Automatic Summarization of City of Miami Commission Meetings

Abdulhadi Sahin
Florida International University
11200 SW 8^{th} St, Miami, FL 33199
asahi004@fiu.edu

Abstract

Recent advances in Natural Language Processing (NLP) and machine learning have opened up exciting new possibilities for using textual data in research. However, summarizing text with specialized vocabulary and complex sentence structures, such as those found in legal text and government bills, can still be challenging. In this project, I explored the City of Miami Commission meetings to extract discussed topics and provide summaries using NLP models. I utilized three widely used pre-trained models, including the Bidirectional auto-regressive transformer (BART). To improve the accuracy of the models, I fine-tuned them with the extracted data. The results were promising, with approximately 40% text similarity between the model-generated and human-made summaries. This study highlights the potential of NLP and machine learning in uncovering valuable information from textual data.

1 Introduction

Scholars of political science have long been explaining and predicting political behavior through quantified approaches. Traditionally, teams of researchers would go through thousands of documents and turn abstract concepts in those texts into measurable observations. Other scholars would then use these metrics and observations in their work and further research in their fields. However, since these efforts took considerable amount of time and energy, in-depth studies of political phenomena were often constrained to the areas where data were more available and quantifiable or where the search interests were more concentrated, such as topics pertaining central governments. With the advancements in textual analysis techniques and computational technology, it is now easier to extract information from more diverse resources. Over the last two decades, we see scholars have been answering substantial political science questions by examining news coverage of the economy, congressional bills, election platforms and campaigns, and twitter feeds (Barberá et al., 2021; De Boef and Kellstedt, 2004; Hillard et al., 2008; Laver et al., 2003; Eshbaugh-Soha, 2010; Akkaya, 2014). There is a growing interest in utilizing NLP and machine learning in these research areas.

I will build on this research and extract more granular data at the local government level. Specifically, I will examine the public meetings of the City of Miami Commission. The minutes of these meetings are divided into twelve broad categories, such as Planning and Zoning items and Public Hearings. Each section includes approved or deferred resolutions and ordinances, along with a one-paragraph definition of each.

In this project, my goal is to summarize the content of each minute into concise and quantifiable data that can be used to answer policy-related questions and evaluate policies at the local government level. The summarization task involves extracting information on the sections and text of resolutions reported in each section. I am primarily interested in the topics discussed in these meetings, and therefore, I will extract summaries of these resolutions using three pre-trained models: Bidirectional Auto-regressive Transformer (BART), Text-To-Text Transfer Transformer (T5), and Pre-training with Extracted Gap-sentences for Abstractive Summarization Sequence-to-sequence model (PEGASUS). Furthermore, I will fine-tune the BART and T5 models with my data and compare their performances.

It is worth noting that federal and local government documents have unique structures compared to regular text. Therefore, models trained on regular text or conversational data may have difficulty summarizing their structure. This paper attempts to demonstrate how cutting-edge pre-trained models perform with such data.

The paper is organized as follows: Section 2 provides an overview of the literature on NLP summarization tasks. Section 3 describes the data and outlines the methods employed in this paper. Section 4 presents the experimental setup and discusses the results, and Section 5 concludes the paper.

2 Related Work

Summarization of textual information in NLP is the process of creating a concise version of a long text while preserving its overall meaning. There are three main summarization techniques; extractive, abstractive and hybrid (Koh et al., 2022). Extractive technique involves selecting important words and sentences from the text and combining them as an output. Abstractive summarization creates an output that gives the essential meaning of the text without copying the original text. The hybrid method combines the two. Researchers generally use extractive summarization to determine important and distinct sections in the text and then use abstractive technique to summarize each section. These are then combined into a summary of the complete text. Hybrid technique is especially relevant for tasks that involve long documents. Documents with more than 700 words are generally difficult to summarize and dividing the text into smaller sections generally gives better performance (Cohan et al., 2018; Gidiotis and Tsoumakas, 2020).

The current research mostly employs transformer architecture for abstrative and extractive summarization tasks, as seen in studies such as Koh et al. (2022); Raffel et al. (2020). If the tasks involves locating distinct segments in the text, researchers may add local attention-based components to capture the sectional differences. Additionally, some studies have utilized hierarchical bidirectional LSTM and convolutional neural networks with encoders for extractive summarization, such as those by Manakul and Gales (2021); Pilault et al. (2020). Furthermore, basic methods such as regular expressions can also be used to locate distinct sections depending on the structure of text (Meystre and Haug, 2005; Shi et al., 2021).

