# Detection of RNA Editing Sites by GPT Fine-tuning

**Zohar Rosenwasser**
Bar-Ilan University
Israel
rosenwz@biu.ac.il

**Erez Y. Levanon**
Bar-Ilan University
Israel
erez.levanon@biu.ac.il

**Michael Levitt**
Stanford University
United States
levittm@stanford.edu

**Gal Oren**
Stanford University, Technion
United States
galoren@stanford.edu

## Abstract

Accurately predicting RNA editing sites is crucial for leveraging endogenous base editing technologies for therapeutic applications. This study introduces a novel methodology leveraging advanced AI techniques, specifically OpenAI's GPT-3.5, to predict both the occurrence and efficiency of RNA editing by base editors such as ADAR enzymes. By fine-tuning GPT models on extensive datasets of RNA sequences and secondary structures, we observe improvements in predictive accuracy, with our approach outperforming existing approaches. Our approach involves framing the problem in two distinct ways: as a generation problem, predicting new edited structures, and as a classification problem, determining if specific sites are edited. We also implement robust data augmentation strategies and threshold adjustments to optimize the model's performance. Our findings highlight the transformative potential of GPT in solving complex biological problems, providing a robust framework for future genetic interventions.[1]

## 1 Introduction

RNA editing is a post-transcriptional modification process that allows for the dynamic regulation of genetic information. These modifications can lead to codon changes, alternative splicing, and altered RNA secondary structures, thereby profoundly impacting gene expression and function [6, 17]. Base editors, which modify specific nucleotides within RNA or DNA sequences, hold significant potential for therapeutic applications [28, 37]. One notable example of RNA editing is adenosine-to-inosine (A-to-I) editing mediated by ADAR (adenosine deaminases acting on RNA) enzymes. ADAR enzymes modify specific adenosines within double-stranded RNA (dsRNA) regions to inosines, which are recognized during translation as guanosines. This type of editing is most common in humans, with many millions of known sites [3, 13, 4]. Recent advancements in technology have further enhanced the application of A-to-I RNA editing for both technological and medical purposes [26, 2, 14, 30]. Accurately predicting RNA editing sites by base editors that are based on endogenous editors is crucial for therapeutic applications. However, predicting the efficiency and specificity of RNA editing is challenging due to the complex interplay between RNA sequence and secondary structure. Effective utilization of these enzymes requires designing RNA constructs with specific secondary structures to guide editing to desired sites. The difficulty is in predicting which secondary structure configurations will lead to successful and targeted editing outcomes. Recent advancements have been

[1]The sources of this work are available at our repository: https://github.com/Scientific-Computing-Lab/GPT_RNA_Editing_Detection.

AI for New Drug Modalities Workshop (AIDrugX), NeurIPS 2024.

made in predicting RNA-binding protein sites [9]. Advanced learning methods have been employed to address these challenges, leveraging features derived from RNA sequences and structures to predict interactions and functions [16, 33]. While these models have shown some success in predicting RNA-protein interactions, they have not been extensively tested for RNA editing. Additionally, the vast number of regions where ADAR binds makes it difficult for current methods to achieve precise resolution at the nucleotide level, further complicating accurate RNA editing predictions.

Recent advances in generative artificial intelligence (GenAI), particularly the development of GPT large language models (LLMs) [21], offer a promising solution to this challenge. *Could the transformative potential of GPT LLMs, renowned for their ability to decode and interpret complex patterns in data, extend to the realm of RNA editing prediction?* Given their extensive pretraining on diverse datasets and their exceptional optimization capabilities, we hypothesize that properly fine-tuning these models on specific RNA editing tasks could unlock unprecedented accuracy and insight. By leveraging GPT-3.5 to focus on RNA sequences and their secondary structures as a downstream task, we aim to explore whether these models can excel not just in language generation or classification, but in predicting RNA editing efficiency as both a generation and classification task.

## 2 Previous Work

RNA editing has been the focus of many studies, utilizing both experimental and computational approaches. The traditional identification of RNA editing sites relied on resource-heavy methods like RNA sequencing and sequence alignment [15, 27, 29], making it an expensive process. In contrast, the prediction of editing sites offers a computational approach that does not require intensive experimental procedures. Rule-based models, which utilized previously identified criteria—such as the presence of a cytosine opposite adenosine in a loop structure—provided a framework for prediction but offered limited accuracy. For instance, these models required the presence of a cytosine (C) opposite an adenosine (A) with no guanine (G) preceding the adenosine [32, 12, 25]. Although such models achieved modest accuracy (Table 1-*Baseline*), they highlighted the limitations of relying solely on predefined rules. Recent studies have explored features influencing RNA editing efficiency, finding that mismatches at specific positions can affect the editing levels. While these studies identified statistically significant characteristics [42], they do not provide sufficient predictive accuracy for determining the precise locations and levels of editing sites. These findings suggest that additional, undiscovered rules governing the ADAR enzyme's behavior may exist, and AI-driven approaches have the potential to uncover these patterns, leading to improved prediction accuracy.

Recent advancements have seen the application of machine learning methods like support vector machines and random forests to RNA editing prediction [8, 34, 35]. While these methods show promise, they typically require extensive feature engineering and often fail to fully capture the complex interactions within RNA sequences and their secondary structures.

In contrast, LLMs such as GPT-3.5, GPT-4 [7, 1], and LLaMA-3.1 [11], which are built on the Transformer architecture, have demonstrated significant advances in AI. These models excel in tasks like translation, summarization, and nuanced question answering. In biomedicine, LLMs have been employed for diverse tasks such as diagnostic assistance, drug discovery, and processing biomedical literature [38]. For instance, models like BioGPT [24] have been fine-tuned specifically for biomedical text generation and question answering, while PubMedBERT [18], trained on biomedical literature from PubMed, has enhanced tasks like named entity recognition and relation extraction.

