CS230

Abstractive Summarization for structured conversational text

Ayush Chordia
Department of Computer Science
Stanford University
ayushc@stanford.edu

1 Problem Description

Abstractive summarization is the task of generating a summary comprising of a few sentences that meaningfully captures the important context from given text input. It is one of the known challenging problems in NLP since summarization doesn't involve selecting existing sentences from the input, instead paraphrasing the main contents of the document using vocabulary previously unseen.

2 Dataset and Features

In order to get the baseline metrics, the current model was trained on combination of CNN/Daily Mail dataset as mentioned in Nallapati et al. [1], the dataset itself contains news article (781 tokens on average) paired with multi-sentence summaries (3.75 sentences or 56 tokens on average). In addition to the CNN corpus, transcripts of the earnings call for public companies along with annotated summaries were part of the training set. The annotated meeting conversation from AMI corpus along with their abstractive summaries were also added to training and test set
The goal as part of this project to have diverse representations from CNN corpus, earning call transcripts and recorded meeting conversations to be used in training

The dataset was prepared by first splitting the sentences with Stanford CoreNLP toolkit (Manning et al. [8]) and then pre-processed using the techniques mentioned in See et al. [6].

3 Methods

Neural approaches to abstractive summarization have been previously implemented by using sequence-to-sequence models where an encoder maps sequence of tokens from the source document $x = [x_1, ..., x_n]$ to sequence of continuous representations $z = [z_1, ..., z_n]$ and a decoder generates target summary $y = [y_1, ..., y_m]$ token-by-token

3.1 Pointer Generator Network with Coverage (Model 1)

The model that was currently explored to get the baseline metrics was a pointer generator model as described in See et al. [6]. The model comprises of 3 parts primarily, as described below

a) Baseline Model: Sequence-to-sequence attention model

The tokens of article are fed into an encoder (single layer bidirectional LSTM), producing sequence of encoder hidden states. On each step t, the decoder (single layer unidirectional LSTM) receives the word embedding of the previous word. During training phase, this is the previous word of the reference summary and at test time, its previous word emitted by decoder

b) Pointer-Generator Network

It allows both copying words via pointing and generating words from a fixed vocabulary. For each decoder timestep a generation probability p(gen) $\in$ [0,1] is calculated, that weighs the probability of generating words from vocabulary versus copying words from source

c) Coverage

Repetition is a common problem in sequence-to-sequence models (Tu et al, 2016 [13]) which gets exasperated when generating multi sentence text. The coverage is used to solve this problem by maintaining a coverage vector which is the sum of attention distributions over all previous decoder timesteps.
This coverage vector is an extra input to the attention mechanism's current decision (choosing where to attend to next) is informed by reminder of its previous decisions.

3.2 Pretrained Encoders using BERT (Model 2)

3.2.2 Background for Pretrained Language Models

Pretrained Language models have shown to provide gains in variety of the NLP tasks. The model extends the idea of word embedding by learning contextual representations from large scale corpora. BERT (Devlin et al. 2019 [11] is a new language representation model trained with masked language modeling.

General architecture of BERT is shown in Figure 1. Input text is first preprocessed by inserting two special tokens. Input text is first pre-processed by inserting two special tokens. [CLS] is appended to beginning of text.

Output representation being used to aggregate information from the whole sequence. And token [SEP] is inserted after sentence to make the sentence boundaries.

