Contextual Approaches to Differentiating Human and Machine-generated Texts

Adam Vinestock, Omer Wachman

Mentors: Dr. Alon Kipnis

Submitted as final project report for the NLP course, IDC, 2023

1 Introduction

In recent years, there has been significant advancement in the domain of generative artificial intelligence, particularly in natural language. Models like ChatGPT have risen to prominence, blurring the lines between human-generated and machine-generated content. As these models become more sophisticated, distinguishing between human-written texts and those produced by machines—or even human-edited machine texts—has become increasingly challenging. This project seeks to further the ongoing efforts to better distinguish between human and machine-generated content.

Various methodologies and strategies abound in this domain, each offering a unique perspective and solution to the problem. In this project we collaborated with Dr. Alon Kipnis regarding his research in the field of information theory addressing authorship attribution, particularly a sub-problem within the domain of detecting human edits in machine-generated text.
In order to understand our project, it is necessary to understand his research's general outline:

1. Inputs: GLM (Generating language model) Distribution of log-ppx sentences, LM by which we evaluate, and an examination document.
2. Evaluate the document's log-ppx (aka log-loss) wrt to the LM
3. Test the individual hypothesis per sentence
   - $H_{0,i}$: S_i is written by the GLM
   - Denote by p_i the corresponding P-value
4. Run global test by applying HC to p-values
5. Use HC to test H_0 : "all sentences were written by the GLM"
6. Identify suspected sentences using the HC threshold.

The current methodology demonstrates challenges in two aspects:
Calibrating the distribution of log-ppx of GLM sentences from different domains and reducing the averaged log-ppx that will in turn increase the difference between GLM sentences to edited sentences, thus resulting in better detection.

Our project will address the second challenge. Within this framework, we aim to map contextually-induced policies that can optimally differentiate the perplexity distributions of text originating from humans and machine, thereby leading to a more proficient detector. We hypothesize that by providing context to the LM not only will the overall log-ppx values be reduced, but also the difference between the machine vs human log-ppx histograms will increase. This will in turn effect calculating the p-values improving the efficacy of the global test. Our exploration encompasses diverse strategies to embed contextual information, each introducing a unique dimension of induced information:

1. **N-gram Approach:** Here, the context is derived from the preceding n sentences. This method adopts a somewhat Bayesian perspective — given our prior observations, what is the likelihood of the following sentence?

2. **Recurrent Attention-Based Context:** This approach employs a dynamically updated hidden state that summarizes the text observed up to the previous sentence. With each new sentence, the hidden state gets refined, encapsulating a condensed representation of all previously seen text. Due to the heavy computation required, we did not test this policy.

3. **Single Hidden State Context:** In this approach, a single hidden state is encoded for the entire text sample, acting as a condensed representation. This policy includes introducing information influenced by future parts of the tested sample.

4. **Question Answering:** This policy involves generating a question from the observed sentence. It explores stressing the stylistic elements inherent from the author.

5. **Naïve approach:** Entailment of a fixed context "The following text was written by a Large Language Model:". While seemingly rudimentary, this policy should also be explored as the LM might exhibit pronounced responses to distinctive human edits.

2 Solution

2.1 General approach

Our central approach revolved around identifying the policy that differentiates the human and machine responses (log-ppx histograms) the greatest. We used the "No Context" policy as our baseline reference against which we would compare our results. The differentiation will be evaluated by three different methods:

1. Visually evaluating the plotted histograms of machine and human responses

2. Calculating and comparing the standardized mean difference between the human and machine response values

3. Comparing the AUC (area under the ROC curve)

The context policies we chose to examine:

- **No context:** The sentence under examination is presented without any additional context.
- **Previous sentence:** The sentence prior to the one under examination is provided as context[1].
- **Previous 3 sentences:** The last three sentences prior to the examined sentence are provided as context.
- **Naïve:** A fixed sentence - "The following text was generated by a Large Language Model:" is provided as a context for every examined sentence.
- **Summary and previous sentence:** Both the summary of the entire tested sample and the sentence prior to the examined one are provided as context. Summaries are crafted for each sample using an existing summarization BART model from hugging face[2].
- **QA (Questions Answering):** For each examined sentence, a contextually generated question was provided as context, using a T5 pretrained question generating model from hugging face[3].

To comprehensively address our research problem, we used distinct datasets from different domains. Our rationale for this was based on the understanding that while a detector might perform well in a specific domain, it might not be as effective in others. Consequently, we ran the test on three separate datasets originating from Wikipedia, news articles and research papers. Each of them contain both original human-written text and machine-generated versions, which are produced using a prompt of the initial 7-9 words from the human text along with a general topic descriptor.

