Contrastive Learning for Abstractive Summarization with Billsum

Kolby Devery, Redwan Hussain, Michael Townsend
UC Berkeley
kdevery@berkeley.edu, rh1330@berkeley.edu, miketown@berkeley.edu

Abstract

This paper is an exploration of abstractive summarization with contrastive learning for legal text on the BillSum dataset. Previous summarization approaches to the Billsum dataset have shown a range of ROUGE-1 performance, with the highest being .486 in the Kornilova, A., Eidelman, V. (2019) paper. However, none of these are attempting to address *exposure bias*, which is the gap between the training metric (MLE) and evaluation metric (ROUGE). Contrastive learning helps to minimize this bias by producing multiple candidate summaries beyond the golden summary that the model can reference during training or inference to better train the model in distinguishing relative quality beyond just the golden summary. We present two contrastive learning approaches - one using the BRIO approach referenced in the Liu et al. (2022) paper and another inspired by both the BRIO method and the SimCLS paper by Liu et al. (2021). These methods scored .493 and .488 (ROUGE-1) respectively, demonstrating enhanced performance over other BillSum abstractive summarization approaches.

1 Introduction

Summarization can be useful in the context of legislation and law as legal documents can be notoriously long and complex, making them difficult for humans to digest and comprehend. Reliable and fast summarization can quickly and accurately extract key points from legislation, making it much easier for lawyers, lawmakers, and the general public to understand what the law entails. This can save not only time and effort, but also reduce the risk of misinterpretation or misunderstanding. Additionally, this can help streamline legal processes, allowing for more efficient and effective decision-making.

Abstractive summarization is a challenging task in natural language processing, especially when it comes to summarizing US legislation due to its length, complexity, and technical jargon. Traditional extractive methods fail to capture the main ideas and context, resulting in summaries that are often incomplete and do not convey the intended meaning accurately. Abstractive summarization, on the other hand, generates summaries that go beyond mere copy-pasting of sentences from the original text and instead focus on understanding and summarizing the main ideas in a concise manner.

Many papers have focused on legislative summarization, but have not implemented state-of-the-art techniques to address and mitigate *exposure bias*, which is the training-inference discrepancy caused by teacher forcing in maximum likelihood estimation (MLE) training for autoregressive neural network language models. One approach to combat this discrepancy is to generate several reference summaries rank-ordered by the evaluation metric and incorporate their scores into the training process, hopefully making a more robust inference that is not distorted by a single reference summary. This type of approach can be categorized as *contrastive learning*. Contrastive learning has emerged as a promising approach for training deep neural networks in various natural language processing tasks, including abstractive summarization. By comparing and contrasting multiple inputs and learning from their differences, contrastive learning enables the model to capture more nuanced and subtle relationships between the inputs, leading to improved performance in downstream tasks.

In this paper, we present two state-of-the-art approaches for abstractive summarization of US legislation using contrastive learning. Our methods involve training a transformer-based neural network using a contrastive loss function that encourages the network to distinguish between an array of positive and negative candidate summaries.

2 Background

The Kornilova, A., Eidelman, V. (2019) paper introduced the BillSum dataset with an array of summarization approaches that do not address exposure bias, so can act as a benchmark for future methods. Leveraging tools like TextBasic and SumBasic, it achieved its highest ROUGE-1 score of .486, which can act as a performance baseline our methods will attempt to surpass.

One previous approach to addressing exposure bias is called SimCLS, a method proposed by Zhang et al. (2021) that utilizes a contrastive learning framework to generate diverse and high-quality summaries. The method consists of two main stages: (1) pre-training a language model on a large corpus of text to generate candidate summaries with MLE loss (2) introducing an evaluation model that is capable of selecting the best candidate. This evaluation model uses a technique that assigns scores to each candidate summary based on its cosine similarity to the source document and trains to this using contrastive learning.

SummaRanker is a method proposed by Li et al. (2022) that simply generates multiple summaries by ranking sentence clusters according to their relevance to the input document. The method first clusters the sentences in the input document using a pre-trained clustering model, and then generates multiple summaries by selecting the top-ranked clusters.

Traditionally, Maximum Likelihood Estimation (MLE) assumes a deterministic distribution where the reference summary usually receives all the probability mass. The BRIO method, proposed by Li et al. (2022), on the other hand, assumes a *non-deterministic* distribution where system-generated summaries also receive probability mass according to their quality. The contrastive loss in the BRIO method encourages the order of model-predicted probabilities of candidate summaries to be coordinated with an actual *quality metric* by which the summaries will be evaluated. As a result, the model has not been solely trained to minimize MLE loss on just a single golden summary. This has shown state-of-the-art performance, addresses exposure bias, and is a technique we will utilize in this paper.

3 Methods

Our proposed approach involves fine-tuning pre-trained transformer models for the task of summary generation, creating candidate summaries using different techniques, and employing contrastive learning with an additional RankingLoss function to further refine the generated summaries. For our experiments, we have chosen BART, Pegasus, BRIO BART, and BRIO Pegasus as they are widely recognized for text generation tasks.

The bill-sum corpus consists of legal documents, such as bills and statutes, that are challenging to summarize due to their length and complexity. Our goal is to generate a summary of this passage that captures its key information in a concise and readable format. Our approach is based on the intuition that pre-trained transformer models can be fine-tuned for specific text generation tasks, and that different decoding strategies can improve the quality and diversity of the generated summaries. We also use contrastive learning to encourage the generated summaries to be more robust and informative.

