Analyzing BART Generalizability to Resume Summarization

CS 333 Natural Language Processing Final Project

Karen Xiao

1. INTRODUCTION

1.1 Motivation

In today's ultra-competitive job market, applicants have become all-too-familiar with recruitment processes that often reject seemingly qualified applicants in a matter of hours. In these cases, recruiters streamline the recruitment process by using resume parsing tools derived from artificial intelligence and natural language processing. These tools aim to extract key information from application documents using pattern matching and other language analysis to easily parse through potential candidates. Though this automated process often feels dehumanizing, the time saved by recruiters is often crucial to making sure that the application review process happens in a timely manner. Therefore, it is critical that these summarization tools accurately capture necessary information from candidates' resumes.

There are many NLP models that accomplish this task of text summarization – one of the most well-known being the Bidirectional and Auto-Regressive Transformer (BART) model (Lewis et al., 2020). To test its summarization capabilities, this model was originally trained on two news datasets: CNN/DailyMail and XSum. While BART performs well on these longer, narrative-style articles, this work examines how BART performs on resumes, which resemble shorter collections and differ substantially from news articles. Therefore, this work aims to answer the question: How well does the BART model perform on resume summarization? Specifically, if we use BART to summarize a resume, how does it compare to a human-created summary of the resume? Does the BART model identify the same key information as the human-generated summary?

1.2 Related Work

Complex models like BART perform well across a variety of tasks. However, their training often relies on large-scale textual artifacts such as books, academic documents, and web articles. As a result, these models often require additional task-specific data to adequately capture the nuances related to the given task. However, the idea of fine-tuning BART using domain-specific data is not new, by any means.

Because BART is a well-known NLP model, there is a lot of previous work both regarding the model itself (Lewis et al., 2020) and also applying it to various domains such as movie scripts (Upadhyay et al., 2025), Amazon reviews (Yadav et al., 2023), and many others. Many such works find that BART generally performs well, but benefits slightly from fine-tuning on more niche domains such as medical dialogues (Sahu et al., 2025).

In addition to these more general applications of BART, there is also some prior work that specifically focuses on the task of resume summarization by analyzing the quality of generated resume summaries using various different methods. One study in particular finds that the BART-Large model outperforms models such as BART-Base, T5, and Pegasus (Mercan et al., 2023).

To evaluate the performance of text-summarization tasks, many prior works make use of the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metrics.

The ROUGE metrics automatically determine the quality of a summary by comparing it to other ideal summaries created by humans. Essentially, ROUGE counts the number of overlapping units between computer and human-generated summaries to compute syntactic similarity between the two summaries (Lin, 2004).

2. DATA

In this study, we use the burberg92/resume_summary dataset from Hugging Face. This dataset contains 100 resumes and corresponding summaries. For the sake of the study, we will use the summaries provided by the dataset as gold-label summaries. A sample of 20 resumes from the dataset has been taken to test the performance of our model before and after fine-tuning. An example of a resume and corresponding summary is depicted in Figure 1 and 2 respectively.

Resume
Maria Bell | Interior Designer Core Competencies: Space planning, color theory, and design concept development Excellent client management and collaboration skills Design Software: AutoCAD, SketchUp, 3ds Max, and Adobe Creative Suite Work History: Design Assistant, STU Interiors (2013-2015) Supported designers in creating design concepts and presentations Assisted with materials selection and project coordination Interior Designer, UVW Design Studio (2015-2023) Developed and implemented design solutions for residential and commercial clients Collaborated with clients and vendors to ensure project success Education: - Bachelor's degree in Interior Design, XYZ University (2009-2013)

Figure 1: Original Resume from Dataset

Gold-label Summary
Innovative Interior Designer with expertise in space planning, color theory, and design concept development. Excellent client management and collaboration skills, with proficiency in AutoCAD, SketchUp, 3ds Max, and Adobe Creative Suite. Holds a Bachelor's degree in Interior Design from XYZ University.

Figure 2: Gold-label Summary from Dataset

3. MODEL

This study uses the Bidirectional and Auto-Regressive Transformer (BART) model originally presented in 2020 by Lewis et al. (Lewis et al., 2020). BART is a denoising autoencoder built with a sequence-to-sequence model that is applicable to a variety of tasks such as text summarization, translation, and question answering. Specifically, BART uses a standard transformer-based architecture which can be seen as a generalization of many other pretraining schemes such as BERT and GPT.