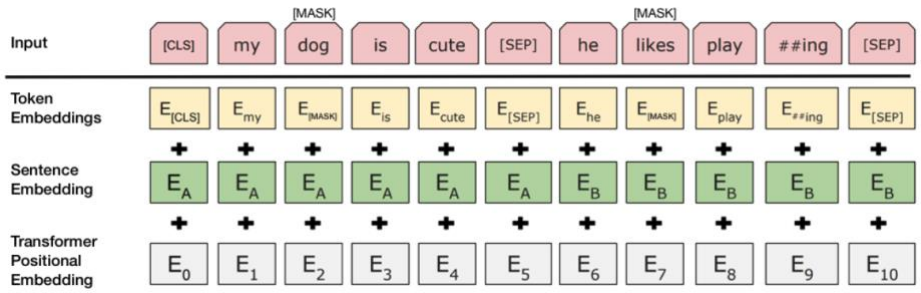

Figure 1: BERT (Devlin et al., 2018 [11]), with modifications

3.2.2 Architecture

Historically, pertained language models have been used as encoders for sentence and paragraph level natural level understanding problems. In this architecture, the impact of language model pertaining was measured on summarization. Architecture is based on Liu et al. [9] where the document level encoder based on BERT is used to encode a document and obtain representation for its sentences. The potential of BERT was explored in the second model by leveraging the encoder-decoder architecture, combining the pertained BERT encoder as described above with randomly initialized 6-layer Transformer decoder as mentioned in Vaswani et al. [12]

In order to account for mismatch between encoder and decoder, as former is pretrained and latter must be trained from scratch, a fine-tuning schedule was used which separates the optimizers of the encoder and decoder

4 Experiments

4.1 Model 1

One of the limitations of the architecture proposed in See et al. [6] was that the article was truncated to 400 tokens during training and test time and limits the length of summary to 100 tokens for training and 120 tokens for testing.

The current model was trained on Quadro P400 GPU with batch size of 16 and trained for 75,000 iterations and it took 3 day 16 hours for the current checkpoint with the 50k vocabulary. As reported in Nallapati et al's [1] paper, it is recommended to train for 35 epochs (600k iterations) to get the best model

4.2 Model 2

I used the Pytorch, OpenNMT (Klein et al., 2017 [14]) and the bert-base-uncased version of BERT, both source and target texts were tokenized with BERT's subwords tokenizer

In the abstractive model, dropout (with probability 0.1) was applied before all linear layers, label smoothing with smoothing factor 0.1 was also used. The transformer decoder has 768 hidden units and hidden size for all feed forward layers is 2048. During decoding,

beam search (size 5) was used and tuned the α for length penalty between 0.6 and 1 on validation set. The model was trained on 2 Tesla P100 GPU and it took 4 days to train the model to 156,000 iterations