To implement our approach, we first perform exploratory data analysis (EDA) to understand the nature of the bill-sum corpus and identify any potential challenges for summary generation. We then establish our baseline by using a pre-trained SBERT model to perform extractive summarization. Next, we fine-tune the BART and Pegasus models using the US train data, and generate candidate summaries using a combination of beam search, diverse beam search, and top-k sampling. We then use a combination of Contrastive and MLE Loss functions to encourage the generated summaries to reference these candidate summaries during training. Our first approach we will refer to as 'BRIO BART' - inspired by the Li et al. (2022) BRIO paper, and 'In-house PEGASUS' - inspired by both the BRIO paper and the Liu et al. (2021) SimCLS paper.

To evaluate the effectiveness of our approach, we use the ROUGE metric, which measures the overlap between the generated summaries and the reference summaries. We decided on this metric, as it is currently the industry standard for evaluation performance. We define success as surpassing the ROUGE-1 score found in the Kornilova and Eidelman (2019) paper, which is 0.4861. We conduct experiments to compare the performance of our approach to previous work in the field, as well as different variations of our approach.

3.1 Data

The Billsum datasource on HuggingFace has both clean legislative data as well as reference summaries for evaluation. The BillSum dataset is a collection of summaries of United States Congressional bills. It includes summaries of bills from both the House of Representatives and the Senate, spanning from the 110th Congress (2007-2008) to the 116th Congress (2019-2020). The dataset is split into three subsets: training, US test, and CA test. Each bill in the dataset contains the original text and a 'golden summary', which contains the summary of the bill.

3.2 Exploratory Data Analysis (EDA)

Performing EDA to compare how different models tokenize the same text can help understand how the models are processing and interpreting the text, and can give insights into the strengths and limitations of each model. Figure 1 shows how two BART and two PEGASUS models tokenize the same text inputs.

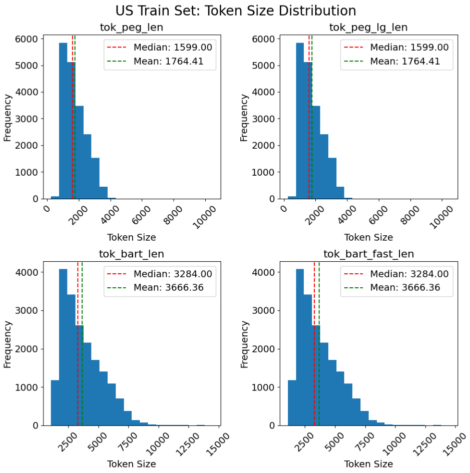


Figure 1: BART uses a standard tokenization process where the input text is split into words and punctuation marks, which are then further processed into subword units using the Byte Pair Encoding (BPE) algorithm. On the other hand, PEGASUS employs a more advanced approach to tokenization called *SentencePiece*. This results in a difference in the tokenization count distribution. Larger tokens can suggest that more detailed information is captured. However, they may make it more difficult for the model to generalize to unseen text. Top right: LSG-Pegasus-4096 tokenizer. Top left: Pegasus base tokenizer. Bottom right: BartTokenizerFast. Bottom left: Bart base tokenizer.

4 Baseline Method

Extractive Summarization is a widely used approach, which involves selecting a subset of sentences from the input text that are deemed most important and informative, while maintaining the overall meaning of the text. BertSummarizer is an implementation of extractive summarization using SBERT models. It uses a pre-trained SBERT model for sentence embedding and cosine similarity to identify important sentences in the input text. This was determined to be our baseline method, as it would act as a competitive comparison to our host of abstractive summarization experiments.

5 Standard Fine-Tuning Methods

5.1 BART

BART (Bidirectional and Auto-Regressive Transformer) is a state-of-the-art language model that has gained popularity in recent years for its ability to perform abstractive summarization. Unlike other summarization techniques that only focus on extracting important sentences or phrases from a text, BART uses a combination of both generative and extractive approaches to produce a summary that captures the main points of the original text in a condensed form. For our experiment, the Facebook-large model was used as it has been pre-trained on a large corpus of text, allowing it to effectively capture the nuances of natural language. Additionally, BartTokenizerFast, an optimized version of the base BART tokenizer which uses SentencePiece, was selected due to its ability to quickly process large texts.

Some limitations are presented by BART:

1) Existing models have not been pre-trained on the Billsum corpus. Like other models, BART's performance is highly dependent on the quality and quantity of training data

2) The maximum token limit is 1024. As shown in the EDA, many input texts have token sizes that far exceed 1024 tokens so we have concerns about information being lost due to truncation.

We begin our experimentation by first fine-tuning the Facebook-large model on 1000 samples from the US train set and using it as our primary model moving forward. Next, we attempt to address the token limit through the following few options.

LSG-BART-Base-4096: This is a version of the base BART model that is modified to handle long sequences (up to 4096 tokens). We attempt to use this model, but due to lack of documentation and community support on Huggingface, we cannot move forward. This will be considered for future experiments.

Sliding Window: We define a function that splits each input text into text chunks and encodes them using the tokenizer. It then groups similar text chunks into clusters based on their similarity scores. For each cluster, the function concatenates the sentences in the corresponding text chunks to create a single text string and passes it into the model for summary generation. The function generates a summary for each cluster and concatenates them together to create a final summary for the entire input text.

Due to a constraint on computational resources, this method was tested on 5 samples and compared against summaries generated with the fine-tuned model using beam search. The sliding window technique yielded a ROUGE-1 score than the extractive baseline, and therefore we chose not to move forward with this method.

The remainder of the experiment was conducted by using the fine-tuned model to produce candidate summaries on the entire US test and CA test sets using beam search, diverse beam search, and top-k sampling.