In the following paragraphs, I will discuss a few works that have implemented NLP-based summarization models on long and content-rich data. Jain et al. (2021a) have implemented several summarization algorithms on publicly available legal documents, known as BillSum (Kornilova and Eidelman, 2019). Two of the models they used, TextRank and LexRank, can be considered unsupervised machine learning models. The study found that, in general, graph-based summarization models perform better than other models.

Similarly, Jain et al. (2021b) proposed an extractive summarization approach and implemented it on the BillSum dataset. In the training phase, they utilized Legal-BERT embeddings (Chalkidis et al., 2020) and Anonymous Walk Embeddings (AWE) (Ivanov and Burnaev, 2018) to create inputs. Then, a multilayer perceptron was used to determine the relevance of each sentence for the summary.

Finally, Salaün et al. (2022) implemented a BART-based model called BARTHez, which is a skilled pre-trained French sequence-to-sequence model. The model was trained on a dataset extracted from the JusticeBot database and CanLII, which are two web resources that aim to make legal data available to the public. The dataset is relatively small, consisting of 156 cases. The best ROUGE-1 score achieved in the study was 37.7 percent.

3 Data and Method

This section discusses the data and the methods used.

3.1 Data

My data are City of Miami's Commission meetings from 2022. These meetings are approximately held twice a month. Video recordings and minutes from these meetings are archived, and they are made publicly available at City of Miami website.[1]. The minutes are, on the average, 70 pages long and they are in pdf format. Although, the length of the text could indicate a difficulty in summarizing the data, the minutes are composed of distinct parts that are not topically related.

Table 1 shows the list of the sections that may be included in the minutes. Each section may contain several ordinances or resolutions which are usually summarized in a single long paragraph. In addition to the information on the ordinance or resolution itself, it may also include details about the resolution's history, such as when they were first introduced and how many times they were deferred. If there is a vote on the resolution, the distribution of votes is also recorded. The resolutions may provide information about sponsors and their financial impact on the budget.

[1]The minutes of Commission meetings are publicly available at https://www.miamigov.com/My-Government/Meeting-Calendars-Agendas-and-Comments/Commission-Agendas

Meeting Agenda Sections
Presentations and proclamations
Mayoral veto(es)
Consent Agenda
Public hearings
Discussion items
Emergency ordinance
First reading ordinances
Second reading ordinances
Boards and committees
Planning and zoning item(s)
Resolutions
Future legislation

Table 1: Sections included in the meeting minutes of the City of Miami Commission

3.2 Methods

As discussed in Section 2, local attention-based transformer models can be used to determine segments in a text. However, since the sections in my data are distinct from each other and the resolutions/ordinances in each section are independent, I opted to use regular expressions to extract section headers and the text of resolutions/ordinances.

I used abstractive methods to extract summaries from the text. Specifically, I employed three widely used pre-trained models: Bidirectional Auto-regressive Transformer (BART), Text-To-Text Transfer Transformer (T5) and Google's Pre-training with Extracted Gap-sentences for Abstractive Summarization Sequence-to-sequence model (PEGASUS) from the *Hugging Face* library.[2] Next, I fine-tuned the best performing model with my data to compare the results. Although I initially planned to include GPT models, I found that pre-trained models were only available for text generation tasks.[3] While GPT text generation models could produce summaries if "TL;DR" is appended to the end of a text, my preliminary analysis did not yield reliable results. therefore, I excluded it from my analysis in this paper.

