Fast Abstractive Summarization with Reinforce-selected Sentence Rewriting

jcykcai

Motivations

- Abstractive models suffer from
  - slow and inaccurate decoding of very long documents.
  - Redundancy (repetitions)

Motivations

- First use an *extractor agent* to select salient sentences.
- Then employs an *abstractive network* to rewrite them (**in parallel**).
- *extractor agent* and *abstractive network* are bridged by **RL** techniques.

Models

Extractor Agent

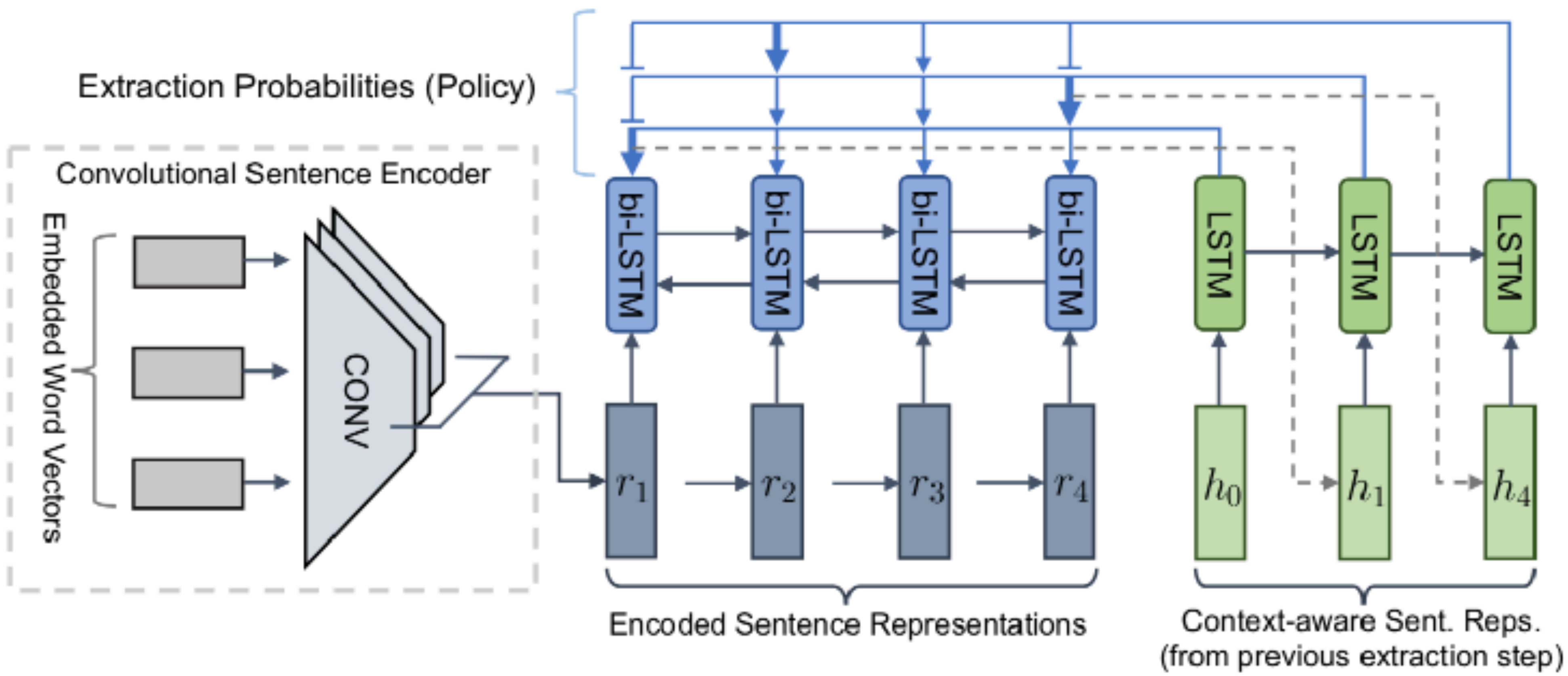


Figure 1: Our extractor agent: the convolutional encoder computes representation r_j for each sentence. The RNN encoder (blue) computes context-aware representation h_j and then the RNN decoder (green) selects sentence j_t at time step t. With j_t selected, h_{j_t} will be fed into the decoder at time $t+1$.

Models

- Abstractor network
- Seq2Seq with attention mechanism and copy mechanism.