In genomics, LLMs are increasingly used to model DNA and RNA sequences, providing powerful tools for sequence interpretation, variant calling, and gene expression analysis. Tools such as DNABERT [43] and scBERT [41] have demonstrated the potential to capture biological sequence patterns and create embeddings that enhance predictive accuracy. For example, scBERT has made notable contributions by generating detailed gene embeddings for single-cell RNA-seq data.

The progress in LLMs suggests a strong potential for improving RNA editing predictions. Traditional methods often struggle with the complex interactions between RNA sequences and their structures. LLMs, with their ability to understand complex patterns and relationships, offer a promising approach to enhancing prediction accuracy. Although there are various language models in the RNA field, such as BERT-RBP [40] for predicting RNA-RNA binding protein (RBP) interactions, the case of predicting RNA editing sites is fundamentally different. In RNA editing prediction, the goal is not to locate regions where the binding will occur but to predict whether a specific site will undergo editing

by enzymes like ADAR. This requires a deeper understanding of the sequence and structural features that influence the editing process.

**Contribution:** In this paper, we introduce a novel methodology for predicting RNA editing sites by framing the problem in two distinct ways: As a generation problem and as a classification problem. For the generation problem, we predict the new edited structure given the RNA secondary structure in Vienna format [23], which GPT models are familiar with. For the classification problem, we determine whether a highlighted potential editing site in the ViennaRNA structure has been edited or not. It is important to note that our approach focuses on predicting RNA editing sites rather than finding them, which is the primary objective of most existing methods. To address these challenges, we develop a robust data augmentation strategy that allows for the creation of new data instances while preserving the true structure. Additionally, for the classification problem, we implement threshold adjustment optimization based on GPT token log probabilities. This optimization enhances the performance and reliability of the model's predictions. Through these contributions, we aim to provide a comprehensive and effective solution for predicting RNA editing sites, leveraging the strengths of both generative and classification approaches.

## 3 Methodology: GPT Fine-Tuning for Detection of RNA Editing Sites

Our study leverages the GPT-3.5 model to predict RNA editing efficiency, focusing on integrating RNA secondary structures in the Vienna format. We chose the Vienna format as GPT effectively processes this form, enabling it to learn intricate patterns within RNA secondary structures. The GPT model can be fine-tuned for both generation and classification tasks [19], each addressing different aspects of the RNA editing prediction problem (as described in Figure 1).

### 3.1 Generation Approach: Predicting RNA Edited Structures

In the generation approach, the GPT model takes RNA structure in Vienna format (sequence and secondary structure) [23] as input and generates modified sequences that indicate potential RNA editing events. This method predicts edited RNA sequences and marks the editing sites within the sequence. Given the limited amount of available data, data augmentation was crucial, as it prevented the model from converging on predicting the edited structure alone due to the scarcity of editing sites, thereby ensuring a more balanced focus on both the edited and non-edited sites. The data augmentation process begins by (1) identifying key areas, focusing on sequences within base-pairing regions that are at least $B = 5$ bases away from editing sites. We then (2) modify these sequences while preserving base pairing and (3) ensure the secondary structure remains consistent post-modification. Specifically, for each RNA secondary structure, we counted all base pairs located at least $B$ bases away from an editing site and randomly selected a number of these pairs to modify by swapping between the bases of each chosen pair. Let $N$ be the total number of base pairs at least $B$ bases away from an editing site, and $k$ be a randomly selected number where $0 \leq k \leq N$. The number of possible sequences generated, $S$, is given by the sum $S = \sum_{k=0}^{N} \binom{N}{k} = 2^N$, where $\binom{N}{k}$ represents the number of options to choose $k$ pairs out of $N$. This formula enables the generation of multiple augmented sequences, thereby increasing the diversity of our training data.

### 3.2 Classification Approach: Binary Classification of Individual Editing Sites

In the classification approach, we frame the problem as a binary task, predicting whether a nucleotide within a specific RNA sequence context is edited. To enlarge the training dataset, we utilize data augmentation by generating sequence windows around potential editing sites. These inputs consist of RNA sequences with annotated windows around potential editing sites, and the outputs provide binary predictions indicating whether editing occurs. This method contrasts with the generation approach by focusing on creating distinct training examples for each potential editing site, enhancing the representation of structural and contextual variations specific to individual editing events.

After training, we refine model performance by optimizing prediction thresholds, known as threshold adjustment. We assess the confidence levels of the model's responses by analyzing the model's output probabilities for the possible predictions. Specifically, we compute the probabilities associated with each outcome (i.e., "Yes" or "No" for editing sites, which also account for exactly one token each) to