1. "alonkipnis/wiki-intro-long"
2. "alonkipnis/news-chatgpt-long"
3. "NicolaiSivesind/ChatGPT-Research-Abstracts"

2.2 Design

The project required incorporating pre-existing code from Dr. Alon Kipnis with our novel introduced methodologies for parsing the examined text, creating the context per-sentence and analyzing the results. This lead us to create a dedicated GitHub repository. Given that our "sensor" leveraged a resource-intensive LLM, and some context policies required deploying an additional transformer based model over each sentence, it was imperative to run inferences in a cloud environment. Initially, our experiments revolved around the GPT2 model, primarily because it was compatible with the complimentary version of Google Colab. After upgrading to Colab Pro, we had the opportunity to work with larger models like Falcon and Llama 2 (7b). However, once we gained access to a Jupiterlab environment running a virtual machine instance equipped with an A100 GPU, which was the most robust hardware we could utilize, we opted for the GPT2-XL model. This decision was driven by our ambition to reach the desired number of processed samples. Each VM instance was available for a time frame of 4 hours, this constraint made the data collection phase quite labor-intensive.

The project outline ran over the following variables:

[1] In edge cases such as the first sentence, no context is given.

[2] BART is a transformer encoder-decoder (seq2seq) model with a bidirectional (BERT-like) encoder and an autoregressive (GPT-like) decoder.

[3] iarfmoose/t5-base-question-generator is a sequence-to-sequence question generator which takes an answer and context as an input, and generates a question as an output. It is based on a pretrained t5-base model.

- Dataset (3 variants)
- Author: Human or Machine (2 variants)
- Context Policy (6 variants)

For each combination of dataset, author, and context policy, we produced a CSV file containing the processed data including the log-ppx of each sentence and its corresponding length. We then used these CSV files to extract and analyze the results. As mentioned, we firstly evaluated the difference visually by observing the plotted log-ppx histograms. Then, we calculated the standardized mean difference between the human and machine response histograms normalized by the pooled standard deviation, which is defined as:

$$pooledSTD = \sqrt{\frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}}$$

The standardized mean difference on its own is not always an accurate indicator. When log-ppx histograms significantly diverge from a normal distribution, the AUC metric applies as more reliable. The AUC metric, defined as the area under the ROC (Receiver operating characteristic) curve, gauges the true positive rate in relation to the false positive rate. In this context, false positives (FP) are machine-generated sentences with a log-ppx value surpassing the set threshold, while true positives (TP) are human-generated sentences exceeding the same threshold. This is because our primary objective is to pinpoint human edits.
To ensure a comprehensive analysis, our goal was to obtain approximately 5000 samples for each context policy over every author. Given the lengthy processing time and the limited 4-hour window access to the VM, this was indeed a challenging task. For each context policy, we processed 1500 samples across each dataset and author type, reaching in total 4500 samples. The study involved discriminating 2 authors (GPT2 and human) with 6 context policies, amounting to 54,000 evaluations of samples, each with an average of 25 sentences. Total run time was approximately 24 hours.

3 Experimental results

	Human mean response	Machine mean response	Human-Machine difference	Diff from baseline	AUC
No Context - Baseline	3.58052	2.9372	0.730682	0	0.725778
Previous Sentence	3.26394	2.37983	0.926891	+ 0.19620936536634925 ↑	0.796196
Previous 3 Sentences	3.04945	2.0539	1.09394	+ 0.3632603987907569 ↑	0.837127
Naive	4.17584	3.4623	0.667109	- 0.0635728448194951 ↓	0.719042
Summary and Previous	2.52894	1.72335	0.726554	- 0.004128163697320719 ↓	0.720989
QA	2.39212	1.70759	0.588459	- 0.1422230585681442 ↓	0.681827

Figure 1: Wiki-intro-long Dataset Results

	Human mean response	Machine mean response	Human-Machine difference	Diff from baseline	AUC
No Context - Baseline	3.88168	2.80222	1.04561	0	0.795574
Previous Sentence	3.79161	2.55382	1.05402	+ 0.008413118852742318 ↑	0.816463
Previous 3 Sentences	3.4234	2.30793	1.01067	- 0.034936762405152866 ↓	0.81249
Naive	4.84367	3.49723	1.11171	+ 0.06610623990061426 ↑	0.807584
Summary and Previous	3.31438	2.03191	0.955955	- 0.08965375595659297 ↓	0.797017
QA	3.01703	2.05662	0.710382	- 0.33522633126677326 ↓	0.673742

Figure 2: News-chatgpt-long Dataset Results

	Human mean response	Machine mean response	Human-Machine difference	Diff from baseline	AUC
No Context - Baseline	3.98493	3.32187	0.833513	0	0.733919
Previous Sentence	3.47173	2.72223	0.892833	+ 0.05932050100717512 ↑	0.755693
Previous 3 Sentences	3.1966	2.42109	0.934712	+ 0.10119966860264717 ↑	0.766955
Naive	4.2007	3.47838	0.821846	- 0.011666191379703372 ↓	0.737197
Summary and Previous	2.83863	2.08135	0.721856	- 0.11165611049160262 ↓	0.712851
QA	2.52567	1.83142	0.562138	- 0.2713746809299171 ↓	0.652809

Figure 3: ChatGPT-Research-Abstracts Dataset Results

Across the Wiki-intro and Research-Abstract datasets, the results appear consistent for all six context policies. The News dataset tends to display distinct outcomes. This variation might be attributed to the less structured nature of text within this dataset.