For the initial fine-tuning loop we use Maximum Likelihood Estimation (MLE) loss which computes the cross-entropy loss between the gold summary and the generated summary:

$$Loss_{mle} = \sum_{i=1}^{N} \left[-\log \left(\frac{\exp(z_{i,y_i})}{\sum_{j=1}^{V} \exp(z_{i,j})} \right) \right]$$

where N is the total number of tokens in the input sequence, V is the size of the vocabulary, z_i is a vector of logits for each token in the sequence, y_i is the gold-standard label for token i. The MLE loss is computed by summing over all tokens in the sequence.

5.2 PEGASUS

PEGASUS is another widely used pre-trained transformer-based language model, but unlike BART it is specifically pre-trained to handle abstractive summarization tasks. For our application, the LSG-4096 Pre-Trained PEGASUS Model was selected for its ability to handle up to 4096 tokens in the input document and its model size enabling it to capture more complex relationships and patterns in the data. With 4096 attention heads 2.5 billion parameters, LSG 4096 PEGASUS model is more than four times larger than the original PEGASUS model and more than six times larger than the BART model. A large majority of the BillSum documents are in excess of 1024 PEGASUS tokens (Max Token Length for traditional BART PEGASUS Models), making this model our only approach that will not have to truncate the input text on any of our documents, which will be critical in capturing all the important details of the gold summaries.

Since LSG-4096 is a general-use model that is pre-trained on a massive corpus of diverse text data, our first step was to fine-tune the model specifically for our BillSum dataset to ensure the model picks up on the patterns unique to legislative documents. To do this, we randomly selected 1500 documents from the training set of 18000 to capture the wide range of document and gold summary lengths. We then took these two samples of documents and fine-tuned the base LSG-4096 model using the same MLE Loss as mentioned in the “BART” section to create two fine-tuned PEGASUS models to be used for our contrastive learning experiment that will be discussed later in the paper.

6 Contrastive Learning with Multiple Candidates

6.1 BRIO (BART)

BRIO As briefly discussed in the background section, the BRIO method has shown powerful performance in addressing exposure bias on both the CNN and DailyMail datasets. With its two-step approach for generating candidate summaries, and then assigning probability mass to these candidate summaries during decoding, it was our most powerful benchmark for leveraging contrastive learning.

For implementation, we will be using BART as our backbone. It has shown strong abstractive summarization performance in previous papers (including BRIO) and will be an interesting point of comparison to our other contrastive learning approach leveraging PEGASUS. Similar to our standard BART fine-tuned method, BRIO will be using the same Facebook-large model for candidate summary generation and fine-tuning. However, the tokenization method is unique to this method, as it is using Stanford CoreNLP. Matching tokenization to other approaches in this paper would have been ideal, but this was the most used and implemented tokenizer for the BRIO method. This means after 1024 tokens all input text is simply truncated, which will become an important point of discussion.

Candidate Generation: For initial candidate summary generation, the BRIO model will generate 16 summaries per text using only diverse beam search and rank-ordered by their ROUGE-1 score. This is in contrast to our In-House PEGASUS method, which will use a variety of beam searches to generate the candidate summaries.

Contrastive Training: Just like our In-house PEGASUS method, the model will then train on both the summary rankings and generated summary text to minimize contrastive and MLE loss as a multi-task fine tuning approach. More details on this can be found in the following 'In-house PEGASUS' section. The training sample consists of 3000 texts from the US BillSum training set. This trained model then can have a dual role - a generation model and an evaluation model. Figure 2 may be helpful to visualize the model being trained.

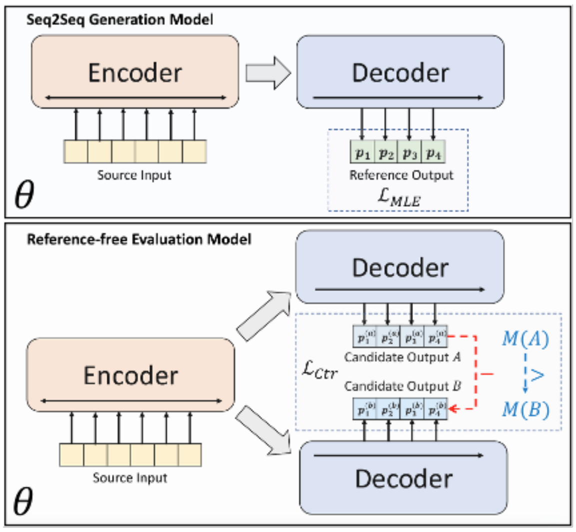


Figure 2: Comparison of *MLE loss* and *contrastive loss* in the BRIO method. MLE assumes a deterministic (one-point) distribution, where the reference summary receives all the probability mass. The BRIO method assumes a *nondeterministic distribution* in which system-generated summaries also receive probability mass according to their quality. The contrastive loss encourages the order of model-predicted probabilities of candidate summaries to be coordinated with their actual quality metric (ROUGE).

6.2 In-house PEGASUS

To address exposure bias in our initial PEGASUS fine-tuned model (that strictly used MLE Loss), we took a two-stage approach similar to Liu et al. (2021/2022) in SimCLS BRIO. This approach first generates a series of candidate summaries with the fine-tuned model, then performs an additional fine-tuning step which introduces a second component to the traditional MLE Loss called Ranking Loss. Ranking Loss adds an additional objective of ranking the generated summary against a set of reference summaries. Since MLE Loss only optimizes for the likelihood of the target summary given the input document, the purpose of this approach is to generate summaries that are informative and diverse compared to the reference summaries. By doing so, it can produce higher quality summaries that capture a broader range of information from the input document and avoid redundancy.

