model**
  - Motivated by the work in [Vinyals et al. 2015], [Shang et al. 2015] and [Bahdanau et al. 2014], we use the seq2seq model to estimate P(Response | Post).



- **EncDec-Reverse model**
  - Many-to-many: Unlike the machine translation task, one post in STC may have several comments to fit with, and one comment can also fit with more than one post.
  - Reconstruction: max P(Post | Response)

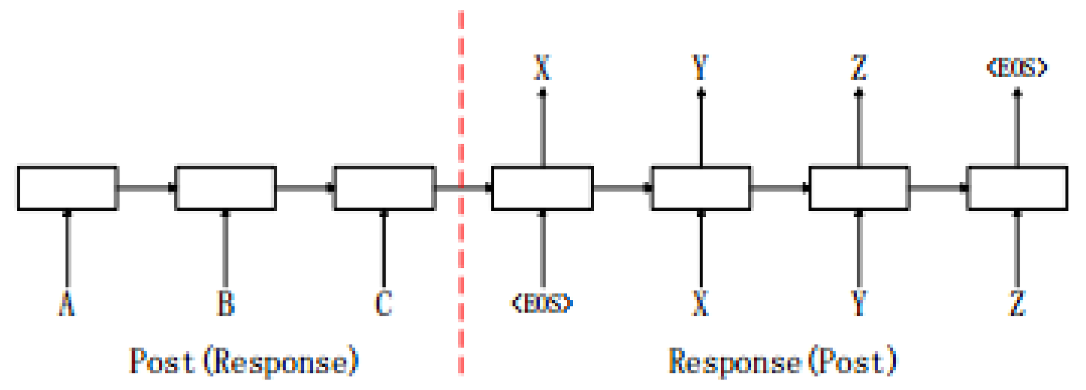


Figure 1: Encoder-Decoder Model for STC

- **Joint-Train model**
  - Generation & Reconstruction: combine P(Response | Post) and P(Post | Response) in one model
  - Decoder1 is regarded as both a decoder and an encoder (Encoder2).

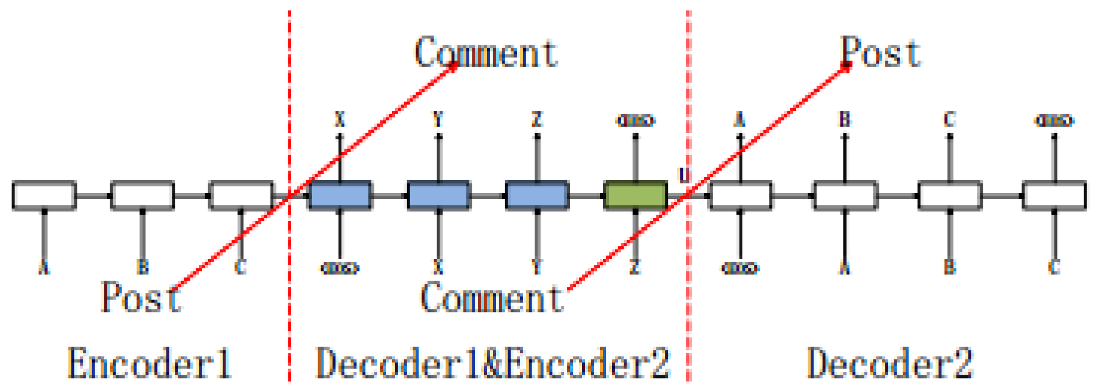


Figure 2: Joint-Train Model

Ranking

- We use linear RankingSVM to merge all the scores and output a final score for each query and response pair.

Experiments

- **We submit 5 runs:**
  - **USTC-C-R1:** Query-Response Similarity + Query-Post Similarity + EncDec-Forward + EncDec-Reverse + Transition-p2c
  - **USTC-C-R2:** Query-Response Similarity + Query-Post Similarity + EncDec-Forward + EncDec-Reverse + JointTrain
  - **USTC-C-R3:** Query-Response Similarity + Query-Post Similarity + EncDec-Forward + Transition-p2c
  - **USTC-C-R4:** Query-Response Similarity + Query-Post Similarity + EncDec-Forward + EncDec-Reverse
  - **USTC-C-R5:** Query-Response Similarity + Query-Post Similarity + EncDec-Forward
- **Results:**
  - Official Results

Table 1: Official STC(Chinese) results

Run	nDCG@1	P+	nERR@10
R5	**0.2867**	**0.4509**	0.4160
R4	0.2767	0.4479	**0.4181**
R1	0.2733	0.4499	0.4169
R2	0.2567	0.4310	0.4001
R3	0.2267	0.4094	0.3848

  - Offline Training Set Results

Table 2: STC(Chinese) training set results

Run	nDCG@1	P+	nERR@10
R5	0.4741	0.6529	0.6327
R4	0.4785	0.6582	0.6395
R3	0.4726	0.6570	0.6347
R2	**0.4889**	**0.6625**	0.6446
R1	0.4859	0.6618	**0.6449**

  - Transition-p2c Case Study

Table 3: Transition score top10 of different word pairs

post words	comment words	transition score
运费 (freight)	代购 (purchasing agents)	0.3207
中型(medium)	谢谢(thanks)	0.1302
警报(alarm)	口水(saliva)	0.1273
元宵节 (Lantern Festival)	快乐 (happy)	0.1260
萌到(sprout)	可爱(lovely)	0.1180
拜年 (pay a New Year call)	新年快乐 (happy new year)	0.1177
王老吉 (Wong Lo Kat)	加多宝 (JDB Beverage)	0.1077
本地(native)	流量(traffic)	0.1066
小家伙(kiddy)	可爱(lovely)	0.1042
张国荣 (Leslie Cheung)	哥哥 (brother)	0.1007

Conclusions

- The results in training set and cases show the efficiency of the models we proposed.
- The online evaluation is inconsistent with the offline evaluation because of the subset selection problem.