BART implements a sequence-to-sequence transformer model as discussed in Vaswani et al. (2017) (See Figure 1 for the architecture), but with the modification of substituting ReLU functions with GeLU (Lewis et al., 2019). The model consists of 12 layers in both the encoder and decoder. Although its structure is similar to the BERT model,

[2]https://huggingface.co/models

[3]I mostly focused on *Hugging Face* libray

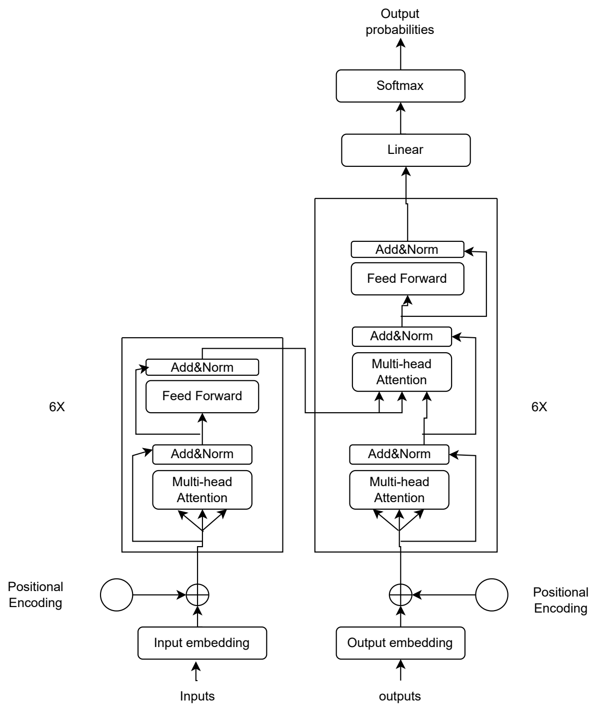


Figure 1: The transformer model architecture as described in Vaswani et al. (2017)

BART does not include a feed-forward network before the predictions.

T5 is also built on Vaswani et al.'s model, with a few key differences. First, they do not apply bias to the layer normalization. Second, they include a residual skip connection after layer normalization. And third, they utilize a simplified embedding system (Raffel et al., 2020).

PEGASUS is also a sequence-to-sequence transformer model, but it is trained differently from BART and T5. PEGASUS is trained by removing important sentences from the input and letting the model predict the missing part from the remaining sentences (Zhang et al., 2020).

After conducting experiments with these models, I fine-tuned the two best performing models with my data and compared the results with the pre-trained models.

4 Experiments

This section outlines the evaluation method, experimental setup and model implementations, and also discusses the results. The code and the data used in this paper are available in a dedicated Github repository.[4]

[4]https://github.com/hadiSahin/NLP-summarization.git

4.1 Evaluation metrics

I used the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) score to evaluate the success of the summaries. This measure compares the summary text generated by the models with summaries created by humans, focusing on n-grams and word sequences (Lin, 2004). We reported all variants of ROUGE scores, including ROUGE-1, ROUGE-2, and ROUGE-L scores.

To compare the model's summaries, I used 150 rows of resolutions that were summarized by humans out of the 958 total rows. For the remaining data, I used the summaries generated by the best performing pre-trained model, which in this case was the BART model.

4.2 Text extraction

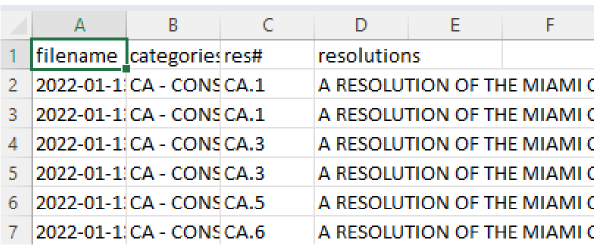
I used regular expressions to extract sections and resolution/ordinance text from the PDF files. The following expression was used to extract sections:

```
[A-Z]{2}\s*-\s*[A-Z\s]+
```

The resolution number is derived through the regular expression:

```
^[A-Z]{2}\.[0-9]{1,3}$
```

Finally, the topics discussed in the documents always start with the expressions "An ordinance of", "A resolution of", "A discussion item regarding", and "A discussion regarding". These expressions are used to search for and extract the contents of the topics.