Candidate Generation: The first step in implementing this approach is to generate candidate summaries from our two fine-tuned PEGASUS models. For this approach to be most effective, each of the candidates had to be sufficiently unique from one another. To accomplish this, we generated three summaries for each of our two fine-tuned PEGASUS models on a third unique population of 1500 training samples assuming beam search, diverse beam search, and top K sampling. In order to confirm these summaries were producing a wide range of diversity, we analyzed the distributions of ROUGE-1 ROUGE-1 Min-Max delta on a per-document basis:

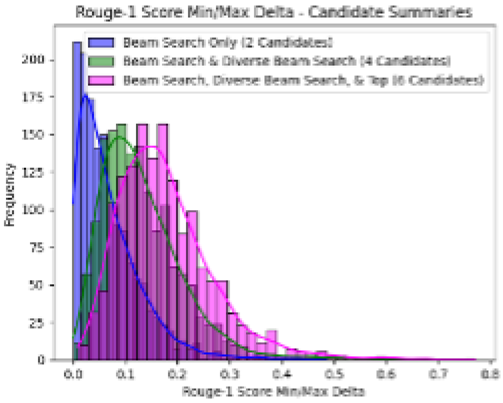


Figure 3: As candidates are added, the ROUGE-1 Min/Max Delta increases, providing more and more diversity in candidate summaries for the model to learn from.

We found that with our 6 candidates generated in this way, we were able to provide a wide-range of ROGUE-1 candidates for the model to learn from with contrastive loss. We decided to stop at a total of 6 candidates based on findings that Liu et al. (2021) had in SimCLS that showed diminishing returns in ROUGE-1 benefit as candidates beyond 6 were added.

Contrastive Training: With the candidates generated, the next step is to go through a 2nd fine-tuning on the LSG 4096 PEGASUS model introducing a new RankingLoss function that is used by Liu et al. (2022) in BRIO to achieve state of the art results on CNN/Daily Mail:

$$Loss_{ctr} = \sum_{i=1}^{num\_cand} \text{RankLoss}(c\_scores, scores)$$

The contrastive loss function takes in the cosine similarity scores between the all summaries (candidate, and generated summary) and the input text itself as scores **c_scores**. It then uses those to compute a **RankLoss** between the scores of the candidates and the model generated summary. Since our multi-candidate training loop will use the traditional MLE loss and the Contrastive Loss added together to train the model and the gold summary is already heavily weighted by the MLE loss, we did not consider the gold summary as part of the Contrastive Loss. For our final loss in this model we added the MLE and Contrastive Losses together and used weights to ensure that the contribution from each was of equivalent weight:

$$Loss_{total} = \text{w1} \times Loss_{ctr} + \text{w2} \times Loss_{mle}$$

7 Results and Discussion

In summary, we were able to meet our goal of improving upon our baseline fine-tuned model for both BART and PEGASUS by implementing contrastive loss. Similar to the success Liu et al. (2021) had improving upon their baseline in SimCLS, our 6-Candidate In-House PEGASUS implementation yielded an improvement of 2% in ROUGE-1. We were also able to make significant improvement over our baseline fine-tuned BART model with the BRIO implementation of BART applied to our dataset, with a 7% improvement in ROUGE-1, scoring the highest of all of our models on the US Test set. Both Multi-Candidate models achieved the goal of exceeding the ROUGE-1 of 0.4861 on the BillSum dataset demonstrated in Kornilova, A., Eidelman, V. (2019).

System	R1	R2	RL
Baseline	0.355	0.166	0.244
PEGASUS	0.470	0.292	0.354
BART	0.432	0.252	0.319
In-House PEGASUS	**0.488**	0.283	0.341
BRIO (BART)	**0.493**	0.311	0.373

Table 1: **US Results:** As expected, the baseline extractive summarization method performed the worst across all metrics with an R1 score of only .355. This is followed by the BART and PEGASUS fine-tuned models with R1 scores of .432 and .470 respectively. The contrastive learning methods were both the highest scoring, with PEGASUS scoring an R1 of .488 and the BRIO (BART) scoring an R1 of .493.

System	R1	R2	RL
Baseline	0.394	0.160	0.221
PEGASUS	0.312	0.154	0.205
BART	0.323	0.152	0.208
In-House PEGASUS	0.318	.146	0.199
BRIO (BART)	0.340	0.170	0.294

Table 2: **CA Results:** The California test set tells a slightly different story. All approaches performed significantly worse compared to the US test set. Consequently, the baseline approach is the surprising summarization front-runner with an R1 of only .394. The performance ranking remains very similar to the US test - barring the BART and PEGASUS fine-tuned models. BART here outperformed PEGASUS with an R1 score of .323 and .312 respectively.

With a more optimized code-base for processing, the BRIO implementation of BART on the BillSum dataset uses more candidates (16 vs. 6), fine-tunes on more training data (1500 vs. 3000), runs for more epochs (4 vs. 20), and implements learning rate decay, so it makes sense that it is able to improve upon its fine-tuned baseline much more and edge out the multi-candidate LSG-4096 PEGASUS model. However, our analysis suggests that the BRIO implementation still could be improved upon quite a bit as it is still heavily handicapped by its max token length of input documents. This allows the LSG-4096 Multi-Candidate approach to rival the results of BRIO with a smaller scale fine-tuning implementation of a similar framework.

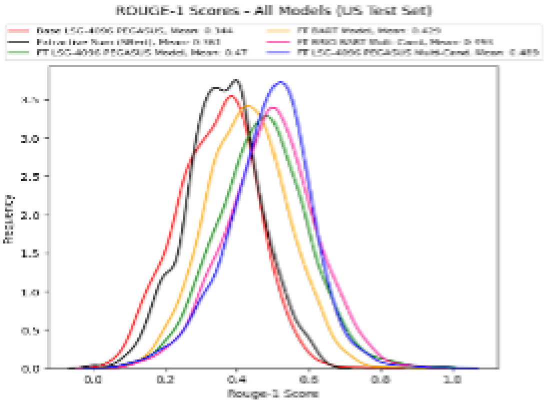


Figure 4: Distribution of ROUGE-1 scores for all of our models on the US Test set. The two models that implement the contrastive loss score the highest (BRIO BART and LSG-4096 PEGASUS), but their two distributions look a lot different. The LSG-4096 PEGASUS model has a higher score at its peak frequency than BRIO BART, but it is unable to achieve many scores in the 0.6-0.8 ROUGE so its distribution falls off quickly after its peak.

