How Personalized is Your Summary? An Empirical Study on Student Lecture Note Summarization

Anonymous ACL submission

Abstract

Personalized text summarizaiton is one of the research hotspots of text summarization. Existing work has focused on how to incorporate the conditions of personalization into the methodology, and very little work has been done to analyze the embodiment of personalization from user real data perspective. In this paper, we conduct a personalized summarization collection experiment, which recruited 200 students who wrote 2,000 lecture note summaries. An analysis of structure, modality, content length, reference usage, and abstractiveness about summaries is conducted to address the question, "How personalized can human-written summaries be?" Additionally,we introduce a personalized text summarization dataset, and benchmark the state of art summarization models, the result indicate that the abstractive models show better performance on our dataset.

1 Introduction

Text summarization has been one of the research hotspots in the field of NLP. The rise of large language models (Zhang et al., 2023b) have also boosted the performance of text summarization. However, most existing summarization systems (Joshi et al., 2023; Zhang et al., 2023a) are data driven and rely on golden document summary training pairs, in the real world scenario the true human-written summaries vary significantly in terms of length, format and content, etc. Controllable text summarization (Li et al., 2022; Zhang et al., 2023c) can leverage some user related information such as keywords and genre in the text generation process (Taieb-Maimon et al., 2023; Móro and Bielikov', 2012). So far, there is a lack of study on the users' real needs for summarization. In this study, we recruited 200 university students and designed a task of writing personalized summaries: 10 different topics related to machine learning were given to the students, together with the corresponding lecture notes, the participants were required to write summaries based on the lecture notes. There is no hard limitation (e.g. length, reference use) on the summary written process, the only limitation is it has to be written by the participants and cannot be returned by machines. Through the analysis of the students' written summaries from different perspectives, we tend to raise the question: **How personalized can human-written summaries be?** Combining the results of the human-written summary analysis and the experiments with state-of-art summarization models , we find that the following aspects contribute to personalization: structure, modality, length, depth, reference usage and abstractiveness. Meanwhile, current end-to-end summarization models can not return personalized summary, and Longformer achieves competitive performance as ChatGPT [1] . Our main contribution is presented as follows:

- We design a study to collect personalized lecture note summaries from university students, and propose a new dataset for personalized text summarization.
- We analyse the different personalizaiton aspects of student-written summarizes and benchmark existing summarization models on this task, we show that ChatGPT is promising in this task but still needs more engineering tricks.

2 Releted Work

Personalized text summarization is controlled text summarization that is conditioning on the requirements of the user. Previous research has concentrated on combining the constraints that control the generation of summaries, such as length (Fan et al., 2018; Liu et al., 2018; He et al., 2023), aspect (Amplayo et al., 2021), style (Cao and Wang, 2021), and user's reading habits (Venington et al.,

[1] gpt-3.5-turbo-16k

2023; Yi et al., 2020), with the methods of summary generation that embody personalization. Futhermore, some researchers have designed an interactive framework for user participation in summary generation, which has the advantage of allowing users to feed their feedback into the generation of summaries, which greatly satisfies their needs(Yan et al., 2011). Previous works reflect personalization either in text summaries concerning users or in varying generation conditions. And the focus of these research work lies in the controlled setting, which is built upon the user's input from specific dimensions. It is not clear whether the controlled summarization setting is well aligned with real users' needs. Some researchers have designed questionnaires to investigate what users consider to be a good summary, and analyze the results of the questionnaires (Ter Hoeve et al., 2022; Arabzadeh et al., 2023). The result of analysis can be used as a guide for the design of the summarization models. In our work, we integrate users and data to better analyze the embodiment of personalization.