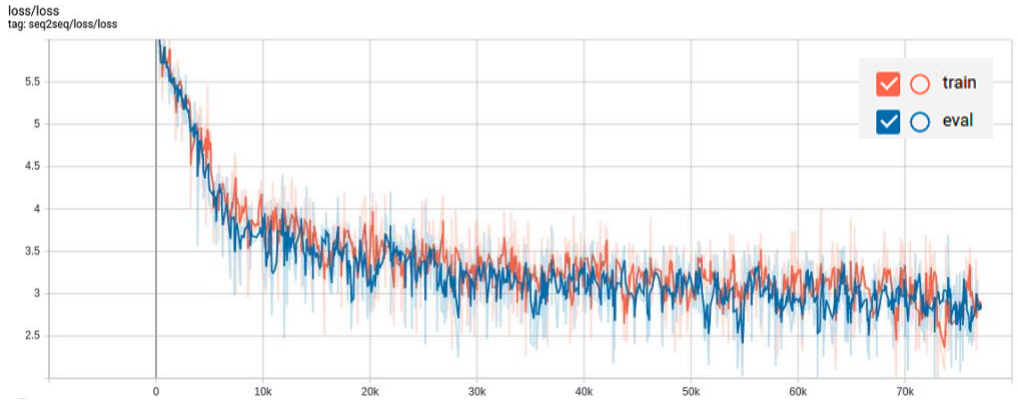

Figure 2 (Model 1): Current seq2seq loss on train and eval data set

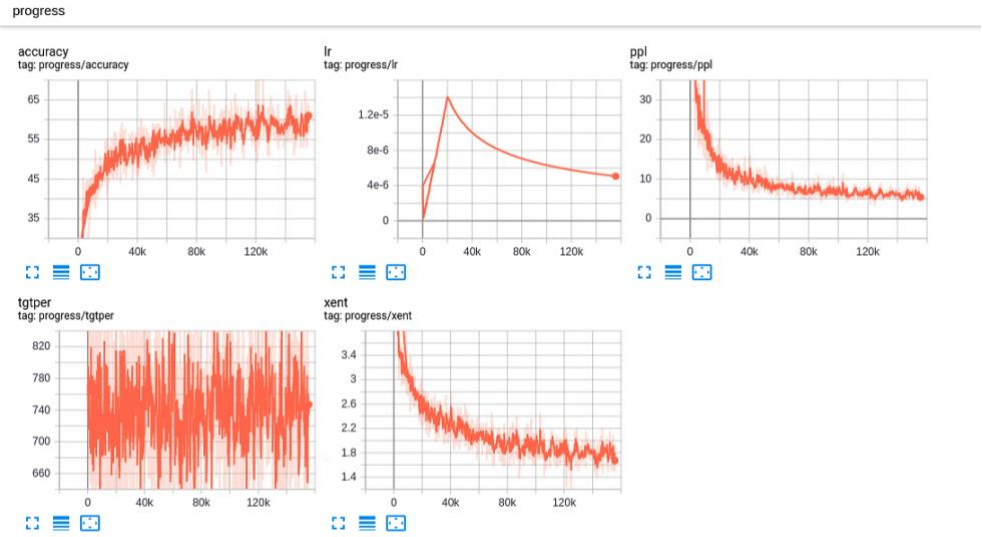

Figure 3 (Model 2): Current metrics on train and validation data set

5 Results

ROUGE score with standard options was used the metric for evaluation. The idea behind Rouge score is to count the number of overlapping unites between generated and referenced summaries.

We plan to report the F-measures of ROUGE-1 (R-1), ROUGE-2 (R-2) (used as a means of assessing information effectiveness using unigram and bigram overlap), ROUGE-L(R-L) (assessing fluency by finding longest common subsequence). The current test set compromised of 12,000 input text and corresponding summaries;
the future experiments will expand this on a larger test set.

5.1 Model 1 Metrics

Metric	Precision	Recall	F1-Score
ROUGE-1	0.3644 with confidence interval (0.3617, 0.3669)	0.3887 with confidence interval (0.3861, 0.3913)	0.3642 with confidence interval (0.3619, 0.3664)
ROUGE-2	0.1558 with confidence interval (0.1535, 0.1580)	0.1654 with confidence interval (0.1629, 0.1678)	0.1552 with confidence interval (0.1530, 0.1572)
ROUGE-l	0.3325 with confidence interval (0.3300, 0.3349)	0.3544 with confidence interval (0.3518, 0.3570)	0.3322 with confidence interval (0.3299, 0.3343)

5.2 Model 2 Metrics:

Metric	Precision	Recall	F1-Score
ROUGE-1	0.38539 (95%-conf.int. 0.38275 - 0.38805)	0.46656 (95%-conf.int. 0.46395 - 0.46908)	0.40948 (95%-conf.int. 0.40718 - 0.41174)
ROUGE-2	0.17642 (95%-conf.int. 0.17404 - 0.17885)	0.21147 (95%-conf.int. 0.20875 - 0.21409)	0.18643 (95%-conf.int. 0.18399 - 0.18882)
ROUGE-l	0.43194 (95%-conf.int. 0.42936 - 0.43445)	0.35720 (95%-conf.int. 0.35457 - 0.35974)	0.37934 (95%-conf.int. 0.37708 - 0.38149)

6 Next Steps

Two different architectures were explored to measure the accuracy of the model on conversational text. Using pretrained language models in the encoder showed considerable performance improvements as seen in the results section over pointer generator network with coverage. The next step would be to make the model robust on training on labeled meeting conversations. As mentioned in Liu et al. [9], the other optimization would be to train the model for higher number of iterations and use a two-stage fine tuning approach where the encoder is first fine-tuned on the extractive summarization task and then further fine tune it on abstractive summarization task. Moreover, the current architecture focused on document encoding for summarization, for future training, I would like to leverage the language generation capabilities of BERT.