7.1 Model Performance vs. Input Text Length

One primary difference between the BRIO BART and PEGASUS models is the input document max token length. BRIO BART is truncating all tokens beyond 1024 which trims out the ends of longer documents, while LSG-4096 PEGASUS is able to tokenize all documents in the BillSum corpus with a max length of 4096. Our initial hypothesis is that BRIO BART would perform best on shorter documents where it was not truncating while LSG-4096 PEGASUS would perform better on the longer documents. To test this, we took each document in the US Test Corpus and chose the model with the best performance out of all 4 fine-tuned models and then plotted their distributions:

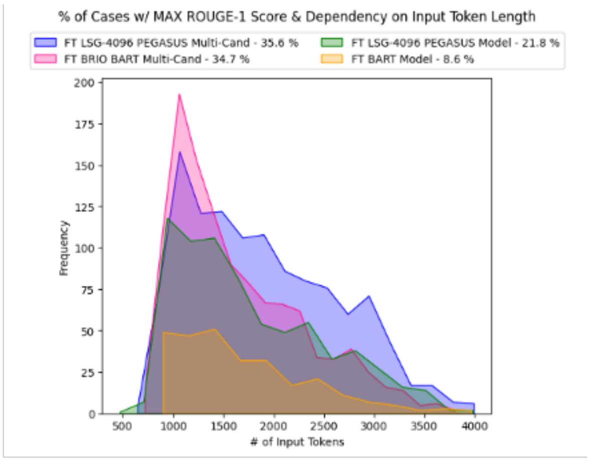


Figure 5: The figure above shows the distribution of of Input Tokens (in PEGASUS Token space) for documents in which each model was the highest performing model. As expected, the BRIO BART model is the highest performer on shorter documents, with the LSG-4096 PEGASUS Multi-Candidate taking over for longer documents. First order fits of ROUGE-1 vs. Input Tokens corroborates this theory on BRIO BART. (See Appendix)

7.2 Model Performance vs. Golden Summary Length

With ROUGE being a metric that measures the recall of the n-grams in the reference summary that are also present in the generated summary, for maximum model performance it is imperative that the generated summary is of similar length to the reference summary. With that being said, it's important to study the distribution of golden summary lengths vs. generated summary lengths for each of the models to develop some intuition to help explain model results. The figures below show that the LSG-4096 Multi-Candidate model is able to replicate a similar distribution of Summary lengths to the Gold Summaries which allows it to maintain performance even when the golden summaries are longer:

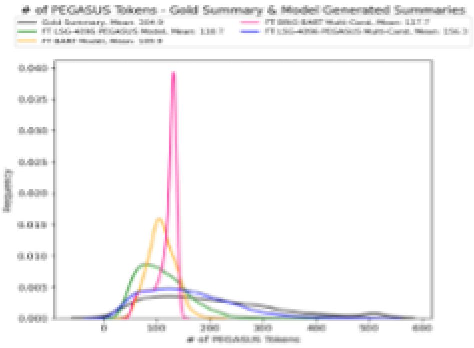


Figure 6: The figure above shows the distribution of tokens for the Gold Summaries vs. each of the models. The BART models were both fairly constrained between 50-150 tokens. On the other hand, the multi-candidate LSG-4096 model was able to replicate the distribution of summary lengths that existed in the Gold Summary population.

7.3 Notable differences in text between US Test and CA Test Sets

Across all models, ROUGE results from the US test set are higher than the CA test set. While California bills follow a similar format and structure to that of US bills, they have a different structure for

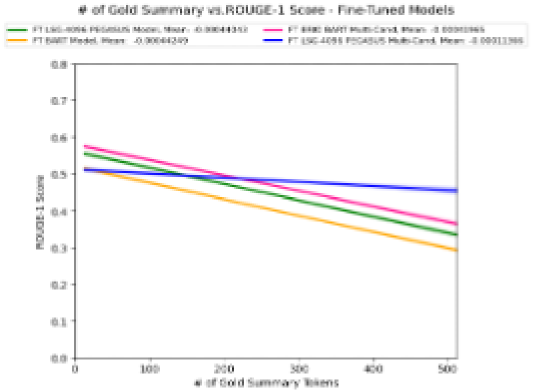


Figure 7: The figure above shows that matching the Gold Summary distribution allowed the Multi-Candidate LSG-4096 model to maintain performance regardless of golden summary token length, while the other models showed a performance hit for longer summaries.

some golden summaries. For example, the existing law is described *in addition* to the explanation of the change, as also pointed out in section C.3 of the Kornilova, A., Eidelman, V. (2019) paper.

Overall, the use of NLP techniques for summarizing US and CA bills has shown promising results in terms of generating accurate and concise summaries. However, two caveats should be kept in mind. Firstly, the standardized structure and length of the summaries, as set by the Congressional Research Service, may pose a challenge for summarizing more complex inputs that do not fit this format. Secondly, the way ROUGE is calculated may not always accurately reflect the quality of the generated summary, particularly when short inputs yield short summaries that still hit all the relevant points.