Training

- Maximum Likelihood Estimate Pre-training
- RL training of Extractor Agent

MLE Pre-training

- Most of the summarization datasets are end-to-end document-summary pairs without extraction labels for each sentence.

$$j_t = \operatorname{argmax}_i(\text{ROUGE-L}_{recall}(d_i, s_t))$$

RL training

- Reward for extracting the sentence d_{j_t}

$$r(t+1) = \text{ROUGE-L}_{F_1}(g(d_{j_t}), s_t)$$

- Terminal action

  - Terminal Reward $\text{ROUGE-1}_{F_1}([\{g(d_{j_t})\}_t], [\{s_t\}_t]);$

  - Any extraneous, unwanted extraction step receives zero award.

Results

- Faster and better than previous state-of-the-art models.
- Interesting listeners can read the paper.

Unpaired Sentiment-to-Sentiment Translation: A Cycled Reinforcement Learning Approach

jcykcai

Motivations

- Most exiting studies fail in keeping the semantic content.
- A possible reason:
  - They attempts to implicitly separate the emotional information from the semantic information.
  - And it is hard?

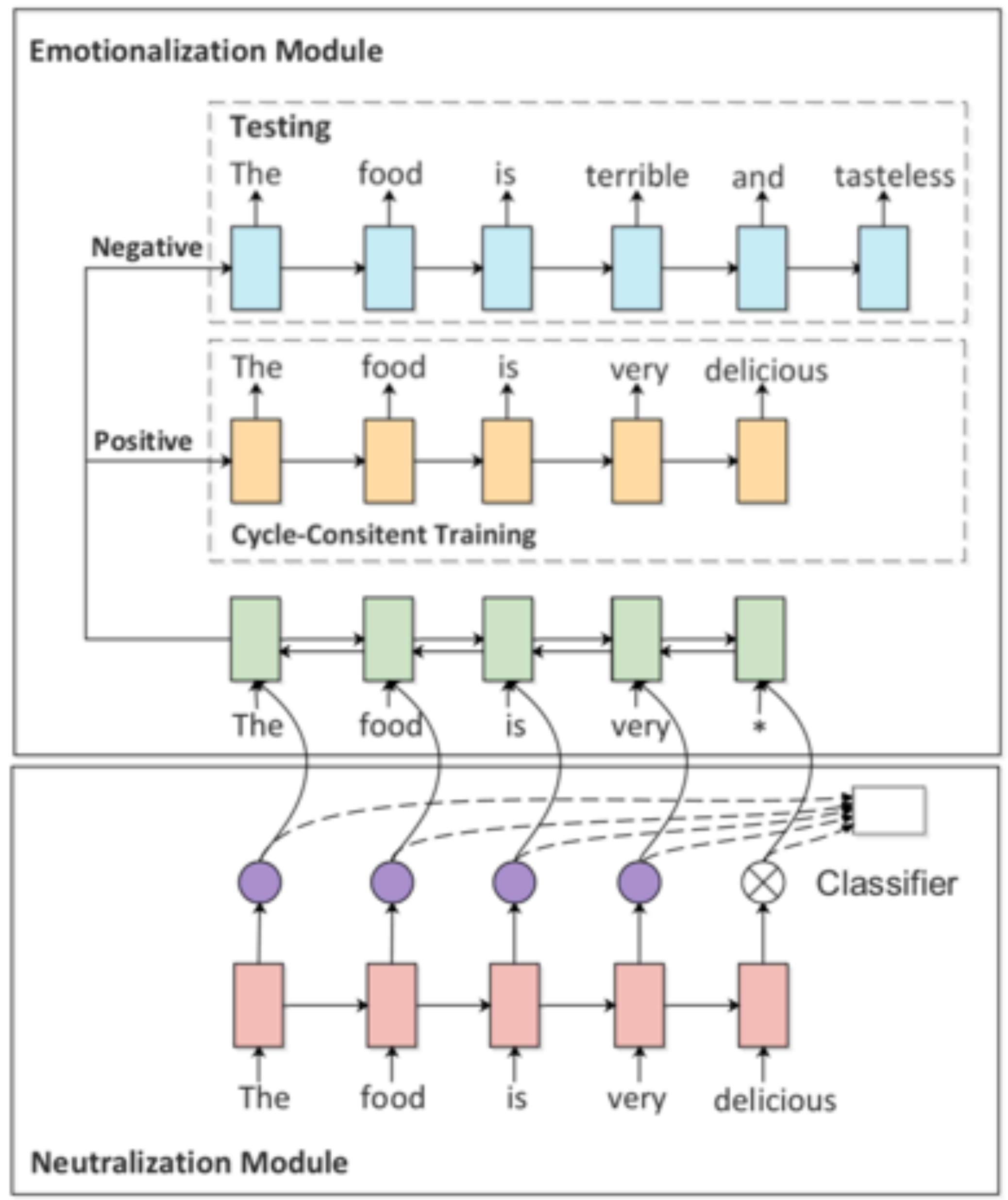
Motivations

- Change the sentiment in two steps:
    - Neutralization
    - Emotionalization

Motivations

- Neutralization module
    - Extracting non-emotional semantic information by explicitly removing emotional words
- Emotionalization module
    - Adding sentiment to the output of neutralization module.

Models



Emotionalization Module
Testing
The
food
is
terrible
and
tasteless
Negative
The
food
is
very
delicious
Positive
Cycle-Consitent Training
The
food
is
very
*
Classifier
The
food
is
very
delicious
Neutralization Module

MLE Pre-training

- Generate labeled data for neutralization module
    - Self attention-based sentiment classifier (SASC)
    - Words with attention weights lower than average are identified as non-emotional words

MLE Pre-training

- Emotionalization Module (Seq2Seq model)
  - Input: neutralized sentence (by SASC)
  - Target: original sentence

RL training

- Generate two sentences, one with the original sentiment and one with the opposite sentiment
- Reward the two sentence ($R_c = R_1 + R_2$)
- Propagate gradients $\nabla_\theta J(\theta) = \mathbb{E}[R_c \cdot \nabla_\theta \log(P_{N_\theta}(\hat{\boldsymbol{\alpha}}|\boldsymbol{x}))]$

RL training

- Reward function: $$R = (1+\beta^2)\frac{2 \cdot BLEU \cdot Confid}{(\beta^2 \cdot BLEU) + Confid}$$