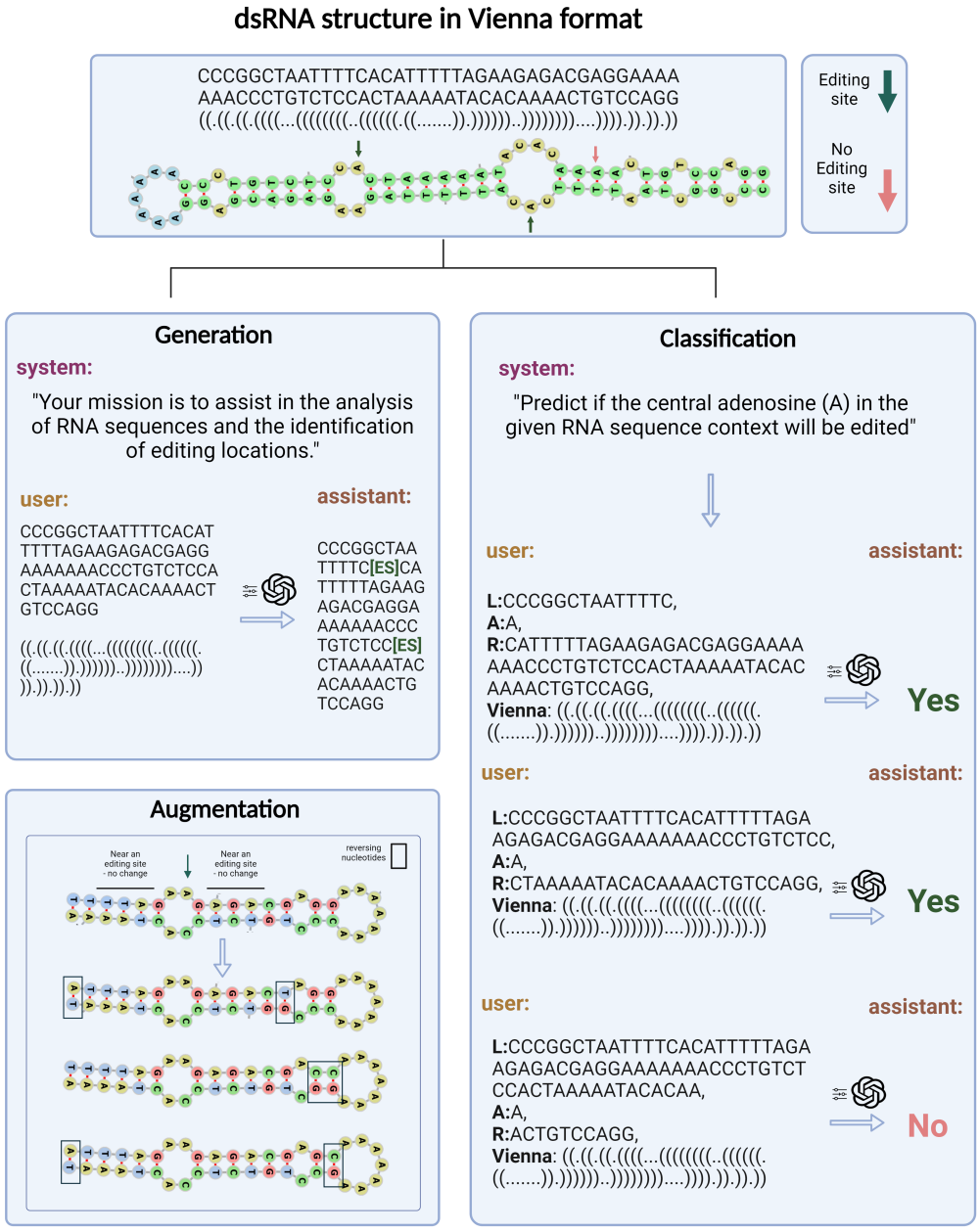



Figure 1: Workflow of RNA editing prediction using GPT model: In the generation approach, the fine-tuned model generates modified RNA sequences with potential editing sites. In the classification approach, the fine-tuned model predicts whether the central adenosine (A) in a given context will be edited (Y/N). Data augmentation involves altering sequences in non-editing regions to enhance training data while preserving RNA structure.

measure the model's certainty. Through an iterative process, we adjust these thresholds to balance sensitivity and specificity, enhancing the accuracy and reliability of RNA editing site identification.

## 4 Use Case: ADAR A-to-I RNA Editing Sites within *Alu* Elements

**Problem Definition and Data Curation:** The primary challenge is designing an RNA oligonucleotide (gRNA) that will hybridize to a target in the genome, recruit ADAR, and efficiently perform editing. The ultimate goal is to harness this system for RNA therapy and mutation correction, enabling precise editing of specific mutations to potentially cure genetic diseases. In this work, we focused on the

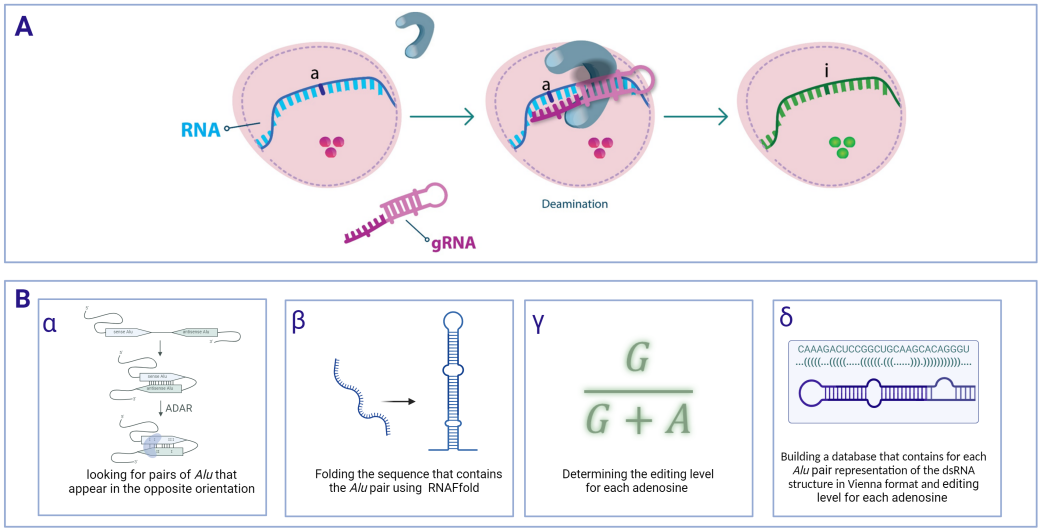



Figure 2: Site-directed RNA editing by oligonucleotide insertion and schematic description of the analysis of editing sites within *Alu* elements. (A) ADARs can be directed to specific nucleotides within the transcriptome to manipulate RNA sequences without altering the genome. The oligonucleotide used comprises two domains — a specificity domain, complementary to the host site, and an ADAR-recruiting domain, which forms dsRNA to recruit endogenous ADAR. (B) Schematic representation illustrating the prevalence of editing sites within repetitive *Alu* SINEs and their potential to form dsRNA structures, attracting ADAR enzymes: ($\alpha$) set of 1810 *Alu*s were selected for structural analysis based on criteria explained in the main text, ($\beta$) visualization of the secondary structures predicted in selected *Alu* pairs using RNAFold, ($\gamma$) quantification of editing frequencies in 905 dsRNA structures, based on RNA-seq data derived from the GTEx database, and ($\delta$) building a database that consists of the secondary structures in Vienna format of all *Alu* pairs and the editing levels for each of the adenosines in Alu.