The naive method increases the overall the log-ppx for both human and machine-generated texts across all datasets. Yet, a slight enhancement in separation is discernible for the News dataset when compared to the baseline.

Previous 3 sentences emerges as the most effective context policy. It exhibits the highest AUC and deviation from the baseline for the Wiki and Research datasets. For the News dataset, even though there's a reduction in the overall difference relative to the baseline, it still maintains the second-highest AUC, surpassed only by the "Previous Sentence" policy.

Both the Summarization and QA approaches considerably lower the overall log-ppx for both authors. This could be attributed to introducing information from forthcoming segments of the sentence or text sample. However, they don't significantly contribute to enhancing the separation between the author response values.

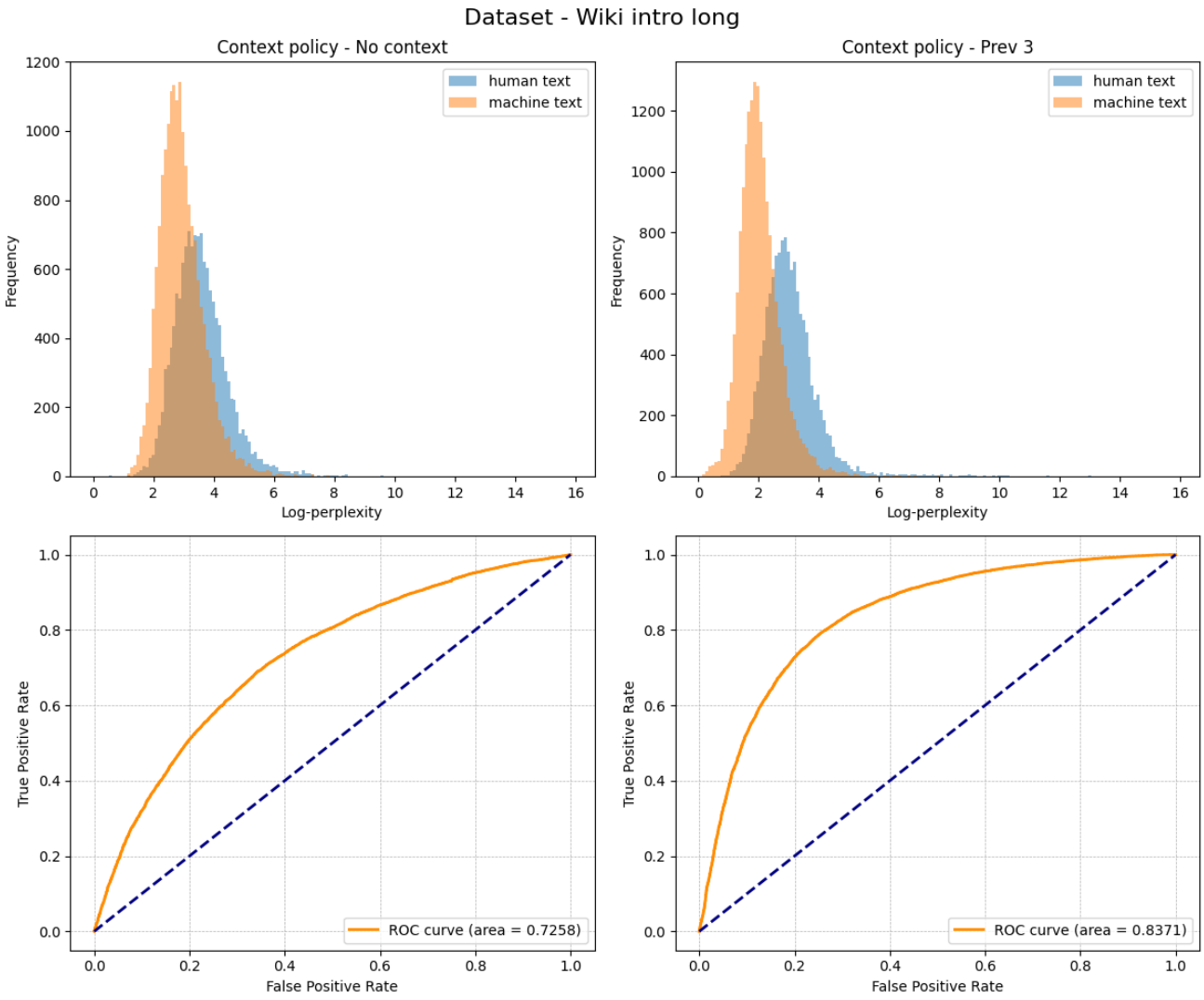

Figure 4: Wiki dataset - Comparing baseline to best performing Context Policy

When analyzing the Wiki dataset and comparing the baseline 'No context' policy to the optimal 'Previous 3 Sentences' policy (refer to Figure 4), we can observe a clear increase in response distributions separation.