	A	B	C	D	E	F
1	filename	categories	res#	resolutions		
2	2022-01-1:	CA - CONS	CA.1	A RESOLUTION OF THE MIAMI C		
3	2022-01-1:	CA - CONS	CA.1	A RESOLUTION OF THE MIAMI C		
4	2022-01-1:	CA - CONS	CA.3	A RESOLUTION OF THE MIAMI C		
5	2022-01-1:	CA - CONS	CA.3	A RESOLUTION OF THE MIAMI C		
6	2022-01-1:	CA - CONS	CA.5	A RESOLUTION OF THE MIAMI C		
7	2022-01-1:	CA - CONS	CA.6	A RESOLUTION OF THE MIAMI C		

Figure 2: A snippet from the data

As discussed, I focused on the City of Miami's Commission meetings for the year 2022, which comprised a total of 18 minutes of meeting time. From these meetings, I was able to extract 958 rows of independent discussion topics. The data is composed of four columns indicating the *datetime*, *category*, *resolution number*, and *text of resolutions*. Figure 2 displays an excerpt from the data. The average and median lengths of these discussions are 257.18 and 210.0, respectively. Figure 3 shows the full distributions of the extracted text.

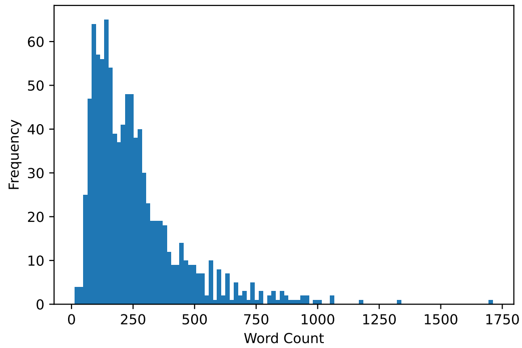


Figure 3: The distribution of lengths of extracted resolutions, ordinances and discussions

Model	ROUGE-1	ROUGE-2	ROUGE-L
BART			
R	39.0	20.3	32.7
P	29.5	14.4	16.2
F1	32.6	27.2	30.1
T5			
R	34.9	17.8	33.0
P	28.2	12.9	26.5
F1	30.2	14.4	28.5
PEG			
R	37.6	18.4	34.9
P	26.2	12.0	24.3
F1	30.2	14.2	28.0

Table 2: Pre-Trained Model Summary Results. PEGASUS is abbreviated as PEG due to space constraints.

4.3 Pre-trained models

I performed summarization tasks on the data using pre-trained T5, BART, and PEGASUS models. To prepare the text for analysis, I converted all text to lowercase, removed hyperlinks, underscores, double dashes, quotes, non-ASCII characters, and possessives. Then, for the models, I set the maximum input length to 800 tokens. For the summaries, I restricted their length to between 10 and 50 tokens.

To evaluate the models, I used ROUGE scores and compared their predictions with the first 150 rows of human summaries. The results of the evaluations are displayed in Table 2. The findings indicate that the BART model performs the best, while PEGASUS has the worst performance. The BART model utilizes a denoising autoencoder, which assists in predicting missing letters in words. In my data, some words were missing letters, and some words were broken with unnecessary spaces while extracting the text. This could be the reason why

BART was more suited to my data. Conversely, PEGASUS is trained with text missing sentences. In my text, most of the essential information is in the first few sentences, and the rest typically contains additional information or details that may not be necessary for the summary. This may explain its relatively low performance.

Based on these results, I fine-tuned the best two models with my data. Regarding the labels of the data, the first 150 rows of the summaries were manually created by humans, while the remaining rows were generated by the best performing pre-trained model BART.

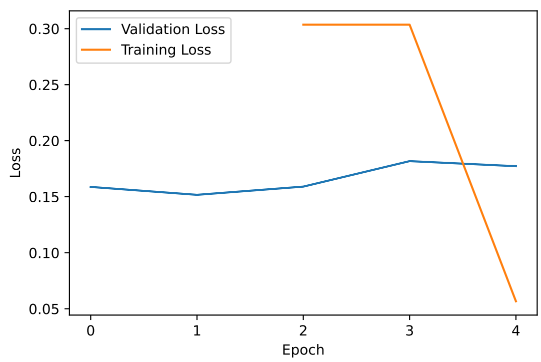


Figure 4: Evolution of training and validation loss for BART model

4.4 Fine tuning BART and T5 models

I fine-tuned the BART and T5 models with my data, excluding the first 25 rows which were reserved for evaluating the performance of all models. The models were trained with a learning rate of $2e-5$ and a batch size of 4 due to limited computational resources.