editing sites located within *Alu* elements [10] to explore the effect of features within the dsRNA structure on ADAR-mediated editing levels. Virtually all editing sites reside within the repetitive *Alu* SINEs. *Alu* sequences, which average about 300 bases in length, are the dominant repeat in the human genome, with over a million copies, and thus are likely to pair with neighboring reversely oriented repeats and form dsRNA structures that are bound by ADAR enzymes [5]. Determining the dsRNA structure that recruits ADAR is not trivial since, at any given moment, several different folding combinations can be formed. Thus, we chose to look at selected *Alu* pairs in which the dsRNA structure can be determined with high confidence. Pairs that meet the following conditions were chosen (Figure 2): (1) located in UTRs, (2) oppositely oriented, and (3) lacked any other *Alu* elements in the UTR. We found 1,810 edited *Alu* elements that met all the criteria. In the next step, we predicted the secondary structure formed in *Alu* pairs using RNAFold [31], and we determined the editing frequency in each adenosine in the 905 dsRNA structures using RNA-seq data from the GTEx database [22] (spanning 9,125 samples from 47 tissues, collected from 548 donors). The total amount of adenosines in the data is 127,015, of which 14,205 are edited by over 15%. We specifically chose sites with over 15% editing to ensure that the editing events were biologically significant and not due to random chance.

**Generation Model — Training and Evaluation**: Initially, we employed GPT to predict RNA editing sites by fine-tuning it on data formatted in a question-answer structure of complete sequences. Despite our efforts, the initial training yielded suboptimal results, characterized by overfitting and poor generalization. To address the overfitting issue, we expanded our dataset through data augmentation. By modifying non-editing regions within sequences while preserving the secondary structure, we increased our dataset from 905 to 9050 examples (for each sequence we generated 100 different sequences). We also experimented with varying sizes of batch, reducing from 128 to 8, and tested GPT-3.5-0125 version. Despite these efforts, the results remained unsatisfactory: The generative model achieved an overall accuracy of 97.6%, while it struggled with precision at 63.7%, recall at 30.4%, and an F1 score of 41.1% (Table 1-*Generation*). The high accuracy is largely due to the

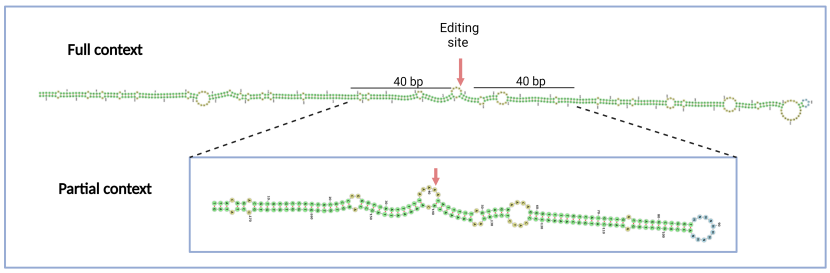



Figure 3: Illustration of the structural differences between the full and partial contexts surrounding RNA editing sites. The partial consists of 40 bases on each side of the potential editing site.

Table 1: Comparison of model performance metrics for ADAR A-to-I RNA editing sites within *Alu* elements: Accuracy (ACC), Precision (PRE), Recall (REC), and F1 score (F1) for different fine-tuned models. The GPT-based model exceeded performance over non-AI baseline metric by an order of magnitude (in F1 score), up to more than 70% in the GPT-3.5 classification fine-tuning model with the necessary threshold adjustment ($T = 0.29$).

| | **ACC** | **PRE** | **REC** | **F1** |
|---|---|---|---|---|
| **Baseline (Non-AI-based method)** | 49.7% | 46.0% | 2.80% | 5.29% |
| **Generation** | 97.6% | 63.7% | 30.4% | 41.1% |
| **Classification-Default (Partial context)** | 61.9% | 70.5% | 40.9% | 51.8% |
| **Classification-Default (Full context)** | 64.8% | 64.7% | 65.0% | 64.8% |
| **Classification-MaxF1 (Partial context)** | 58.3% | 55.0% | 91.2% | 68.6% |
| **Classification-MaxF1 (Full context)** | 61.9% | 57.0% | 96.3% | **71.6**% |
| **Classification-Intersection (Partial context)** | 63.5% | 63.5% | 63.4% | 63.4% |
| **Classification-Intersection (Full context)** | 64.8% | 64.7% | 65.0% | 64.8% |

model's success in predicting many parts of the sequence that are identical in input and output. This is reflected in the low recall, indicating poor performance in correctly identifying true editing sites. Additionally, the generative model experienced overfitting, particularly beyond 500 training steps.

**Classification Model — Training and Evaluation**: To better utilize the GPT model for detecting RNA editing sites, we re-framed the problem as a binary classification task. This involved generating sequence windows around potential editing sites and creating training examples for edited versus non-edited sites. We explored different classification model configurations to address these challenges.

We employed two context methods: Partial and full context (Figure 3). In the partial context method, we considered 40 bases on each side of the potential editing site, capturing a localized sequence window. This choice was based on previous research indicating that mismatches up to 35 bases after the editing site can significantly increase editing levels [42]; hence, we extended the range slightly to ensure coverage. In the full context method, we used the entire sequence of the dsRNA. This approach allowed us to assess how the length of the sequence context influences model performance.