When analyzing the best-performing context policy across varying sentence lengths (Figures 5, 6, and 7 in appendix), we observe that the optimal separation for the wiki data occurs with sentences longer than 40 tokens, for the news data with sentences shorter than 20 tokens, and for research data in the mid-range of 20 to 39 tokens. Contrary to our initial hypothesis, neither extreme (very long or very short sentences) consistently outperforms the other. This suggests that a broad spectrum of sentence lengths could be valuable for discrimination."

In Figures 8 to 13, presented in the appendix, we examine the characteristics of the log-ppx distributions across various domains. Notably, the summary exhibits a marked elevation in low-ppx responses for both authors. This phenomenon might be attributed to sentences that convey core information or those sentences in the summary that are directly extracted from the sample. Additionally, the QA approach exhibits the greatest variance across all domains.
In the news dataset, the human response distribution displays certain irregularities - A pronounced tail is evident across all context policies. There are noticeable spikes in high perplexity values, with the exception of the 'summary and prev' policy. Such spikes could potentially be attributed to a recurring phrase present in the human text.

4 Discussion

In conclusion, our analysis reveals that the n-gram approach, which leverages "look-back" context, consistently outperforms other policies across all dataset types in enhancing the separation between human and machine-generated texts. While we initially hypothesized that a decrease in overall perplexity would amplify this separation, our findings suggest otherwise. This observation becomes especially pronounced when contrasting the 'summary and prev' policy with the 'previous sentence' policy. While the former reduces overall perplexity, it doesn't match the latter in terms of separation efficacy (see Figure 14 in appendix). Thus, it becomes evident that merely introducing additional contextual information isn't always beneficial for differentiation.

Additional exploration and future steps that could further advance this research:

1. **Model Evaluation:** Assessing different language models, such as Llama2, as the 'sensor'. It would be intriguing to ascertain whether similar characteristics occur across various context policies and sentence lengths when using a different LM.

2. **Anomaly Analysis:** A deeper dive into the anomalous human responses observed in the news dataset, particularly focusing on both the extended tail and the spikes in high perplexity values.

3. **Full Pipeline Evaluation:** Running Dr. Alon Kipnis' complete pipeline on a test set using our proposed context policies can shed light on any potential improvements in detection performance.

5 Code

Links to the project's code:
GitHub Repository - Includes the entire projects code and csv results.
Google Colab Notebook - used to run in the jupyterlab environment for processing the samples. This notebook can also be run post-hoc for analyzing the results by reading the csv files from the repository.