3 Personalized Lecture Note Summary Collection

Personalized summary collection The main research question for this study is: How personalized is a summary from lecture notes? In this study, we conducted the following three steps to collect a personalized summarization dataset: First, we collected machine learning lecture notes which cover ten major topics, the lecture notes are all open source and can be found from the Internet, most of them are released by public universities. Second, we recruited 200 students who were doing an Artificial Intelligence or Data Science course at the time and asked them to write summary for each topic after reading the lecture notes. In order to collect the true personalization information, there is no summary length or content limitation for the participants, i.e., they are free to write the summary without the constraints of length, style, etc. While recruiting student participants, we also collect the following information: gender, first language, qualification, machine learning working experience. Third, we validated the returned summaries by cross checking whether the summary content is aligned with the source lecture notes.

Dataset construction We constructed a dataset by combining lecture notes and student-writing summaries, as shown in Figure 1.55.5% summaries included hyperlinks to references, these links were utilized to retrieve the original text of the references and augmented them with inputs for the dataset.

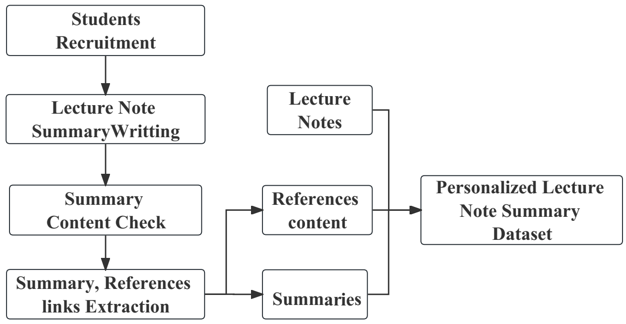

Figure 1: DataSet Construction Process

4 Summary Personalization Analysis

In this section, we analyze the returned summaries based on the following aspects: structure and modality, content length and depth, reference usage, abstractiveness as well as summary comparison with other data sets.

Summary structure and modality The lecture notes demonstrate a structured and multimodal nature. We explore whether this characteristic is consistently reflected by all students in their abstract writing, conducting an analysis focused on both structure and modality.

Summary Structure Type	Percentage
No structure	60.8%
Contain one level headings	30.1%
Contain two level headings	8.0%
Contain three level headings	1.2%

Table 1: Statistics on whether summaries contain structure.

Contain image or not	Percentage
No image	77.1%
Contain image	22.9%

Table 2: Statistics on whether summaries contain images.

In terms of structure, our primary focus is on the usage of subheadings in students' summaries, while images serve as the focal point in our analysis of modality. As shown in Tables 1 and 2, only 30% of students utilize subheadings to enhance the textual structure, and 22.9% of students incorporate images to enrich the modality of their summaries.

Summary length and depth The emphasis of the content will differ among students, influenced by their familiarity with the subject matter and the frequency of writing summaries in their study.Our initial analysis involved examining the length of summaries, as shown in Figure 2, the average number of tokens in the note summaries is 535 tokens, and the longest one can be up to 7256 tokens. However, it can be observed that the majority of returned summaries have a length of less than 1000 tokens.

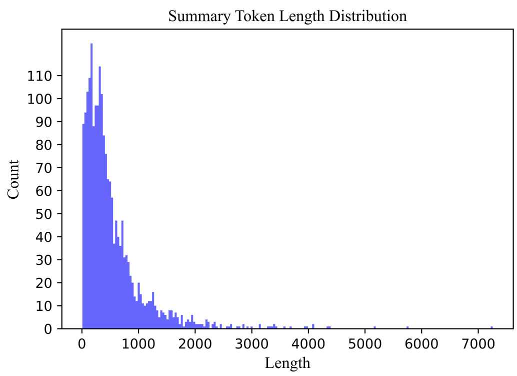

Figure 2: The summary length distribution

In addition, the in-depth details of the content are considered. We refer to the previous researchers' criteria for the details of the content of the summary and redefine them as either *General* or *Detailed* (Ter Hoeve et al., 2022): *General:* Only including material titles or summarize materials or learning process. *Detailed:* Include material contents and knowledge details. As shown in Table 3 demonstrate, students were more inclined to describe the details of their knowledge when writing their summaries. We further analyzed the detailed summary content and observed that only 5% of the students elaborated on formulas, principles, etc. when writing their summaries, and the remaining students described the definition and framework of knowledge. At the same time, we analyzed the 10 summaries written by individuals and found that each individual had a different focus when writing them. For example, some individuals tended to focus on model principles while others preferred to systematize knowledge under a specific topic.