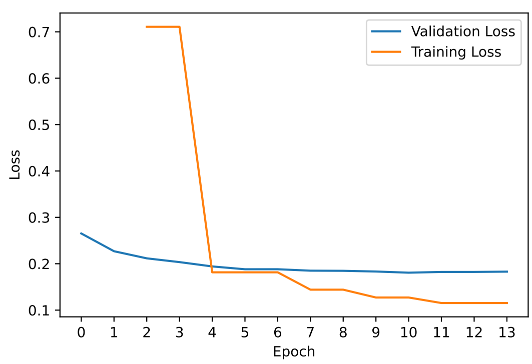


Figure 5: Evolution of training and validation loss for T5 model

Model	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**
BART			
R	37.6	14.3	33.8
P	32.2	11.9	29.2
F1	34.2	12.7	30.9
T5			
R	28.7	10.5	27.8
P	27.4	8.9	9.6
F1	27.8	9.6	26.1
PEG			
R	29.1	8.2	26.3
P	21.7	5.8	19.6
F1	24.6	6.7	22.3
Fine-tuned Models			
BART			
R	39.7	17.3	35.8
P	29.6	11.6	26.7
F1	33.3	13.6	30.0
T5			
R	41.6	17.8	38.0
P	33.0	13.5	29.6
F1	37.8	14.86	32.3

Table 3: Summaries generated by pre-trained and fine-tuned models. PEGASUS is referred to as PEG due to space constraints.

The training was conducted for 20 epochs, but I implemented *Early stopping* with a patience of 3, which meant that the training stopped when the validation loss did not improve for three consecutive epochs. Figure 4 displays the loss function of the BART model, which showed an increase in validation loss after the third epoch and was therefore stopped after five epochs.

I trained the T5 model with the same specifications, but in contrast to BART, it ran for 14 epochs. Figure 5 displays the change in validation and training loss for the model.

4.5 Results

The results of the fine-tuned models are presented in Table 3. In Table 2, I evaluated the pre-trained model using the first 150 rows of data that were summarized by humans. However, since I used most of these summaries to fine-tune the models, I reevaluated the pre-trained models on the same sample, which is the first 25 rows of data. This enabled me to make a better comparison between the performance of the pre-trained and fine-tuned models.

The results from the fine-tuned model are quite similar to the results from the pre-trained models. It appears that fine-tuning with the available data did not lead to significant improvement in model performance. This could be due to the fact that the BART model is quite large and the amount of data used for fine-tuning was not sufficient. Additionally, a large proportion of the summaries (830 out of 955, approximately 87%) were already generated by the pre-trained BART model.

T5 model performs slightly better than the fined tuned BART model and pre-trained models. However, the results does not suggest a considerable improvement.

Finally, I would like to qualitatively analyze a summary sample. Table 4 presents summaries from a randomly chosen resolution out of the 25 text samples. The table reveals that all models were quite effective in capturing the resolution's summary.

5 Conclusion and Future Work

In this paper, I examine the City of Miami Commission public meetings and created a pipeline to extract and summarize the topics discussed in these minutes. I run the summarization tasks with three pre-trained models; BART, T5 and PEGASUS. The models performed similarly, but BART slightly performed better than the other two, showing around 35 percent ROUGE-1 score. Then, I fine-tuned BART and T5 models with my data. The results show slight improvements in the predictions reaching to 40 percent levels.

In my data, the first five sentences of each paragraph are more significant in summarizing the resolutions than the rest. In most cases, the remaining sentences can be confusing, and certain patterns may be irrelevant to the summary. In my future work, I aim to explore how attention mechanisms can capture this text structure. Furthermore, human summaries are essential to evaluate the models' performance. In the future, I plan to increase the number of human-made summaries and involve multiple people to provide summaries, improving the consistency and accuracy of the human summaries.

Overall, the design and training of a model capable of summarizing or extracting keywords from official texts could significantly enhance research in social sciences. This area remains open for further research and data modeling. This paper represents an early attempt to explore this aspect.