8 Conclusion

In this work, we presented a contrastive learning paradigm to abstractive summarization of the Billsum dataset. Our methods achieved ROUGE score improvement over our baseline methods as well as previous summarization methods on Billsum mentioned in the Kornilova, A. Eidelman, V. (2019) paper. Although, there is still room for future work.

First, it would be useful to explore ways of addressing the CA-US test data discrepancy through a larger training sample or different training approaches. Second, other hyperparameter variations such as loss function weights, number of candidate summaries, BART/PEGASUS backbone, and beam search/size can be further explored for optimal performance. Finally, the exploration of other metrics beyond ROUGE could yield interesting results, as the CA dataset and examples in the Appendix demonstrate how focusing solely on golden summaries for performance may not tell the full story of model performance.

9 References

Kornilova, A., Eidelman, V. (2019). BillSum: A Corpus for Automatic Summarization of US Legislation. ArXiv (Cornell University). https://doi.org/10.18653/v1/d19-5406

Liang, X., Wu, L., Li, J., Wang, Y., Meng, Q., Qin, T., Chen, W., Zhang, M., Liu, T. (2021). R-Drop: Regularized Dropout for Neural Networks. ArXiv (Cornell University). https://doi.org/10.48550/arxiv.2106.14448

Liu, Y., Dou, Z., Liu, P. (2021). RefSum: Refactoring Neural Summarization. North American Chapter of the Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.naacl-main.113

Liu, Y., Liu, P., Radev, D. R., Neubig, G. (2022). BRIO: Bringing Order to Abstractive Summarization. Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). https://doi.org/10.18653/v1/2022.acl-long.207

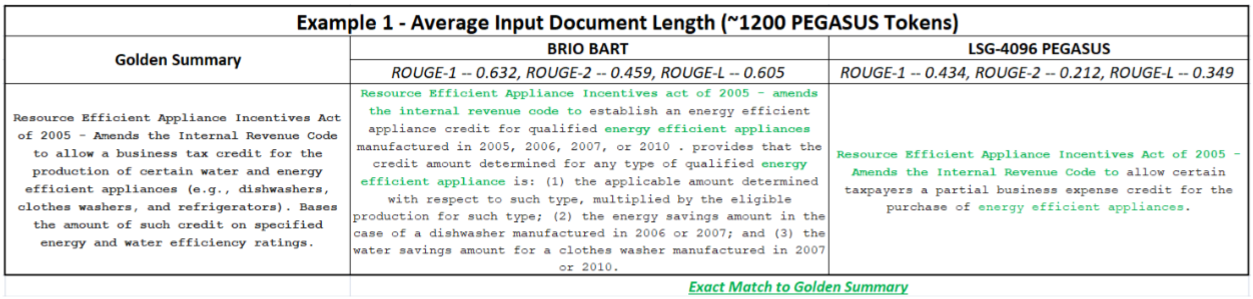
Liu, Y., Liu, P. (2021). SimCLS: A Simple Framework for Contrastive Learning of Abstractive Summarization. Meeting of the Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-short.135

Ravaut, M., Joty, S., Chen, N. (2022). SummaReranker: A Multi-Task Mixture-of-Experts Re-ranking Framework for Abstractive Summarization. Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). https://doi.org/10.18653/v1/2022.acl-long.309

A Appendix

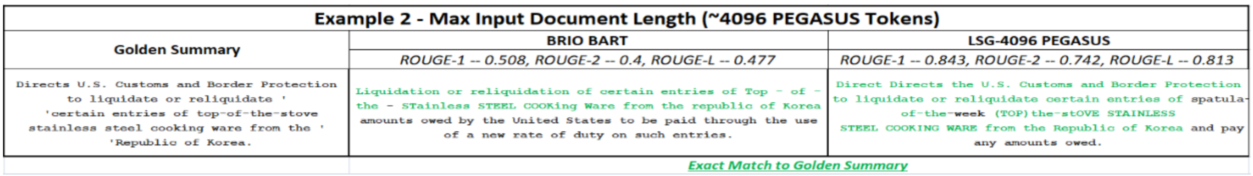
A.1 Summary Comparison - BRIO BART vs. LSG 4096 PEGASUS

See below for three examples of how well BRIO BART and LSG-4096 PEGASUS match the golden summary. As can be seen by these examples, ROUGE scores don't always correlate directly with the "best" summary, just the best exact match to a golden reference. Sometimes paraphrasing the same sentiment as the Golden Summary in different words can be punished by the metric. This is why in the future work section we proposed looking at more evaluation metrics beyond ROUGE.