Reference usage Students were advised that they could utilize additional resources as references during the experiment. Our scrutiny of the gathered summaries centered on the existence of reference links. The findings, illustrated in Figure 4, reveal that out of 1,102 summaries, many included supplementary references. Web pages were the most prevalent source among the reference links, followed by books and journals in Figure 5.

Summary depth	**Percentage**
General	62.2%
Detailed	37.8%

Table 3: Statistics on the summary content depth : general or detailed.

References or not	**Percentage**
No references	55.5%
Contain references	44.5%

Table 4: Statistics on whether summaries contain references.

Sources of References	**Percentage**
Web page	72.5%
Journals and Book	14.9%
video	12.6%

Table 5: Statistics on sources of references

Abstractiveness Abstractiveness measures how effectively the abstract summarizes the source text, indicating whether the user has directly replicated the source text when crafting the abstract. To evaluate the abstractiveness of the target summaries, we quantify it by calculating the percentage of novel n-grams in the summaries that do not appear in the source text. Additionally, we have chosen several classic text summarization datasets, including CNN-DM (Nallapati et al., 2016), arXiv (Cohan et al., 2018), PubMed (Cohan et al., 2018), for comparison. Table 7 displays an increased presence of novel trigrams and 4-grams (Liu et al., 2023). The abstractiveness of our dataset's target summaries surpasses that of other datasets.

Comparison with common datasets We also adopt five measures used to evaluate our dataset shown in Table 8: compression ratio, extractive coverage, extractive density(Grusky et al., 2018), redundancy and uniformity(Koh et al., 2022). Since all corresponding input data are included in calculating the metrics for the dataset, the length of input documents is a factor, resulting in a high compression ratio compared to other datasets, and relatively

Model	Rouge-1	Rouge-2	Rouge-L	BERTScore
TextRank (10k)	15.08	7.04	7.25	77.24
SummPip (10k)	22.30	3.46	11.15	79.69
Longformer (10k)	30.54	**9.13**	**15.65**	83.58
PEGASUS-X (10k)	6.72	0.95	5.53	76.61
gpt-3.5-turbo-16k (10k)	**31.97**	8.23	14.66	**83.89**

Table 6: Automatic evaluation results on test sets of our dataset

Dataset	**% of novel n-grams in target summary**			
	unigrams	**bigrams**	**trigrams**	**4-grams**
CNN/DM	19.5	56.88	74.41	82.83
PubMed	18.38	49.97	69.21	78.42
arXiv	15.04	48.21	71.66	83.26
Our	13.87	54.29	**79.67**	**88.32**

Table 7: The proportion of novel n-grams in target summaries

Dataset	**CNN-DM**	**PubMed**	**arXiv**	**Our**
Compression	8.3	15.6	44.3	87.3
Coverage	0.89	0.893	0.92	0.86
Density	3.6	5.6	3.7	4.13
Redundancy	0.157	0.146	0.144	**0.115**
Uniformity	0.856	0.896	0.894	**0.95**

Table 8: Comparison of Summarization Datasets

low coverage ratio.

5 Summarization Model Evaluation

Baseline models Our dataset includes 2000 lecture note summary pairs, We use 80% of the data for training, 10% for validation and 10% for test. Both state of art extractive and abstractive models are evaluated. The extractive models we use are: TextRank (Mihalcea and Tarau, 2004), a classical extractive summarization model,was opted. Subsequently, considering the task as akin to multi-document summarization, we selected Summpip (Zhao et al., 2020), which exhibits enhanced effectiveness in the unsupervised domain. For thee abstractive models which are mostly rely on pre-trained language models, we consider Longformer-Encoder-Decoder (Longformer)(Beltagy et al., 2020), PEGASUS-X (Phang et al., 2023). Moreover, the increasing prominence of large language models in text summarization research has led us to select ChatGPT (gpt-3.5-turbo-16k), a highly acclaimed model developed by OpenAI for our evaluation.