Appendix

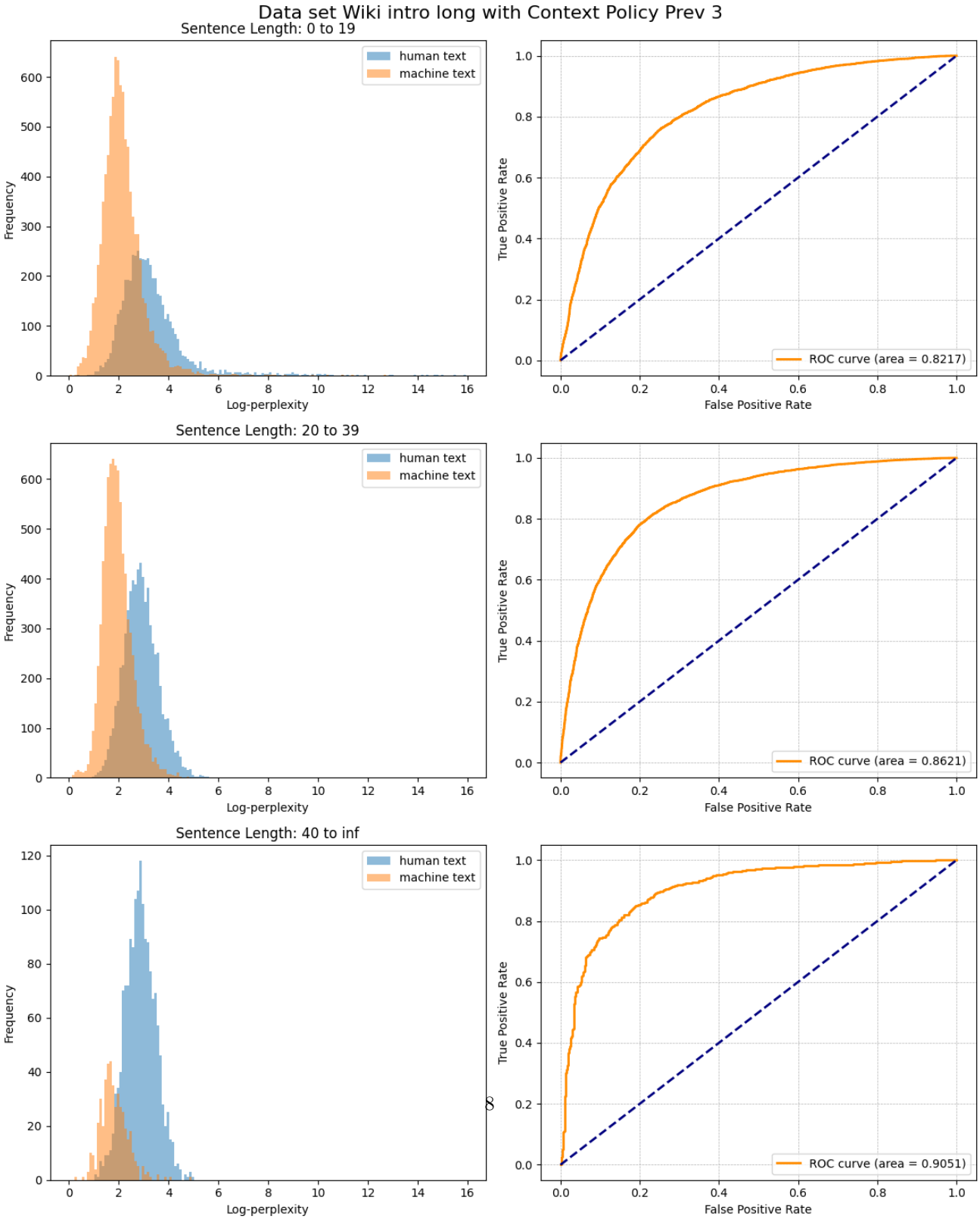

Figure 5: Wiki-intro-long Sentence lengths

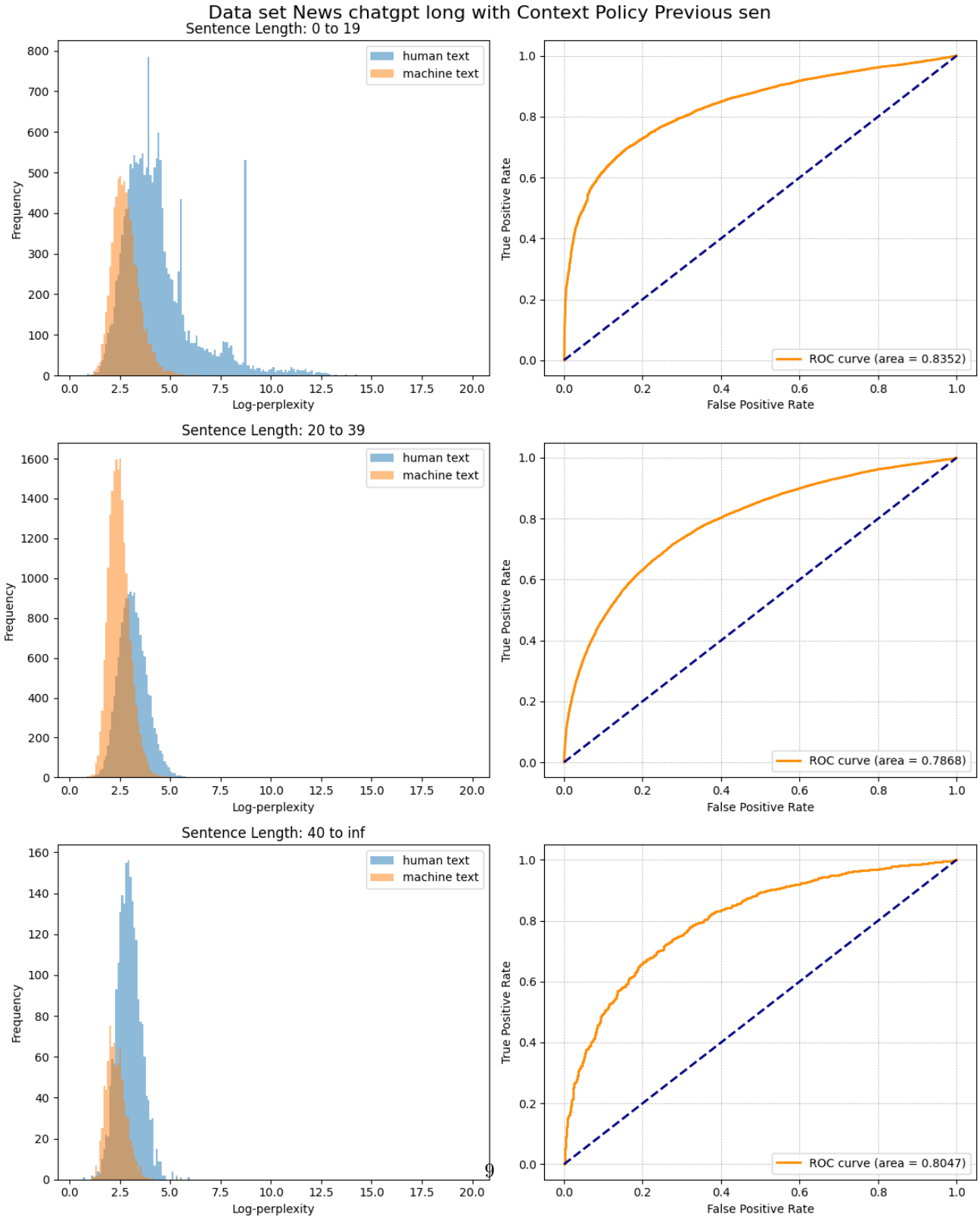

Figure 6: News-chatgpt-long Sentence lengths