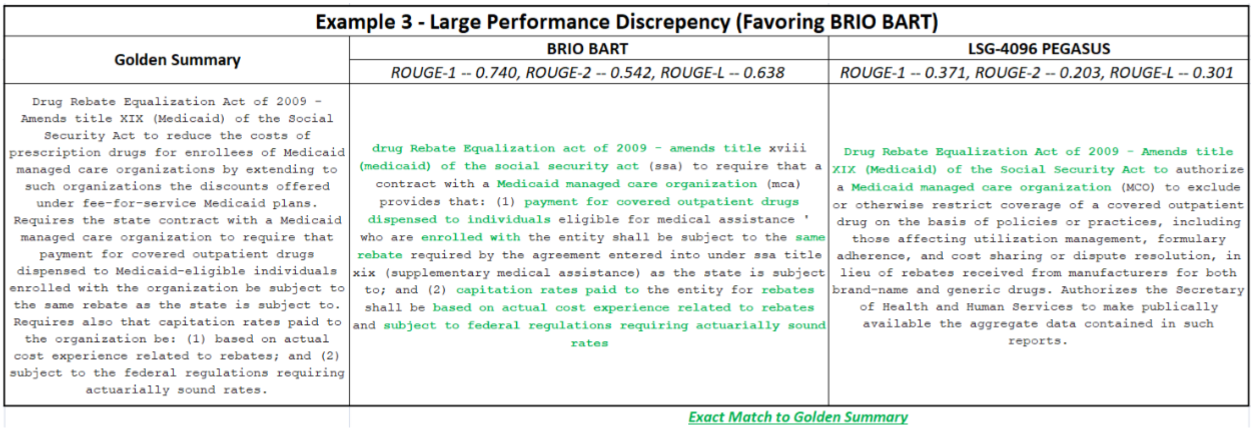
Example 1 - Average Input Document Length (~1200 PEGASUS Tokens)		
Golden Summary	BRIO BART	LSG-4096 PEGASUS
	ROUGE-1 -- 0.632, ROUGE-2 -- 0.459, ROUGE-L -- 0.605	*ROUGE-1 -- 0.434, ROUGE-2 -- 0.212, ROUGE-L -- 0.349*
Resource Efficient Appliance Incentives Act of 2005 - Amends the Internal Revenue Code to allow a business tax credit for the production of certain water and energy efficient appliances (e.g., dishwashers, clothes washers, and refrigerators). Bases the amount of such credit on specified energy and water efficiency ratings.	Resource Efficient Appliance Incentives act of 2005 - amends the internal revenue code to establish an energy efficient appliance credit for qualified energy efficient appliances manufactured in 2005, 2006, 2007, or 2010 . provides that the credit amount determined for any type of qualified energy efficient appliance is: (1) the applicable amount determined with respect to such type, multiplied by the eligible production for such type; (2) the energy savings amount in the case of a dishwasher manufactured in 2006 or 2007; and (3) the water savings amount for a clothes washer manufactured in 2007 or 2010.	Resource Efficient Appliance Incentives Act of 2005 - Amends the Internal Revenue Code to allow certain taxpayers a partial business expense credit for the purchase of energy efficient appliances.
	Exact Match to Golden Summary	

Figure 8: Average Input Document Length



Example 2 - Max Input Document Length (~4096 PEGASUS Tokens)		
Golden Summary	BRIO BART	LSG-4096 PEGASUS
	ROUGE-1 -- 0.508, ROUGE-2 -- 0.4, ROUGE-L -- 0.477	*ROUGE-1 -- 0.843, ROUGE-2 -- 0.742, ROUGE-L -- 0.813*
Directs U.S. Customs and Border Protection to liquidate or reliquidate ' 'certain entries of top-of-the-stove stainless steel cooking ware from the ' 'Republic of Korea.	Liquidation or reliquidation of certain entries of Top - of - the - STainless STEEL COOKing Ware from the republic of Korea amounts owed by the United States to be paid through the use of a new rate of duty on such entries.	Direct Directs the U.S. Customs and Border Protection to liquidate or reliquidate certain entries of spatula-of-the-week (TOP) the-stOVE STAINLESS STEEL COOKING WARE from the Republic of Korea and pay any amounts owed.
	Exact Match to Golden Summary	

Figure 9: Long Input Document Length



Example 3 - Large Performance Discrepency (Favoring BRIO BART)		
Golden Summary	BRIO BART	LSG-4096 PEGASUS
	ROUGE-1 -- 0.740, ROUGE-2 -- 0.542, ROUGE-L -- 0.638	*ROUGE-1 -- 0.371, ROUGE-2 -- 0.203, ROUGE-L -- 0.301*
Drug Rebate Equalization Act of 2009 - Amends title XIX (Medicaid) of the Social Security Act to reduce the costs of prescription drugs for enrollees of Medicaid managed care organizations by extending to such organizations the discounts offered under fee-for-service Medicaid plans. Requires the state contract with a Medicaid managed care organization to require that payment for covered outpatient drugs dispensed to Medicaid-eligible individuals enrolled with the organization be subject to the same rebate as the state is subject to. Requires also that capitation rates paid to the organization be: (1) based on actual cost experience related to rebates; and (2) subject to the federal regulations requiring actuarially sound rates.	drug Rebate Equalization act of 2009 - amends title xviii (medicaid) of the social security act (ssa) to require that a contract with a Medicaid managed care organization (mca) provides that: (1) payment for covered outpatient drugs dispensed to individuals eligible for medical assistance ' who are enrolled with the entity shall be subject to the same rebate required by the agreement entered into under ssa title xix (supplementary medical assistance) as the state is subject to; and (2) capitation rates paid to the entity for rebates shall be based on actual cost experience related to rebates and subject to federal regulations requiring actuarially sound rates	Drug Rebate Equalization Act of 2009 - Amends title XIX (Medicaid) of the Social Security Act to authorize a Medicaid managed care organization (MCO) to exclude or otherwise restrict coverage of a covered outpatient drug on the basis of policies or practices, including those affecting utilization management, formulary adherence, and cost sharing or dispute resolution, in lieu of rebates received from manufacturers for both brand-name and generic drugs. Authorizes the Secretary of Health and Human Services to make publically available the aggregate data contained in such reports.
	Exact Match to Golden Summary	

Figure 10: Large Performance Discrepency

A.2 Additional Figures

Candidate Summaries & Fine-Tuned Models - Hyperparameter Selection			
HP	*Beam Search*	*Diverse Beam*	*Top K*
`max_length`	**512**	**512**	**512**
`num_beams`	**8**	**8**	**8**
`length_penalty`	**2**	**2**	N/A
`no_repeat_ngram_size`	**2**	**2**	N/A
`early_stopping`	**TRUE**	**TRUE**	N/A
`diversity_penalty`	N/A	**1**	N/A
`num_beam_groups`	N/A	**4**	N/A
`do_sample`	N/A	N/A	**TRUE**
`top_k`	N/A	N/A	**50**
`top_p`	N/A	N/A	**92**

