CS224n Winter 2019

Sentence-Level Extractive Text Summarization

Nathan Zhao, Yi Liu, Yijun Jiang

Introduction

Extractive summarization is the identification of the most relevant sentences in a document which encapsulates its main points. Graph-based, Bayesian, and machine learning have all been applied to this difficult task. Recently, deep learning has also found success in this domain. Here, we investigate a recent end-to-end deep learning framework called NeuSum.

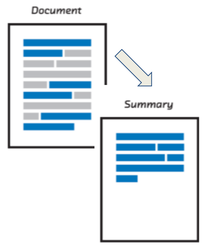

Fig. 1 Cartoon of sentence-level extractive summarization

Model

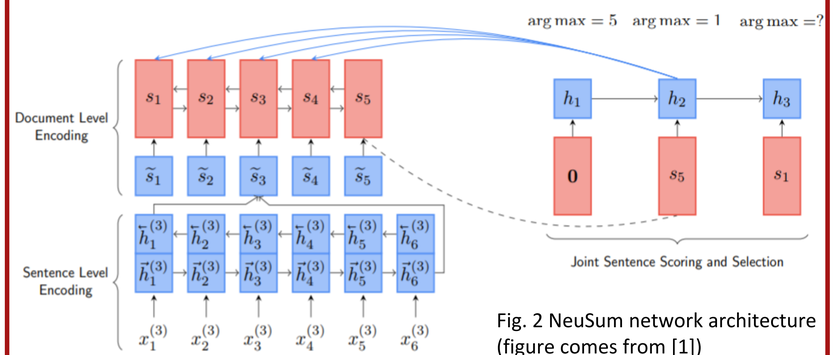

Fig. 2 NeuSum network architecture (figure comes from [1])

The model consists of two parts:

1. Sentence encoder: two BiGRUs that encode sentences on sentence level and then document level.
2. Joint sentence scoring and selection: scores encoded sentences and selects one at each time step. The sentence scores dynamically changes with selection.

Objective function requires evaluation of two distributions at each time step *t*:

Model prediction distribution

$$P_t(S_i) = \frac{\exp(\delta_t(S_i))}{\sum_{k=1}^{L} \exp(\delta_t(S_k))}$$

Reference distribution

$$Q_t(S_i) = \frac{\exp(\tau g_t(S_i))}{\sum_{k=1}^{L} \exp(\tau g_t(S_k))}$$

Loss is the KL-divergence between P_t and Q_t, summed over t.

$$D_{KL}(P_t||Q_t) = -\sum_{S} P_t(S) \log\left(\frac{Q_t(S)}{P_t(S)}\right) \qquad L = \sum_{t} D_{KL}(P_t||Q_t)$$

Data

The Cornell Newsroom dataset is a corpus of 1.3 million documents from 38 different news outlets with abstractive summaries to pair. We siphon off a subsample of 100,000 documents for this project (Fig. 3).

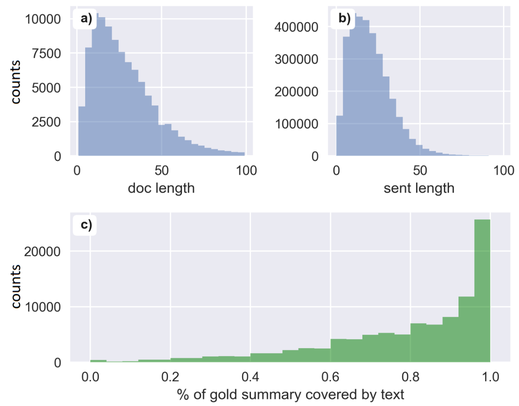

Fig. 3 a) distribution of document length, b) distribution of sentence length in corpus, and c) content overlap of abstractive gold summaries vs. documents

To design an extractive baseline, we test a few algorithms which runs significantly faster than a brute-force combinatorial search (Fig. 4).