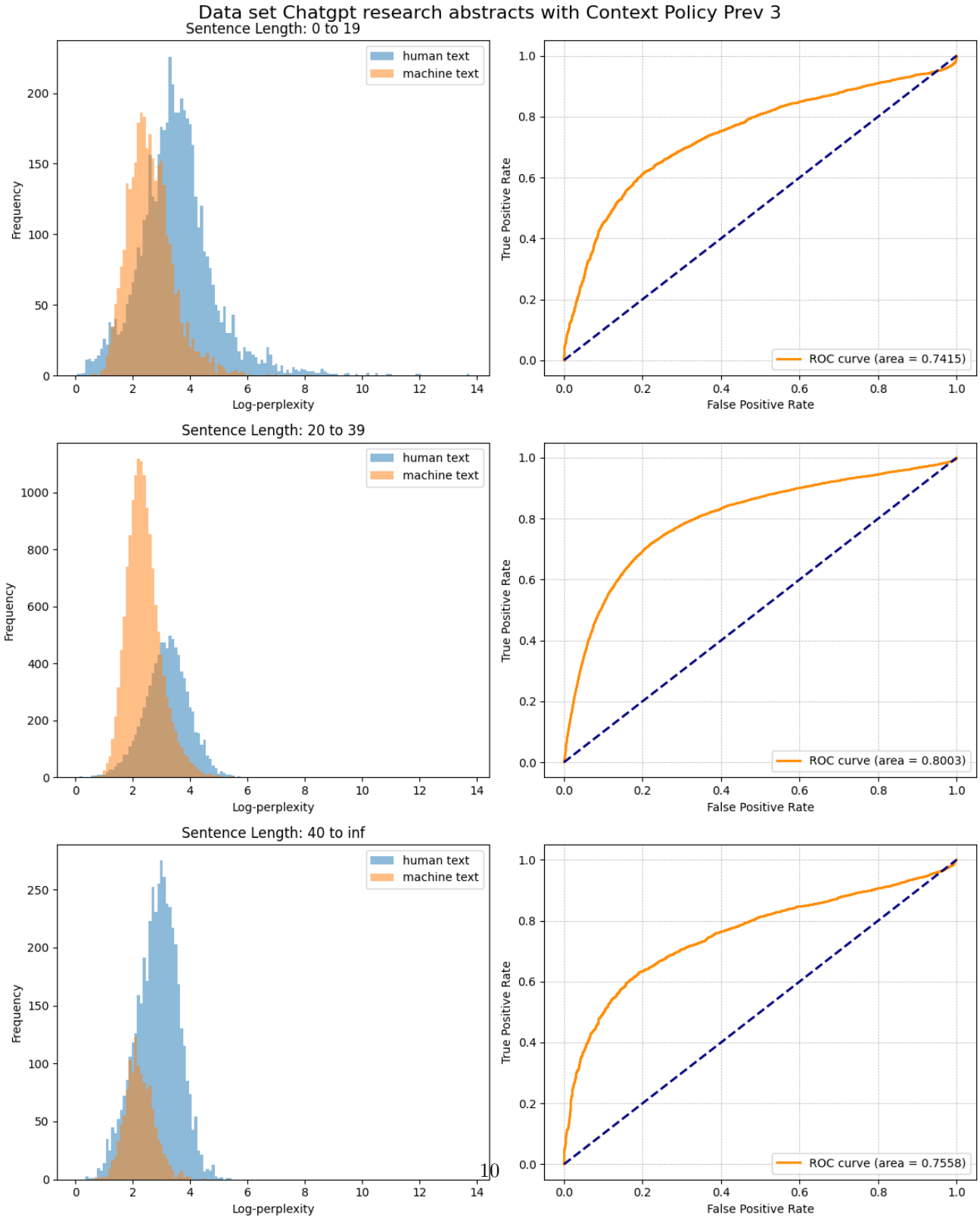

Figure 7: ChatGPT-Research-Abstracts Sentence lengths

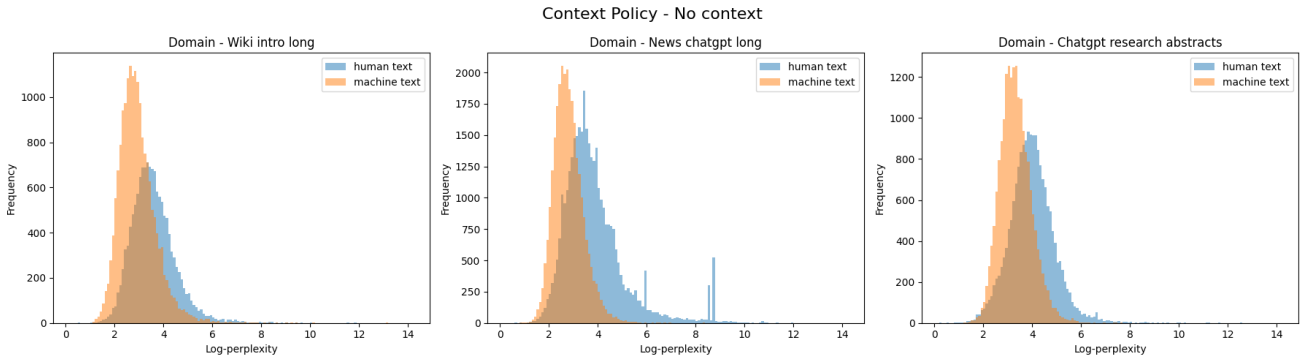

Figure 8: No Context performance