	Summaries
Original text	resolution of the miami city commission authorizing the city manager to negotiate and execute memorandum of agreement, in form acceptable to the city attorney, with the florida department of transportation (fdot) for the installation of wayfinding iosks on fdot right ofway within the city of miami. history: 11 18 21 city commission deferred next: 01 13 22 result: deferred [unanimous] next: 13 2022 9:00 am mover: joe carollo, vice chair, district three seconder: manolo reyes, commissioner, dist rict four ayes: russell, diaz de la portilla, carollo, reyes, king city commission meeting agenda january 13, 2022 city of miami page printed on 2022 ca.4 11031 department of resilience and public works resolution.
Human summary	Miami City Commission authorizes city manager to negotiate installation of wayfinding kiosks with Florida Department of Transportation.
BART	The city commission voted to defer action on a resolution to install wayfinding iosks on the city's right ofway. The vote was unanimous. The city commission will meet again on January 13, 2022.
T5	resolution authorizing the city manager to negotiate and execute memorandum of agreement with the florida department of transportation (fdot) for the installation of wayfinding iosks on fdot right resolution deferred until jan
PEGASUS	The Miami City Commission passed a resolution authorizing the city manager to negotiate and execute memorandum of agreement with Florida Department of Transportation for the installation of wayfinding iosks on right of way within the City of Miami.

Table 4: A single excerpt from summaries. BART and T5 models are fine-tuned models while PEGASUS is pre-trained.

References

Aslihan Akkaya. 2014. Language, discourse, and new media: A linguistic anthropological perspective. *Language and Linguistics Compass*, 8(7):285–300.

Pablo Barberá, Amber E Boydstun, Suzanna Linn, Ryan McMahon, and Jonathan Nagler. 2021. Automated text classification of news articles: A practical guide. *Political Analysis*, 29(1):19–42.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. Legal-bert: The muppets straight out of law school. *arXiv preprint arXiv:2010.02559*.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. *arXiv preprint arXiv:1804.05685*.

Suzanna De Boef and Paul M Kellstedt. 2004. The political (and economic) origins of consumer confidence. *American Journal of Political Science*, 48(4):633–649.

Matthew Eshbaugh-Soha. 2010. The tone of local presidential news coverage. *Political Communication*, 27(2):121–140.

Alexios Gidiotis and Grigorios Tsoumakas. 2020. A divide-and-conquer approach to the summarization of long documents. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:3029–3040.

Dustin Hillard, Stephen Purpura, and John Wilkerson. 2008. Computer-assisted topic classification for mixed-methods social science research. *Journal of Information Technology & Politics*, 4(4):31–46.

Sergey Ivanov and Evgeny Burnaev. 2018. Anonymous walk embeddings. In *International conference on machine learning*, pages 2186–2195. PMLR.

Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. 2021a. Automatic summarization of legal bills: A comparative analysis of classical extractive approaches. In *2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, pages 394–400. IEEE.

Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. 2021b. Cawesumm: A contextual and anonymous walk embedding based extractive summarization of legal bills. In *Proceedings of the 18th International Conference on Natural Language Processing (ICON)*, pages 414–422.

Huan Yee Koh, Jiaxin Ju, Ming Liu, and Shirui Pan. 2022. An empirical survey on long document summarization: Datasets, models, and metrics. *ACM computing surveys*, 55(8):1–35.

Anastassia Kornilova and Vlad Eidelman. 2019. Billsum: A corpus for automatic summarization of us legislation. *arXiv preprint arXiv:1910.00523*.

Michael Laver, Kenneth Benoit, and John Garry. 2003. Extracting policy positions from political texts using words as data. *American political science review*, 97(2):311–331.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Potsawee Manakul and Mark JF Gales. 2021. Long-span summarization via local attention and content selection. *arXiv preprint arXiv:2105.03801*.

Stephane Meystre and Peter J Haug. 2005. Automation of a problem list using natural language processing. *BMC medical informatics and decision making*, 5:1–14.

Jonathan Pilault, Raymond Li, Sandeep Subramanian, and Christopher Pal. 2020. On extractive and abstractive neural document summarization with transformer language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9308–9319.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Olivier Salaün, Aurore Troussel, Sylvain Longhais, Hannes Westermann, Philippe Langlais, and Karim Benyekhlef. 2022. Conditional abstractive summarization of court decisions for laymen and insights from human evaluation. In *Legal Knowledge and Information Systems*, pages 123–132. IOS Press.

Yiwen Shi, Ping Ren, Yi Zhang, Xiajing Gong, Meng Hu, and Hualou Liang. 2021. Information extraction from fda drug labeling to enhance product-specific guidance assessment using natural language processing. *Frontiers in Research Metrics and Analytics*, 6:670006.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.