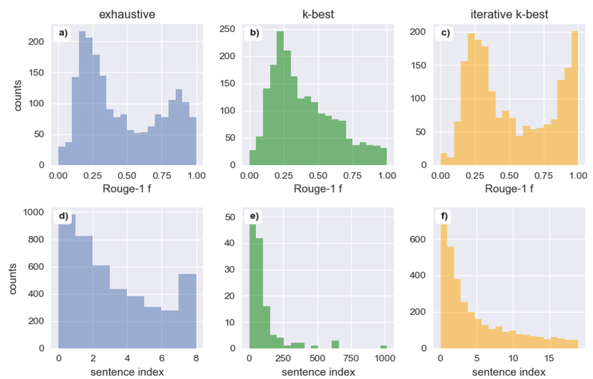

Fig. 4 Statistics on sentence-level extractive reference summary over 2,000 documents

Multi length summaries appear to better optimize Rouge-1 F1 score with the abstractive baseline (Fig. 5).

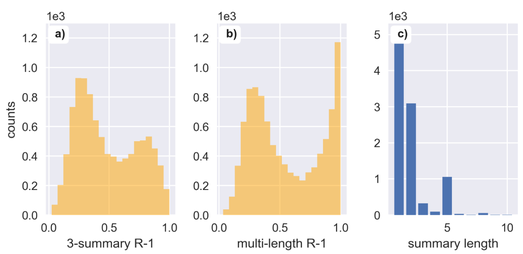

Fig. 5 distribution of extracted summary Rouge-1 F1 scores of a) forced 3-sentence extracted summaries and b) variable length summaries. c) Distribution of summary lengths in b)

Results and Discussion

Table 1: Rouge-1 F1 scores (%) of LEAD-n baselines and n-summary models ($n = 2$ or 3)

Test Dataset	LEAD-2	LEAD-3	2-summary Model	3-summary Model
2-Sent Extraction	49.7	–	38.7	39.4
3-Sent Extraction	–	53.7	40.6	41.7

We trained the fixed-length NeuSum (*n*-summary) model on *n*-sentence extraction data

- For Rouge-1 F1 score (Table 1), NeuSum does not beat LEAD-3.
- Distribution of predicted sentence indices (Fig. 6) with a long tail matches training data well, which is also observed in the original paper.

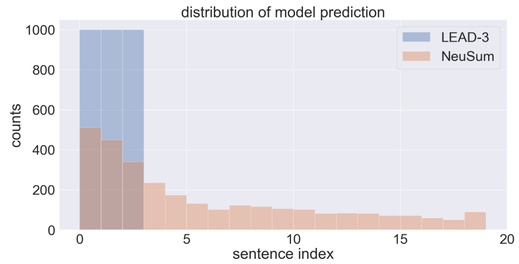

Fig. 6 Distribution of sentence indices extracted by NeuSum

In adaptive-length NeuSum, we allow the model to choose padding sentences. The model learns to pad once the summary reaches optimal length, as <pad> does not count into Rouge.

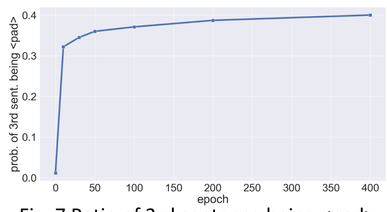

Fig. 7 Ratio of 3rd sentence being <pad> when trained on 2-sentence summaries

We trained NeuSum on 2-sentence data while forcing 3-sentence predictions. The ratio of the 3rd sentence being <pad> increases over the time of training (Fig. 7).

Future Work

- Fully vectorize loss evaluation and train on larger datasets
- Investigate into adaptive NeuSum model

References

[1] Q. Zhou et al. Neural Document Summarization by Jointly Learning to Score and Select Sentences, Proc. of 56th Annual Meeting of ACL, pp. 654–663 (2018)
[2] M. Grusky, M. Naaman, and Y. Artzi, Newsroom: A Dataset of 1.3 Million Summaries with Diverse Extractive Strategies, arXiv:1804.11283 (2018)
[3] Newsroom dataset website: https://summari.es/ (accessed 03/2019)