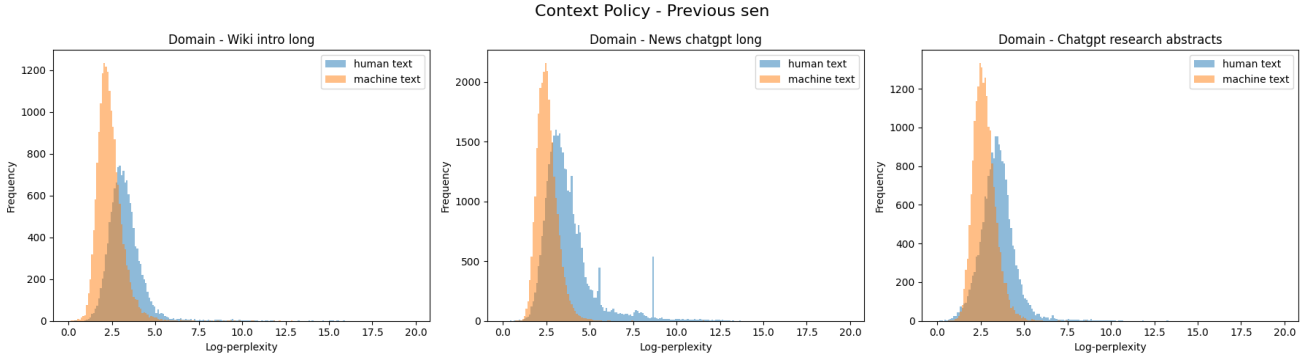

Figure 9: Previous Sentence performance

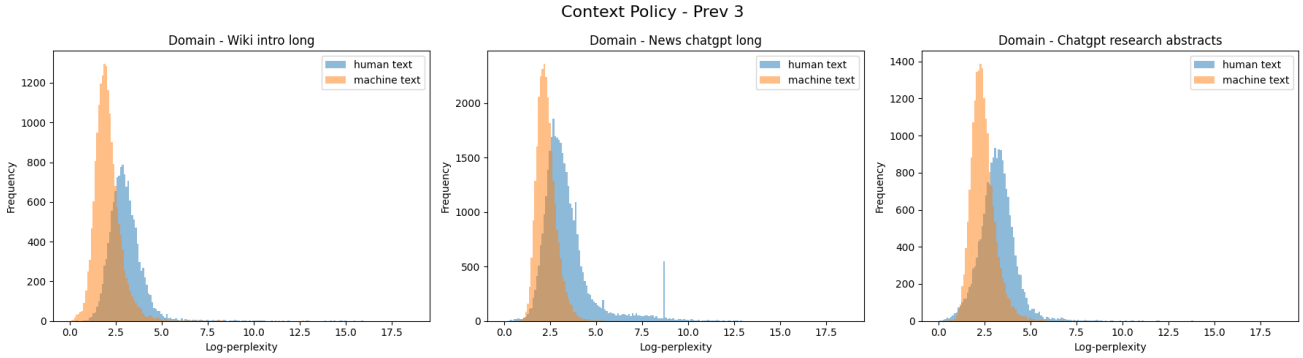

Figure 10: Previous 3 Sentences performance

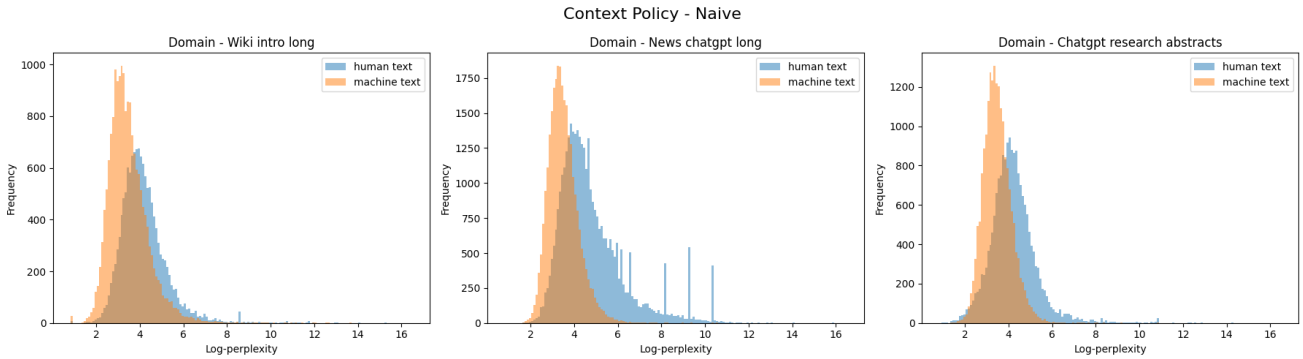