Result We report ROUGE scores and BERTScore (Zhang et al., 2020) for evaluating the summarization quality. Table 6 shows the task of personalized text summarization is challenging for both extractive and abstractive approaches. Also, Longformer and gpt-3.5-turbo returns similar performance, with gpt-3.5-turbo achieves higher in Rouge-1 and BERTScore and Longformer returns higher Rouge-2 and Rouge-2 score. Surprisingly, PEGASUS-X exhibits low Rouge scores as we see repetitive characters and short summary generation but the BERTScore result shows the generated summaries are semantically correlated with student written summaries. In general, the abstractive models show better performance on our dataset but still lie behind from those news summarization tasks.

6 Conclusion

We conduct a study for personalized text summarization of lecture notes, and analyze the personalization of these summaries across several dimensions, including structure, modality, content length, reference usage, and abstractiveness. We also introduce a personalized text summarization dataset, and benchmark the state of art summarization models. The findings and dataset may provide a valuable reference and resource for the community to design educational interactive systems and conduct methodological studies on personalized text summarization.

Limitation

As this study involves human participants, there are a few limitations: First, the topics of this summarization data set are only limited to machine learning. Second, due to the expensive annotation cost, we only recruited 200 participants. Third, we on experimented with common summarization model and did not try any controlled summarization models, this can be our future work.

Ethics Statement

Data collection approval was received from an ethics review board. No identified personal information are collected in the data collection process. All codes and data used in this paper comply with the license for use.

Acknowledgments

This paper was supported by the National Natural Science Foundation of China (project numbers are 72274096, 71774084, 72301136, and 72174087), the Foreign Cultural and Educational Expert Program of the Ministry of Science and Technology of China (G2022182009L).

References

Reinald Kim Amplayo, Stefanos Angelidis, and Mirella Lapata. 2021. Aspect-controllable opinion summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6578–6593, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Negar Arabzadeh, Ali Ahmadvand, Julia Kiseleva, Yang Liu, Ahmed Hassan Awadallah, Ming Zhong, and Milad Shokouhi. 2023. PREME: Preference-based meeting exploration through an interactive questionnaire. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 331–342, Dubrovnik, Croatia. Association for Computational Linguistics.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Shuyang Cao and Lu Wang. 2021. Inference time style control for summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5942–5953, Online. Association for Computational Linguistics.

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. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics.

Angela Fan, David Grangier, and Michael Auli. 2018. Controllable abstractive summarization. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 45–54, Melbourne, Australia. Association for Computational Linguistics.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. 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 708–719, New Orleans, Louisiana. Association for Computational Linguistics.

Pengcheng He, Baolin Peng, Song Wang, Yang Liu, Ruochen Xu, Hany Hassan, Yu Shi, Chenguang Zhu, Wayne Xiong, Michael Zeng, Jianfeng Gao, and Xuedong Huang. 2023. Z-code++: A pre-trained language model optimized for abstractive summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5095–5112, Toronto, Canada. Association for Computational Linguistics.

Akanksha Joshi, Eduardo Fidalgo, Enrique Alegre, and Laura Fernández-Robles. 2023. Deepsumm: Exploiting topic models and sequence to sequence networks for extractive text summarization. *Expert Systems with Applications*, 211:118442.

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

Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori Hashimoto. 2022. Diffusion-lm improves controllable text generation. *ArXiv*, abs/2205.14217.