- Confid comes form a pre-trained classier. BLEU is used to measure the content preservation performance

Results

Automatic evaluation

Yelp	ACC	BLEU	G-score
CAAE (Shen et al., 2017)	93.22	1.17	10.44
MDAL (Fu et al., 2018)	85.65	1.64	11.85
Proposed Method	80.00	22.46	**42.38**
Amazon	ACC	BLEU	G-score
CAAE (Shen et al., 2017)	84.19	0.56	6.87
MDAL (Fu et al., 2018)	70.50	0.27	4.36
Proposed Method	70.37	14.06	**31.45**

Results

Human evaluation

Yelp	Sentiment	Semantic	G-score
CAAE (Shen et al., 2017)	7.67	3.87	5.45
MDAL (Fu et al., 2018)	7.12	3.68	5.12
Proposed Method	6.99	5.08	**5.96**
Amazon	**Sentiment**	**Semantic**	**G-score**
CAAE (Shen et al., 2017)	8.61	3.15	5.21
MDAL (Fu et al., 2018)	7.93	3.22	5.05
Proposed Method	7.92	4.67	**6.08**

Input: *I would strongly advise against using this company.*
CAAE: *I love this place for a great experience here.*
MDAL: *I have been a great place was great.*
Proposed Method: *I would love using this company.*

Input: *The service was nearly non-existent and extremely rude.*
CAAE: *The best place in the best area in vegas.*
MDAL: *The food is very friendly and very good.*
Proposed Method: *The service was served and completely fresh.*

Input: *Asked for the roast beef and mushroom sub, only received roast beef.*
CAAE: *We had a great experience with.*
MDAL: *This place for a great place for a great food and best.*
Proposed Method: *Thanks for the beef and spring bbq.*

Input: *Worst cleaning job ever!*
CAAE: *Great food and great service!*
MDAL: *Great food, food!*
Proposed Method: *Excellent outstanding job ever!*

Input: *Most boring show I've ever been.*
CAAE: *Great place is the best place in town.*
MDAL: *Great place I've ever ever had.*
Proposed Method: *Most amazing show I've ever been.*

Results

Michael is absolutely wonderful.
I would strongly advise against using this company.
Horrible experience!
Worst cleaning job ever!
Most boring show i 've ever been.
Hainan chicken was really good.
I really don't understand all the negative reviews for this dentist.
Smells so weird in there.
The service was nearly non-existent and extremely rude.

Take-away

- Easiest way to apply RL in NLP?
  - Decompose a task to several sub-tasks (i.e. build a pipeline method).
  - Evaluate the output in the last step and propagate the reward to all preceding sub-modules.