Figure 11: Hyperparameters used to generate candidate summaries and also generate final results for the fine-tuned models

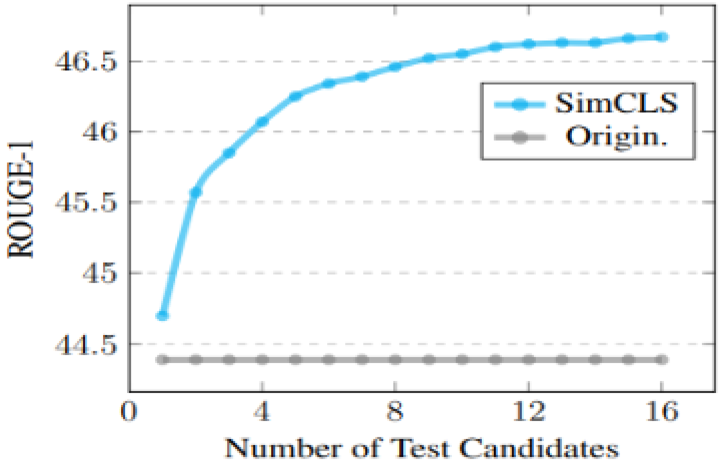
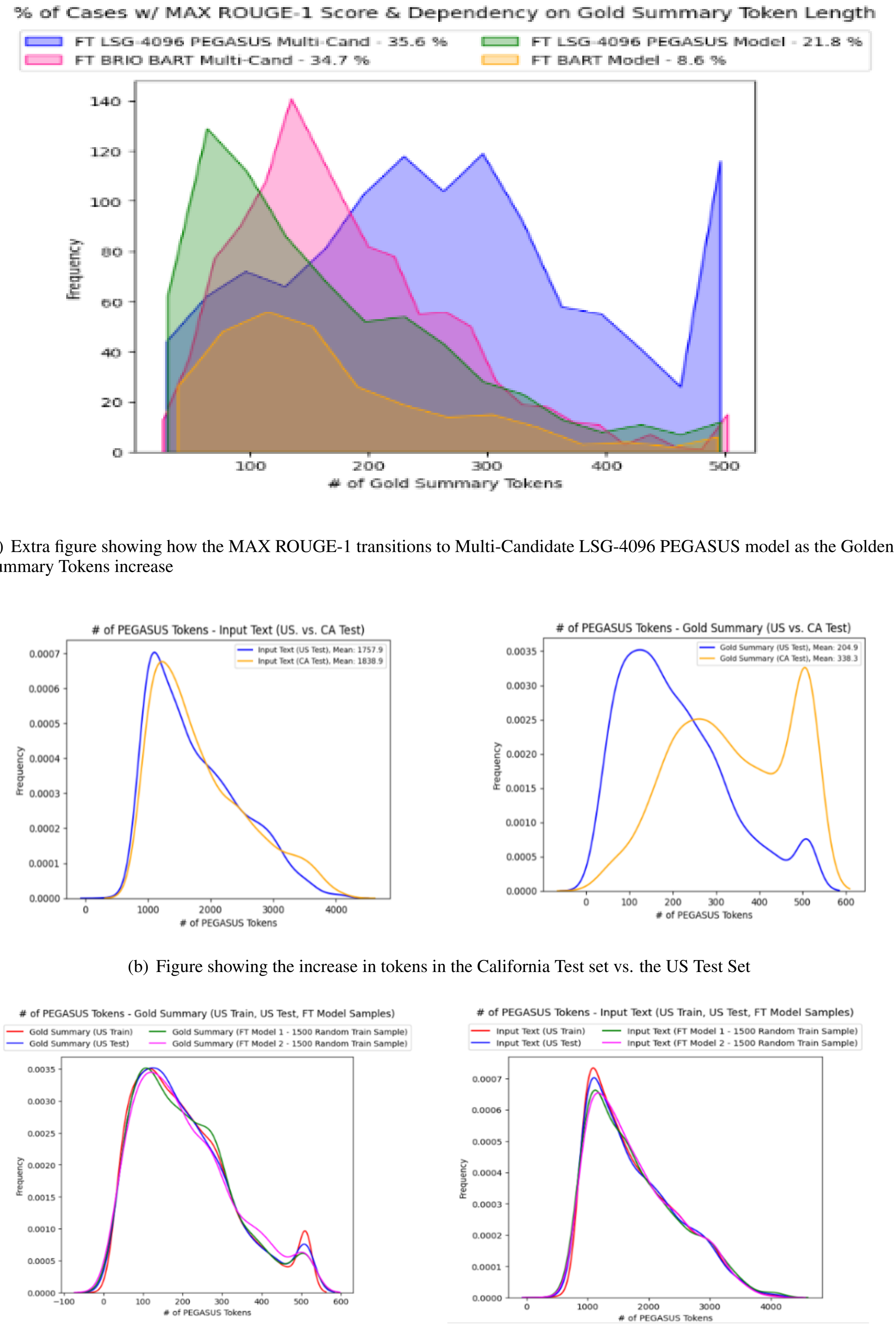


Figure 2: Test performance with different numbers of candidate summaries on CNNDM. **Origin** denotes the original performance of the baseline model.

Figure 12: Hyperparameters used to generate candidate summaries and also generate final results for the fine-tuned models



(a) Extra figure showing how the MAX ROUGE-1 transitions to Multi-Candidate LSG-4096 PEGASUS model as the Golden Summary Tokens increase

(b) Figure showing the increase in tokens in the California Test set vs. the US Test Set

(c) Two unique samples used for fine-tuning align w/ the distributions of PEGASUS tokens for Input Text Summaries

Figure 13: Additional Figures