References

[1] Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre and Bing Xiang "Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond". In: arXiv preprint arXiv:1602.06023 (2016).

[2] Piji Li, Wai Lam, Lidong Bing and Zihao Wang "Deep Recurrent Generative Decoder for Abstractive Text Summarization". In: arXiv preprint arXiv:1708.00625 (2017).

[3] Li Wiang, Junlin Yao, Yunzhe Tao, Li Zhong, Wei Lu and Qiang Du “A Reinforced Topic-Aware Convolutional Sequence-to-Sequence Model for Abstractive Text Summarization”. In: arXiv preprint arXiv:1805.03616 (2018).

[4] Wojciech Kryscinski, Romain Paulus, Caiming Xiong, Richard Socher “Improving Abstraction in Text Summarization”. In: arXiv preprint arXiv:1808.07913 (2018).

[5] Romain Paulus, Caiming Xiong, Richard Socher “A Deep Reinforced Model for Abstractive Summarization”. In: arXiv preprint arXiv:1705.04304 (2017)

[6] Abigail See, Peter J Liu, Christopher D Manning “Get to the Point: Summarization with Pointer Generator Network”. In: arXiv preprint arXiv:1704.04368 (2017)

[7] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In International Conference on Learning Representations

[8] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pages 55–60, Baltimore, Maryland.

[9] Yang Liu, Mirella Lapata “Text Summarization with Pretrained Encoders”. In: arXiv preprint arXiv:1908.08345 (2019)

[10] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 2227–2237, New Orleans, Louisiana.

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota.

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Advances in Neural Information Processing Systems, pages 5998–6008.

[13] Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. In Association for Computational Linguistics.

[14] Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In Proceedings of ACL 2017, System Demonstrations, pages 67–72, Vancouver, Canada

Appendix (Sample of input text and generated summary)

Sample summaries generated by the two models that were trained using different architecture

1 Input text (truncated)

Q3 was another great quarter at Google with strong revenue growth driven by mobile search YouTube and Cloud. We celebrated Google 21st birthday this quarter. While our mission to organize the world's information and make it universally accessible and useful hasn't changed. We have evolved from a Company that helps people find answers to a Company that helps you get things done.

Since the beginning, we've always invested in tackling deep computer science problems that can have a significant impact on society. The chance to be part of these fundamental engineering challenges is why so many people want to work at Google. In just the last week, we've announced two significant advances. First powered by our long-term investment in AI, we dramatically improved our understanding of the questions people ask Google search. It's the biggest leap forward for search in the past five years. It's all possible because of a new type of neural network-based technique for natural language processing

1.1 Generated Summary

Model 1:

q3 was another great quarter at google with strong revenue growth driven by mobile search youtube and cloud . the chance to be part of these fundamental engineering challenges is why so many people want to work at google

Model 2:

we celebrated google 21st birthday this quarter. we have evolved from a company that helps people find answers. it's the biggest leap forward for search in the past five years. retailers are also turning to cloud solutions to make the shopping experience as seamless and personalized as possible

2 Input text (truncated)

We're thrilled to report Apple's biggest quarter ever, which set new all-time records in both revenue and earnings. We generated revenue of $91.8 billion, which is above the high end of our guidance range, with revenue growth accelerating for the third consecutive quarter. Geographically, we set all-time records in the Americas, Europe and Rest of Asia Pacific, and solid Greater China returned to growth. Our record performance was fueled by iPhone where December quarter revenue was up 8% year-over-year and by our fifth consecutive quarter of double-digit growth outside of iPhone, including a new all-time record for Services and another blowout quarter for Wearables.