To ensure fair evaluation and effective learning, we implemented balanced classification techniques. Before balancing, the partial context dataset had 8,370 instances labeled "Yes" and 54,535 instances labeled "No". After balancing, both "Yes" and "No" classes had 8,370 instances. In the full context model, which had slightly more data to begin with, both positive and negative instances were balanced at 14,205 instances. This approach addressed the inherent imbalance in the dataset, which contained many more negative instances than positive ones. This imbalance occurs because the ADAR enzyme specifically edits adenosines, but not every adenosine is been edited. Additionally, we chose to consider adenosines as edited only if they had over 15% editing, further reducing the number of edited instances, as most adenosines in *Alu* regions are edited at a low level.

The results of our experiments appear in Table 1-*Classification* and are explained hereafter: Using the default threshold with partial context, the model achieved an accuracy of 61.9%, precision of 70.5%, recall of 40.9%, and an F1 score of 51.8%. Expanding to full context improved performance, with an accuracy of 64.8%, precision of 64.7%, recall of 65.0%, and an F1 score of 64.8%. Further optimization involved adjusting the classification threshold to maximize the F1 score (Figure 5a).

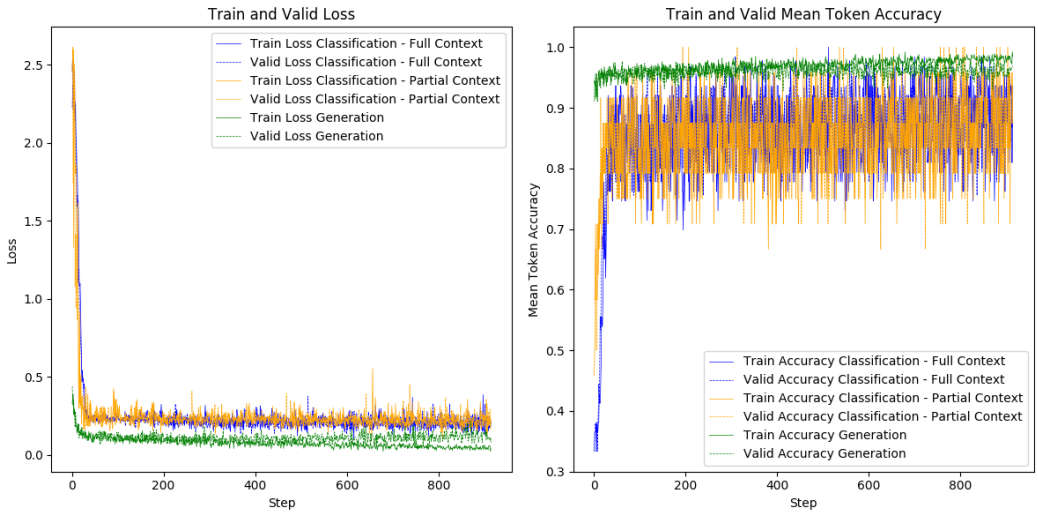


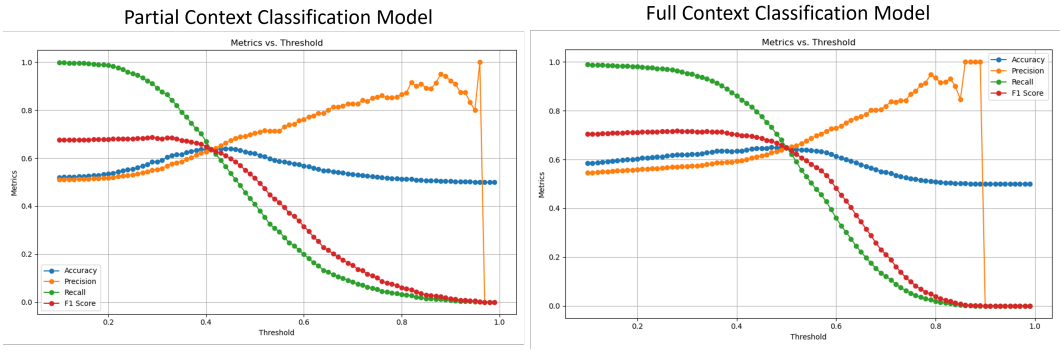

Figure 4: Training and validation performance of fine-tuned GPT-3.5 models for detecting RNA editing sites: The left plot shows the training and validation loss across steps for the same dataset but with different approaches: Classification with full context; Classification with partial context; and Generation. The right plot presents the train and validation mean token accuracy for the same datasets. The trends demonstrate the fine-tuned models' convergence behavior and performance stability across different datasets.

This was done by analyzing the predicted probabilities and selecting a threshold that best-balanced precision and recall. With partial context, this yielded an accuracy of 58.3%, precision of 55.0%, recall of 91.2%, and an F1 score of 68.6%. Using full context and optimizing the threshold resulted in the best performance, with an accuracy of 61.9%, precision of 57.0%, recall of 96.3%, and an F1 score of 71.6%.