Shuaiqi Liu, Jiannong Cao, Zhongfen Deng, Wenting Zhao, Ruosong Yang, Zhiyuan Wen, and Philip S. Yu. 2023. Neural abstractive summarization for long text and multiple tables. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–14.

Yizhu Liu, Zhiyi Luo, and Kenny Zhu. 2018. Controlling length in abstractive summarization using a convolutional neural network. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4110–4119, Brussels, Belgium. Association for Computational Linguistics.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Róbert Móro and M'ria Bielikov'. 2012. Personalized text summarization based on important terms identification. In *2012 23rd International Workshop on Database and Expert Systems Applications*, pages 131–135.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Jason Phang, Yao Zhao, and Peter Liu. 2023. Investigating efficiently extending transformers for long input summarization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3946–3961, Singapore. Association for Computational Linguistics.

Meirav Taieb-Maimon, Aleksandr Romanovski-Chernik, Mark Last, Marina Litvak, and Michael Elhadad. 2023. Mining eye-tracking data for text summarization. *International Journal of Human–Computer Interaction*, pages 1–19.

Maartje Ter Hoeve, Julia Kiseleva, and Maarten Rijke. 2022. What makes a good and useful summary? Incorporating users in automatic summarization research. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 46–75, Seattle, United States. Association for Computational Linguistics.

K Veningston, P V Venkateswara Rao, and M Ronalda. 2023. Personalized multi-document text summarization using deep learning techniques. *Procedia Computer Science*, 218:1220–1228. International Conference on Machine Learning and Data Engineering.

Rui Yan, Jian-Yun Nie, and Xiaoming Li. 2011. Summarize what you are interested in: An optimization framework for interactive personalized summarization. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1342–1351, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Kun Yi, Yu Guo, Zhi Wang, Lifeng Sun, and Wenwu Zhu. 2020. Personalized text summarization based on gaze patterns. In *2020 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 307–313.

Haopeng Zhang, Xiao Liu, and Jiawei Zhang. 2023a. DiffuSum: Generation enhanced extractive summarization with diffusion. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13089–13100, Toronto, Canada. Association for Computational Linguistics.

Haopeng Zhang, Xiao Liu, and Jiawei Zhang. 2023b. Extractive summarization via ChatGPT for faithful summary generation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3270–3278, Singapore. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Yusen Zhang, Yang Liu, Ziyi Yang, Yuwei Fang, Yulong Chen, Dragomir Radev, Chenguang Zhu, Michael Zeng, and Rui Zhang. 2023c. MACSum: Controllable Summarization with Mixed Attributes. *Transactions of the Association for Computational Linguistics*, 11:787–803.

Jinming Zhao, Ming Liu, Longxiang Gao, Yuan Jin, Lan Du, He Zhao, He Zhang, and Gholamreza Haffari. 2020. Summpip: Unsupervised multi-document summarization with sentence graph compression. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 1949–1952, New York, NY, USA. Association for Computing Machinery.

A Appendix

A.1 Examples of student-writting summaries about lecture notes

We give some examples of student-writting summaries about one lecture note in the Table 9.

A.2 An Example of generated summaries about baseline models

We give an example of generated summaries for every baseline model in the Table 10.