Our active installed base of devices has now surpassed 1.5 billion, up over 100 million in the last 12 months alone reaching a new all-time high for each of our main product categories and geographic segments. Not only is our large and growing installed base of powerful testament to the satisfaction engagement and loyalty of our customers, but it's also fueling our growth across the board, particularly in Services.

2.1 Generated Summary

Model 1:

apple 's biggest quarter ever , which set new all-time records in both revenue and earnings .we generated revenue of $ 91.8 billion , which is above the high end of our guidance range , with revenue growth accelerating for the third consecutive quarter .our active installed base of devices has now surpassed 1.5 billion , up over 100 million in the last 12 months alone.

Model 2:

we generated revenue of $91.8 billion , which is above the high end of our guidance range. our record performance was fueled by iphone where december quarter revenue was up 8-year and by our fifth consecutive quarter of double-digit growth outside of iphone. active base of devices has now surpassed 1.5 billion in the last 12 months alone reaching a new all-time high for each of our main product categories and geographic segments

3 Input text (truncated)

We are off to a strong start in fiscal 2020, delivering $33 billion in revenue this quarter. Our Commercial Cloud business continues to grow at scale, as we work alongside the world's leading companies to help them build their own digital capability. Microsoft provides a differentiated technology stack spanning application infrastructure, data and AI, developer tools and services, security and compliance, business process, productivity

For each of these areas -- first, each of these areas represents secular long-term growth opportunity. Second, we are delivering best-in-class innovation and openness in each layer. And third, we offer unparalleled integration and architectural coherence across the entire stack to meet the real world needs of our customers. Now I'll briefly highlight how we are accelerating our progress in innovation, starting with Azure. Organizations today need a distributed computing fabric to meet their real world operational sovereignty and regulatory needs.

3.1 Generated Summary

Model 1:

microsoft provides a differentiated technology stack spanning application infrastructure, data and ai, developer tools and services, security and compliance , business process , productivity , and collaboration .

organizations today need a distributed computing fabric to meet their real world operational sovereignty and regulatory needs.

Model 2:

we are off to a strong start in fiscal 2020 , delivering $33 billion in revenue this quarter. we are working with the world 's leading companies to help them build their own digital capability. microsoft provides a distinguished technology stack spanning application infrastructure , data and ai, developer tools and services. the quintessential characteristic of every application going forward will be ai, and we have the most comprehensive portfolio of ai tools and infrastructure and services

4 Input text (truncated)

We delivered a solid quarter against a challenging macro environment. While we're pleased with this performance, we're most focused on the environment as we move forward. We'll discuss this more in a moment. What's happening inside Cisco regardless of the macro is an unrelenting focus on driving innovation, transforming our business, and exceeding our customers' expectations. In Q1, as you've seen, we had revenue growth of 2% and double-digit non-GAAP earnings-per-share growth.

We also delivered strong non-GAAP gross margins and non-GAAP operating margins along with solid operating cash flow. We continued to invest in innovation and expand our market opportunities, while maintaining our commitment to maximizing shareholder return.Over the last year, many of you have heard me talk about the resilience of the global macro environment. However, on our last earnings call, we indicated that we had begun to see some weakness and that weakness continued throughout Q1 and was more broad based. While the main challenges continue to be service provider and emerging markets, this quarter we also saw relative weakness in enterprise and commercial.

4.1 Generated Summary

Model 1:

we 're pleased with this performance , we 're most focused on the environment as we move forward . in q1, we also delivered strong non-gaap gross margins and non-gaap operating margins along with solid operating cash flow. we also saw relative weakness in enterprise and commercial.

Model 2:

we delivered a solid quarter against a challenging macro environment. but on our last earnings call , we indicated that we had begun to see some weakness and that weakness continued throughout q1 and was more broad based. the main challenges continue to be service provider and emerging markets