The pretraining for this model has two stages. In the first stage, text is corrupted and then in the second stage, a model is learned to reconstruct the original text. In their 2020 paper, the authors discuss a variety of pretraining objectives and approaches such as permuted language models, masked language models, and multitask masked language models. From this, the authors conclude that the effectiveness of pre-training methods is highly dependent on the task and that BART achieves the most consistently strong performance over other models. Specifically within the context of summarization, BART outperforms other previous work that only leverages BERT (Lewis et al., 2020).

For the purpose of this study, we use the facebook/bart-large-cnn model available on Hugging Face. This particular version of the BART model has been fine-tuned on CNN

Daily Mail, a large collection of article-summary pairs.

4. METRIC

To analyze the generalizability of the BART model on resume summarization, we will run this model on each of the resumes provided in our dataset and calculate evaluation metrics to compare the similarity between the BART-generated and the gold-label summary provided by the dataset. We will use two different types of evaluation metrics to detect both syntactic and semantic similarity between BART-generated and gold-label summaries.

4.1 Syntactic Similarity

As mentioned in the related work, we will use the ROUGE-1, ROUGE-2, and ROUGE-L measures to compute syntactic similarity between generated and reference summaries. For each ROUGE measure, we will focus on the balanced *F* measure to account for both precision and recall. The three ROUGE measures do the following:

- *ROUGE-1:* measures the overlap of individual words (unigrams) between the generated and reference summaries
- *ROUGE-2:* measures the overlap of bigrams between the generated and reference summaries
- *ROUGE-L:* measures the longest common subsequence between generated and reference summaries

Each of these measures accomplishes a different purpose in terms of evaluation. For example, since ROUGE-L looks for the longest common subsequence, it will capture sentence-level structure more than ROUGE-1 or ROUGE-2. However, ROUGE-1 and ROUGE-2 do a better job of checking for essential keywords and short phrases.

To implement the ROUGE metrics in this study, we will use the public Python rouge library available on GitHub.

4.2 Semantic Similarity

Though the ROUGE measure captures syntactic similarities between generated and gold-label summaries, we are also interested in capturing semantic similarity to provide a more robust result. To do this, we will first create sentence embeddings using the sentence-transformers/all-MiniLM-L6-v2 model on Hugging Face. Using the sentence embeddings, we will calculate semantic similarity between the two embeddings using cosine similarity.

4.3 Evaluation Criteria

In addition to purely evaluating the performance of the BART model on resume summarization, this study also performs fine-tuning, and compares the performance of the fine-tuned BART model to the original BART model using the same evaluation metrics defined above. To measure success, we aim to see better performance, indicated by higher similarity scores, across all metrics in the fine-tuned model.

5. RESULTS

After trying out different settings for max_length and min_length for summary generation based on the number of tokens in the input text and summary text, we settled on max_length=80 and min_length=30. These values consistently resulted in generated summaries with lengths comparable to the gold-label summaries in the dataset.

5.1 Baseline BART Evaluation

Table 1 presents the evaluation metrics calculated after running the BART model on our test set of 20 resumes.

ROUGE 1	ROUGE 2	ROUGE L	Cosine
0.885	0.837	0.885	0.928

Table 1: BART evaluation metrics before fine-tuning

In these results, we see that before fine-tuning, the BART model already performs quite well on the resume dataset. With all evaluation metrics scoring above 80%, we conclude that the BART-generated summaries are similar to the gold-label summaries. Furthermore, we see that the cosine similarity metric of 0.922 indicates that the two summaries are very similar semantically. This result makes sense because many resume terms extracted by BART are also present in the gold-label summary.

As an example of the task, Figure 3 presents the BART-generated summary, before fine-tuning, for the resume in Figure 1 earlier.

BART-generated summary
Creative Interior Designer with expertise in space planning, color theory, and design concept development. Strong client management and collaboration skills, with experience in designing visual assets for various clients. Holds a Bachelor's degree in Interior Design from XYZ University.

Figure 3: BART-generated summary for resume in Figure 1 without fine-tuning

Here, we can see that the BART-generated summary does a good job of summarizing the candidate's core competencies (i.e., space planning, color theory, design concept development), but falls slightly short in its ability to summarize the candidates key technical skills (i.e., AutoCAD, SketchUp, 3ds Max, etc). While the gold-label summary (Figure 2) chooses to highlight these software skills, the BART-generated resume tries to produce a slightly more abstractive summary by adding the phrase "experience in designing visual assets for various clients", which is not present in the original resume.

In addition to examining the overall performance of the BART model, we also explored how performance differs for resumes of different lengths. Figure 4 depicts a bar-chart of the evaluation metrics split by resume length (number of characters). Note that the resume dataset did not have any resumes between 300-499 characters long, so the x-axis ticks in the graph are 100, 200, 500, 600, 700, and overall (which captures resumes of all lengths).