Lecture note:Defining Machine Learning What makes Machine Learning (ML) different from a simple computer program? The 1959 definition below makes Machine Learning (ML) seem like magic: “Field of study that gives computers the ability to learn without being explicitly programmed,”(Samuel 1959) As we explore further you will see ML is a set of tools to derive meaning from data. For now, here is another definition that describes ML in mathematical terms. Machine Learning as an equation “A computer program is said to learn from experience with respect to some class of task and performance measure , if its performance at tasks in , as measured by , improves with experience ,”(Mitchell 1997, p. 2) This is an elaborated, perhaps overly complicated definition of how a computer program learns. It learns only when performance improves with experience. So, what is experience? It is the data that we provide for the machine to process...
Summary with no reference:A basic introducton to Machine Learning was covered this week. I learned the definiton of machine learning and its types from the lecture. Also learned about the types of machine learning which the following:Supervised learning, Unsupervised learning, and Reinforcement learning. A major difference between supervised and unsupervised learning is the data used for training. Models are trained using labeled data for supervised learning,whereas unsupervised models are trained using unlabeled data. Moreover, supervised learning is classified into two categoriesclassificaton and regression. At the same tme,unsupervised learning is classified asclustering and associaton analysis...
Summary with references:In this topic, we have come around some of the basics about Machine Learning. Machine Learning or (ML for short) is defined as field of study where computers could learn without explicitly programmed. It learns from some experience E with respect to some tasks in T, as measured by P, improves with experience E. Some of its real-world applicaFons include in the field of RoboFcs, Computer Vision, Speech RecogniFon, Board games etc. We later learned about the kind of steps a ML Algorithm usually takes; some of the main stages include Data ManipulaFon, Data AnalyFcs, and Data VisualizaFon. Secondly, we learned about some of the three main types of ML Algorithms: namely...

Table 9: Examples of student-writting summaries

Dataset Inupt:Boosting Boosting methods are built sequentially, and one tries to reduce the bias of the combined estimator. The motivation is to combine several weak models to produce a powerful ensemble.More detailsAdaBoostAdaBoostConde exampleRequire library for AdaBoost classifierfrom sklearn.ensemble import AdaBoostClassifierfrom future import printfunctionfrom ipywidgets import interact, interactive, fixed, interactmanualfrom IPython.display import clearoutput, displayimport ipywidgets as widgetsCode ExampleTrain the AdaBoost model with a number of estimators is and a learning rate...
Dataset Outpt:A. Ensemble learning.The process by which multiple models, such as classifiers or experts, arestrategically generated and combined to solve a particular computationalintelligence problemii.Random Forest To reduce variance, train multiple decision trees, each withslightly different subsets of data, use classificationregression to takecombined decisions via averaging for regression or voting for classificationB. Bootstrap Estimation...
Textrank:More detailsAdaBoostAdaBoostConde exampleRequire library for AdaBoost classifierfrom sklearn.ensemble import AdaBoostClassifierfrom future import printfunctionfrom ipywidgets import interact, interactive, fixed, interactmanualfrom IPython.display import clearoutput, displayimport ipywidgets as widgetsCode ExampleTrain the AdaBoost model with a number of estimators is and a learning rate is .def fnestimators,learningrate abc AdaBoostClassifiernestimatorsnestimators,learningrate learningrate, randomstate abc.fit Xtrain...
Summpip:you can be of two s and five s s and s are our classes and desire to separate the classes using their features . to increase the models performance , feature importance utilising random forest . this way , the model with a number of iterations weak classifier htversion . ,mathmath xmlnshttp-www.w.orgmathmathmlmsubmizmimitmimsubmath is a normalization constant that ensures the weights sum to z t for each i , , . the first place , random forest uses the ensemble technique called bagging . as you can see , different independent classifiers voted for different boundaries light green , but if you take the aggregation of these boundaries dark green it will clean up and improve the final decision boundary . figure .
LED:This topic, we learned about a variety of machine learning techniques, including random forest, ensemble learning, and out of bag error and feature importance. We also learned about the random forest algorithm, which is a machinelearning algorithm for classification problems, and how it can be used to build a strong classifier by combining weak classifiers to create a strongclassifier. We learned about how to use random forest in Python to train a random forest classifier, and the advantages and disadvantages of random forest...
Pegasus-X:BootstrapAdaBoost - Adaptive boostingBaggingRandom forest
ChatGbt:Boosting methods, such as AdaBoost, are sequential models that aim to combine weak models to create a powerful ensemble. The AdaBoostClassifier from the sklearn.ensemble library is used as an example in the code. The model is trained with a given number of estimators and learning rate. The accuracy of the model on test data is printed...

Table 10: An example of generated summaries for every baseline model