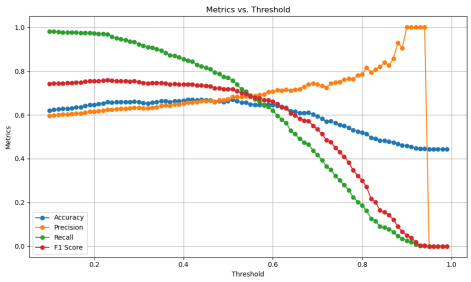
Threshold adjustment typically aims to maximize the F1 score, but we also evaluated the performance at the cut-off point to achieve high indices across other important metrics. The point of maximum F1 was at $T = 0.29$, while at the intersection point, it rose to $T = 0.4$ for the partial model and $T = 0.5$ for the full model, which is essentially the same as the default model. This approach ensures that neither metric is disproportionately favored, leading to a more reliable model for practical applications. At the intersection point, the classification-intersection model with partial context achieved an accuracy of 63.5%, precision of 63.5%, recall of 63.4%, and an F1 score of 63.4%. The classification-intersection model with full context achieved an accuracy of 64.8%, precision of 64.7%, recall of 65.0%, and an F1 score of 64.8%. By using the intersection point, the classification models achieved a balanced trade-off between accuracy, precision, and recall. Although we observed some noise while training and the mean token accuracy did not reach the levels observed in the generative model (Figure 4), the final results were promising. The classification model demonstrated its effectiveness, with improved performance metrics and a balanced trade-off between precision and recall. All of the inferences were done at $temperature = 0$, as this setting ensures deterministic predictions, eliminating randomness and allowing us to consistently evaluate the model's performance.

**Tissue-Specific Analysis:** In our expanded analysis, we focused on examining RNA editing patterns across six distinct tissues, each selected for its unique expression profile of ADAR enzymes, which are known to influence A-to-I RNA editing. The tissues included Muscle (Skeletal), Whole Blood, Artery (Tibial), Brain (Cerebellum), Esophagus (Muscularis), and Brain (Spinal Cord). For example, Muscle (Skeletal) is characterized by a low level of ADAR1 expression, while Whole Blood exhibits a high level of ADAR1 and a low level of ADAR2. Artery (Tibial) is notable for its high level of ADAR2 expression. The Esophagus (Muscularis) features expression of ADAR1 and ADAR2, and the Brain (Cerebellum and Spinal Cord) is marked by a high expression of ADAR3.

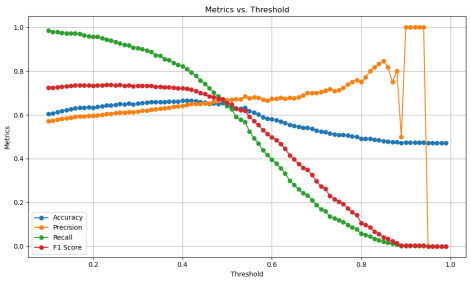
The motivation for this tissue-specific analysis stems from the understanding that RNA editing can vary significantly between tissues, influenced by differences in ADAR expression and the presence of other RNA-binding proteins that can modulate ADAR activity. By focusing on these six tissues, where ADAR expression profiles deviate notably from those observed across the broader GTEx



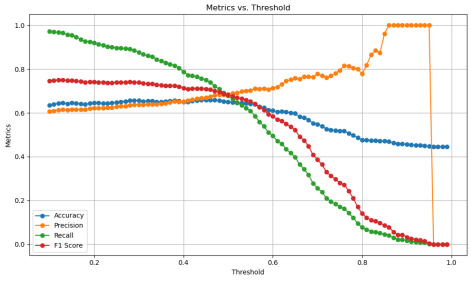

(a) Threshold adjustment for detecting RNA editing sites using fine-tuned GPT models. The figures illustrate the effect of threshold adjustment on the performance metrics (accuracy, precision, recall, and F1 score). The left figure shows the metrics for the partial context, with the optimal F1 score around a threshold of 0.29 and the precision-recall intersection at 0.4. The right plot depicts the full context, where the optimal F1 score remains around 0.29, but the precision-recall intersection shifts to approximately 0.5. These adjustments highlight the impact of context on optimizing the balance between precision and recall.



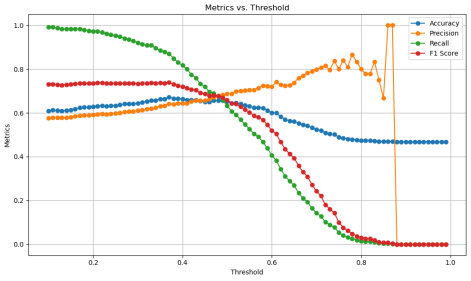

(b) Artery Tibial.



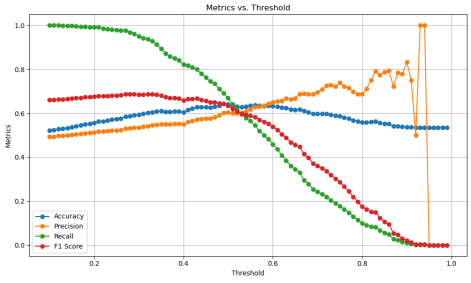

(c) Brain Cerebellum.



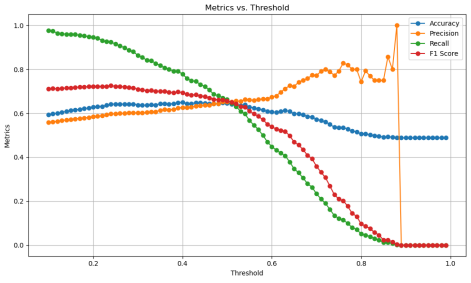

(d) Brain Spinal Cord.




(e) Esophagus Muscularis.




(f) Muscle Skeletal.




(g) Whole Blood.

Figure 5: Threshold adjustments for detecting RNA editing sites using fine-tuned GPT models across the entire data (Figure 5a) and for different tissues, each with its own fine-tuned model (Figure 5b–Figure 5g). Each subfigure shows the effect of threshold adjustment on performance metrics (accuracy, precision, recall, and F1 score) for a specific tissue. It is clear that the specific fine-tuned models for each tissue follow the performances of the general model.

dataset, we aimed to determine whether these variations would result in distinct editing patterns. For each tissue, we classified whether adenosines were edited or not, based on tissue-specific information, rather than using a unified dataset as done in earlier analyses. This approach allowed us to explore how well our model could learn the specific editing patterns associated with each tissue.