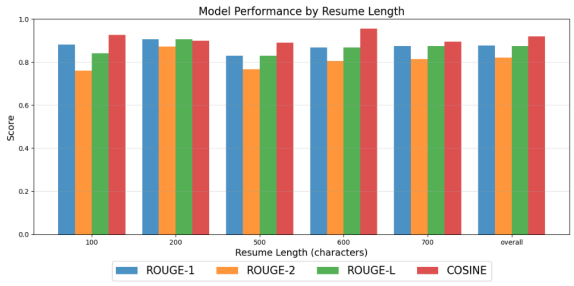

Figure 4: Model Performance by Resume Length

In this figure, an interesting trend emerges. First, we see that performance is relatively stable across resume lengths. However, resumes that have a length between 200-299 characters perform slightly better than resumes that are either shorter or longer. One explanation for this is that very short resumes lack context necessary to create a good summary, while very long resumes may introduce redundant information.

Overall, the BART model, as is, generalizes well to the resume dataset. In response to our original research questions, the BART-generated summary is quite similar to the gold-label summary, but the summaries occasionally identify different key

information. Though the BART model already performs well on the resume dataset, this study also explores the impact of fine-tuning on the model.

5.2 Fine-tuning

Fine-tuning of the model was done on the NCSA Delta cloud computing system with the following resources:

- *Name of account:* bftp-delta-gpu
- *Partition:* gpuA40x4
- *CPU:* 8
- *RAM:* 48GB
- 1 GPU

The training function used for fine-tuning iterates over each batch in our training data and does the following:

(1) Move input ids, attention mask, and labels to the device
(2) Forward pass: call the BART model on each item in the batch
(3) Backpropagation: zero out the gradients in the optimizer, call backward() on the loss, call step() on the optimizer
(4) Add the batch loss to the total training loss for the epoch

The training loop trains the BART model on the training data for 4 epochs and evaluates its performance on the test data after each epoch using the same evaluation metrics defined above. To create training and test data, we did an 80-20 split of the 100 resumes in the dataset.

Figure 5 shows the training loss over the 4 epochs and Figure 6 shows the performance over the 4 epochs. In these plots, we can see that the training loss consistently decreases, as expected. Though the performance seemingly peaks at Epoch 4, the fluctuation in performance up to that point demonstrates that the improvement seen at Epoch 4 is likely not true improvement, but rather some sort of noise due to the fact that the test dataset is so small. We conclude that the model begins overfitting at Epoch 2 because this is the first instance where we see a clear peak and subsequent decrease in performance. Therefore, we will evaluate our fine-tuned BART model after 2 epochs of training.

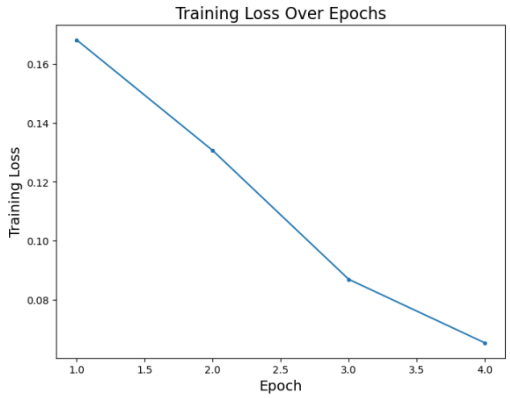

Figure 5: Training Loss Over Epochs

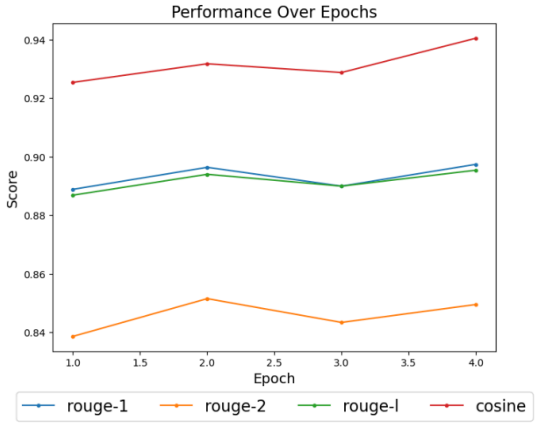

Figure 6: Performance Over Epochs

5.3 Fine-tuned BART Evaluation

The evaluation metrics calculated after running the fine-tuned BART model on our test set is presented in Table 2.