The results, illustrated in Figure 5b–Figure 5g, show that while there are minor differences in initial learning across tissues – with some tissues exhibiting a higher or lower starting point – all models ultimately converged to a similar level of learning.

## 5 Limitations and Future Work

Despite the promising results of using GPT models to predict RNA editing sites, several limitations must be acknowledged. First, the demonstration here is still on a relatively small dataset, focusing primarily on editing sites within *Alu* elements and not including coding regions, which may limit the applicability of our findings to other genomic contexts. Second, the editing percentage threshold, which we chose to be 15%, ensures that most of the sites are likely correct but may not capture the complete systematicity of RNA editing. Additionally, while data augmentation techniques helped mitigate some data scarcity issues, they may introduce artificial patterns not present in real biological data, potentially skewing the results. Lastly, the current approach's dependency on existing RNA secondary structure prediction tools, which themselves have limitations, may propagate errors into the final predictions. Further improvements in data quality, augmentation techniques, and integration with more accurate RNA structure prediction methods are essential for enhancing the robustness and applicability of this approach.

In addition to refining our current approaches, several promising avenues from the AI prespective can be explored to further enhance the prediction of RNA editing sites. Firstly, we can leverage Retrieval-Augmented Generation (RAG) [20] methods by framing the data as Question-Answer (Q-A) sets. This would enable the model to identify broad patterns or trends associated with 'Yes' (adenosine edited) and 'No' (adenosine not edited) predictions. Such insights could help derive new explainable rules, which can be incorporated into a rule-based system, enhancing both accuracy and interpretability. Secondly, employing the chain-of-thoughts [39] methodology could be instrumental. We can initiate by asking the model to identify broad trends and then progressively refine our questions to extract specific rules and potentially generate code snippets. By providing the model with current known rules and asking for new ones through a chain-of-thoughts approach, we can systematically build upon existing knowledge. Thirdly, given the Q-A format of the data, few-shot learning [7] can be utilized. By presenting the model with a subset of training data, we can prompt it to generate new examples from the validation set and predict their 'Yes' or 'No' outcomes. This method not only aids in uncovering additional patterns but also enhances the model's explainability, offering clearer insights into the decision-making process. Finally, we can explore the potential of other models beyond OpenAI's GPT, such as Gemini [36] or LLaMA [11], to see if they offer unique advantages or improvements in predictive performance and interpretability.

## Acknowledgments

This work was supported by US National Institutes of Health award R35GM122543 (M.L.), National Natural Science Foundation of China grant No. 31770776 (F.Z.), Foundation Fighting Blindness (grants TA-GT-0620-0790-HUJ and PPA-0923-0865-HUJ), grants from the Israeli Ministry of Science (grant 3-17916) and by Israel Science Foundation (2637/2). Michael Levitt is the Robert W. and Vivian K. Cahill Professor of Cancer Research.

## References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] Md TA Azad, S Bhakta, and T Tsukahara. Site-directed rna editing by adenosine deaminase acting on rna for correction of the genetic code in gene therapy. *Gene therapy*, 24(12):779–786, 2017.

[3] Brenda L Bass. Rna editing by adenosine deaminases that act on rna. *Annual review of biochemistry*, 71(1):817–846, 2002.

[4] Lily Bazak, Ami Haviv, Michal Barak, Jasmine Jacob-Hirsch, Patricia Deng, Rui Zhang, Farren J Isaacs, Gideon Rechavi, Jin Billy Li, Eli Eisenberg, et al. A-to-i rna editing occurs at over a hundred million genomic sites, located in a majority of human genes. *Genome research*, 24(3):365–376, 2014.

[5] Lily Bazak, Erez Y Levanon, and Eli Eisenberg. Genome-wide analysis of alu editability. *Nucleic acids research*, 42(11):6876–6884, 2014.

[6] Axel Brennicke, Anita Marchfelder, and Stefan Binder. Rna editing. *FEMS microbiology reviews*, 23(3):297–316, 1999.

[7] Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[8] Ruyi Chen, Fuyi Li, Xudong Guo, Yue Bi, Chen Li, Shirui Pan, Lachlan JM Coin, and Jiangning Song. Attic is an integrated approach for predicting a-to-i rna editing sites in three species. *Briefings in Bioinformatics*, 24(3):bbad170, 2023.

[9] Meredith Corley, Margaret C Burns, and Gene W Yeo. How rna-binding proteins interact with rna: molecules and mechanisms. *Molecular cell*, 78(1):9–29, 2020.

[10] Prescott Deininger. Alu elements: know the sines. *Genome biology*, 12:1–12, 2011.

[11] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[12] Julie M Eggington, Tom Greene, and Brenda L Bass. Predicting sites of adar editing in double-stranded rna. *Nature communications*, 2(1):319, 2011.

[13] Eli Eisenberg and Erez Y Levanon. A-to-i rna editing—immune protector and transcriptome diversifier. *Nature Reviews Genetics*, 19(8):473–490, 2018.

[14] Masatora Fukuda, Hiromitsu Umeno, Kanako Nose, Azusa Nishitarumizu, Ryoma Noguchi, and Hiroyuki Nakagawa. Construction of a guide-rna for site-directed rna mutagenesis utilising intracellular a-to-i rna editing. *Scientific reports*, 7(1):1–13, 2017.

[15] Orshay Gabay, Yoav Shoshan, Eli Kopel, Udi Ben-Zvi, Tomer D Mann, Noam Bressler, Roni Cohen-Fultheim, Amos A Schaffer, Shalom Hillel Roth, Ziv Tzur, et al. Landscape of adenosine-to-inosine rna recoding across human tissues. *Nature Communications*, 13(1):1184, 2022.

[16] Mahsa Ghanbari and Uwe Ohler. Deep neural networks for interpreting rna-binding protein target preferences. *Genome research*, 30(2):214–226, 2020.