ROUGE 1	ROUGE 2	ROUGE L	Cosine
0.896	0.852	0.894	0.932

Table 2: BART evaluation metrics after fine-tuning

Here, we can see that after 2 rounds of training, the fine-tuned BART model outperforms the baseline BART model in each of the 4 metrics. The biggest improvement occurs in the ROUGE-2 metric which originally scored 0.837 in the baseline BART model.

To illustrate an example, Figure 7 presents the BART-generated summary, after fine-tuning, for the resume presented in Figure 1.

BART-generated summary
Detail-oriented Interior Designer with expertise in space planning, color theory, and design concept development. Strong client management and collaboration skills, with experience in AutoCAD, SketchUp, 3ds Max, and Adobe Creative Suite. Holds a Bachelor's degree in Interior Design from XYZ University.

Figure 3: BART-generated summary for resume in Figure 1 after fine-tuning

Now, we see that the BART-generated summary has successfully identified the technical skills from the original resume. As a result, the new BART-generated summary after fine-tuning more closely resembles the gold-label summary presented in Figure 2.

Therefore, because the fine-tuned BART model scores higher in every evaluation metric compared to the baseline BART model, we conclude that the fine-tuned BART model performs better on the resume summarization task than the baseline BART model.

6. CONCLUSION

In this study, we explored the generalizability of the bart-large-cnn model on a resume summarization task. Because BART was originally trained on a large corpus of news articles, this study focused on exploring how BART performs on resumes, which represent a different structure than the narrative-style articles that BART was trained on. We evaluated the performance of the model on 4 metrics: ROUGE-1, ROUGE-2, ROUGE-L, and cosine similarity.

In regards to our research question, we found that the BART model without any fine-tuning already generalizes well to the resume dataset with scores over 80% for all metrics. However, occasionally, the baseline BART model fails to identify the same key information presented in the gold-label summary.

After 2 rounds of training, we found that the fine-tuned version of the BART model slightly outperformed the baseline model in all metrics. Furthermore, a qualitative review of the generated summaries showed that the fine-tuned model also did a better job of extracting the same key information as the gold-label summaries.

Overall, this study concludes that the BART model generalizes well to the resume summarization task, but performance can be improved through fine-tuning.

6.1 Limitations

The validity of the results presented in this study is greatly impacted by the quality of the dataset used. Though the resume dataset contained both resumes and summaries, there is no documentation available that specifies how the summaries were created. Due to the lack of more robust data, and the timing and scope of this project, we assumed that the summaries in the dataset were created by a human to serve as a human-generated baseline for the BART-generated summaries. However, if the summaries were already generated by BART, or a similar model, then the comparison between the BART-generated summaries and the gold-label summaries would be trivial.

Furthermore, these results are also limited by the amount of data used since our dataset only contained 100 resumes. Though we did see consistently significant improvements to the model performance after fine-tuning, the small size of the dataset likely limits the training capability because we only have 80 resumes to work with.

6.2 Future Work

Future work on this project will be sure to improve and validate the results on a more robust dataset. This improved dataset will be sure to include more resumes, and will also guarantee that the resumes are human-generated. If time allows, part of the future work could even be dedicated to the creation of such a dataset as no such dataset currently exists for this task.

REFERENCES

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., ... & Zettlemoyer, L. (2020, July). BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 7871-7880).

Sahu, N. K., Yadav, M., Chaturvedi, M., Gupta, S., & Lone, H. R. (2025, January). Leveraging language models for summarizing mental state examinations: A comprehensive evaluation and dataset release. In *Proceedings of the 31st International Conference on Computational Linguistics* (pp. 2658-2682).

Upadhyay, A., Bhavsar, N., Bhatnagar, A., Singh, M., & Motlicek, P. (2022, October). Automatic summarization for creative writing: BART based pipeline method for generating summary of movie scripts. In *Proceedings of The Workshop on Automatic Summarization for Creative Writing* (pp. 44-50).

Yadav, H., Patel, N., & Jani, D. (2023). Fine-tuning BART for abstractive reviews summarization. In *Computational intelligence: Select proceedings of InCITe 2022* (pp. 375-385). Singapore: Springer Nature Singapore.

Mercan, Ö. B., Cavsak, S. N., Deliahmetoglu, A., & Tanberk, S. (2023, October). Abstractive text summarization for resumes with cutting edge NLP transformers and LSTM. In *2023 Innovations in Intelligent Systems and Applications Conference (ASYU)* (pp. 1-6). IEEE.

Lin, C. Y. (2004, July). Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out* (pp. 74-81).