[17] Jonatha M Gott and Ronald B Emeson. Functions and mechanisms of rna editing. *Annual review of genetics*, 34(1):499–531, 2000.

[18] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23, 2021.

[19] Kevin Maik Jablonka, Philippe Schwaller, Andres Ortega-Guerrero, and Berend Smit. Leveraging large language models for predictive chemistry. *Nature Machine Intelligence*, 6(2):161–169, 2024.

[20] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

[21] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt understands, too. *AI Open*, 2023.

[22] John Lonsdale, Jeffrey Thomas, Mike Salvatore, Rebecca Phillips, Edmund Lo, Saboor Shad, Richard Hasz, Gary Walters, Fernando Garcia, Nancy Young, et al. The genotype-tissue expression (gtex) project. *Nature genetics*, 45(6):580–585, 2013.

[23] Ronny Lorenz, Stephan H Bernhart, Christian Höner zu Siederdissen, Hakim Tafer, Christoph Flamm, Peter F Stadler, and Ivo L Hofacker. Viennarna package 2.0. *Algorithms for molecular biology*, 6:1–14, 2011.

[24] Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. Biogpt: generative pre-trained transformer for biomedical text generation and mining. *Briefings in bioinformatics*, 23(6):bbac409, 2022.

[25] Melissa M Matthews, Justin M Thomas, Yuxuan Zheng, Kiet Tran, Kelly J Phelps, Anna I Scott, Jocelyn Havel, Andrew J Fisher, and Peter A Beal. Structures of human adar2 bound to dsrna reveal base-flipping mechanism and basis for site selectivity. *Nature structural & molecular biology*, 23(5):426–433, 2016.

[26] Tobias Merkle, Sarah Merz, Philipp Reautschnig, Andreas Blaha, Qin Li, Paul Vogel, Jacqueline Wettengel, Jin Billy Li, and Thorsten Stafforst. Precise rna editing by recruiting endogenous adars with antisense oligonucleotides. *Nature biotechnology*, 37(2):133–138, 2019.

[27] Hagit T Porath, Shai Carmi, and Erez Y Levanon. A genome-wide map of hyper-edited rna reveals numerous new sites. *Nature communications*, 5(1):4726, 2014.

[28] Elizabeth M Porto, Alexis C Komor, Ian M Slaymaker, and Gene W Yeo. Base editing: advances and therapeutic opportunities. *Nature Reviews Drug Discovery*, 19(12):839–859, 2020.

[29] Gokul Ramaswami and Jin Billy Li. Radar: a rigorously annotated database of a-to-i rna editing. *Nucleic acids research*, 42(D1):D109–D113, 2014.

[30] Philipp Reautschnig, Carolin Fruhner, Nicolai Wahn, Charlotte P Wiegand, Sabrina Kragness, John F Yung, Daniel T Hofacker, Jenna Fisk, Michelle Eidelman, Nils Waffenschmidt, et al. Precise in vivo rna base editing with a wobble-enhanced circular cluster guide rna. *Nature Biotechnology*, pages 1–13, 2024.

[31] Jessica S Reuter and David H Mathews. Rnastructure: software for rna secondary structure prediction and analysis. *BMC bioinformatics*, 11:1–9, 2010.

[32] Shalom Hillel Roth, Erez Y Levanon, and Eli Eisenberg. Genome-wide quantification of adar adenosine-to-inosine rna editing activity. *Nature methods*, 16(11):1131–1138, 2019.

[33] Thomas Schwarzl, Sudeep Sahadevan, Benjamin Lang, Milad Miladi, Rolf Backofen, Wolfgang Huber, Matthias W Hentze, and Gian Gaetano Tartaglia. Improved discovery of rna-binding protein binding sites in eclip data using dewseq. *Nucleic Acids Research*, 52(1):e1–e1, 2024.

[34] Huseyin Avni Tac, Mustafa Koroglu, and Ugur Sezerman. Rddsvm: accurate prediction of a-to-i rna editing sites from sequence using support vector machines. *Functional & Integrative Genomics*, 21(5):633–643, 2021.

[35] Meng How Tan. Identification of bona fide rna editing sites: History, challenges, and opportunities. *Accounts of Chemical Research*, 56(21):3033–3044, 2023.

[36] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[37] Paul Vogel and Thorsten Stafforst. Critical review on engineering deaminases for site-directed rna editing. *Current opinion in biotechnology*, 55:74–80, 2019.

[38] Chong Wang, Mengyao Li, Junjun He, Zhongruo Wang, Erfan Darzi, Zan Chen, Jin Ye, Tianbin Li, Yanzhou Su, Jing Ke, et al. A survey for large language models in biomedicine. *arXiv preprint arXiv:2409.00133*, 2024.

[39] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[40] Keisuke Yamada and Michiaki Hamada. Prediction of rna–protein interactions using a nucleotide language model. *Bioinformatics Advances*, 2(1):vbac023, 2022.

[41] Fan Yang, Wenchuan Wang, Fang Wang, Yuan Fang, Duyu Tang, Junzhou Huang, Hui Lu, and Jianhua Yao. scbert as a large-scale pretrained deep language model for cell type annotation of single-cell rna-seq data. *Nature Machine Intelligence*, 4(10):852–866, 2022.

[42] Marlon S Zambrano-Mila, Monika Witzenberger, Zohar Rosenwasser, Anna Uzonyi, Ronit Nir, Shay Ben-Aroya, Erez Y Levanon, and Schraga Schwartz. Dissecting the basis for differential substrate specificity of adar1 and adar2. *Nature Communications*, 14(1):8212, 2023.

[43] Zhihan Zhou, Weimin Wu, Harrison Ho, Jiayi Wang, Lizhen Shi, Ramana V Davuluri, Zhong Wang, and Han Liu. Dnabert-s: Learning species-aware dna embedding with genome foundation models. *ArXiv*, 2024.