Figure 11: Naive performance

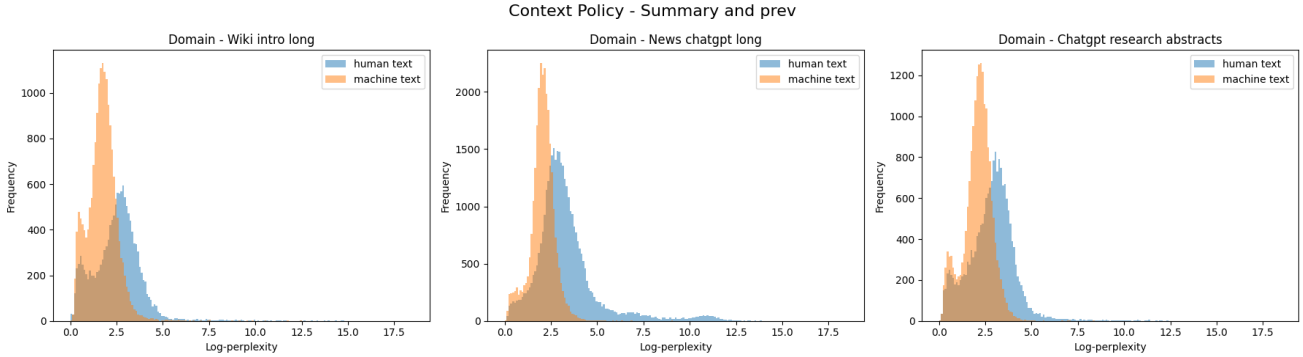

Figure 12: Sum and Prev performance

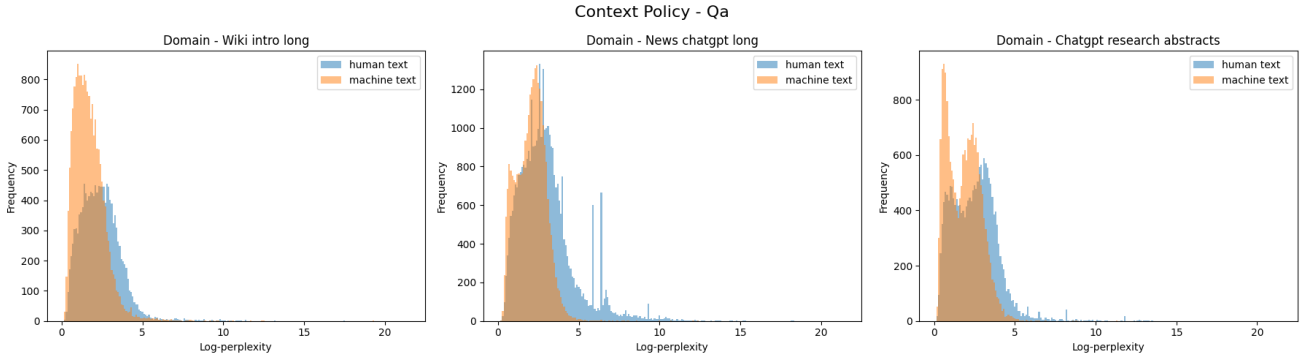

Figure 13: QA performance

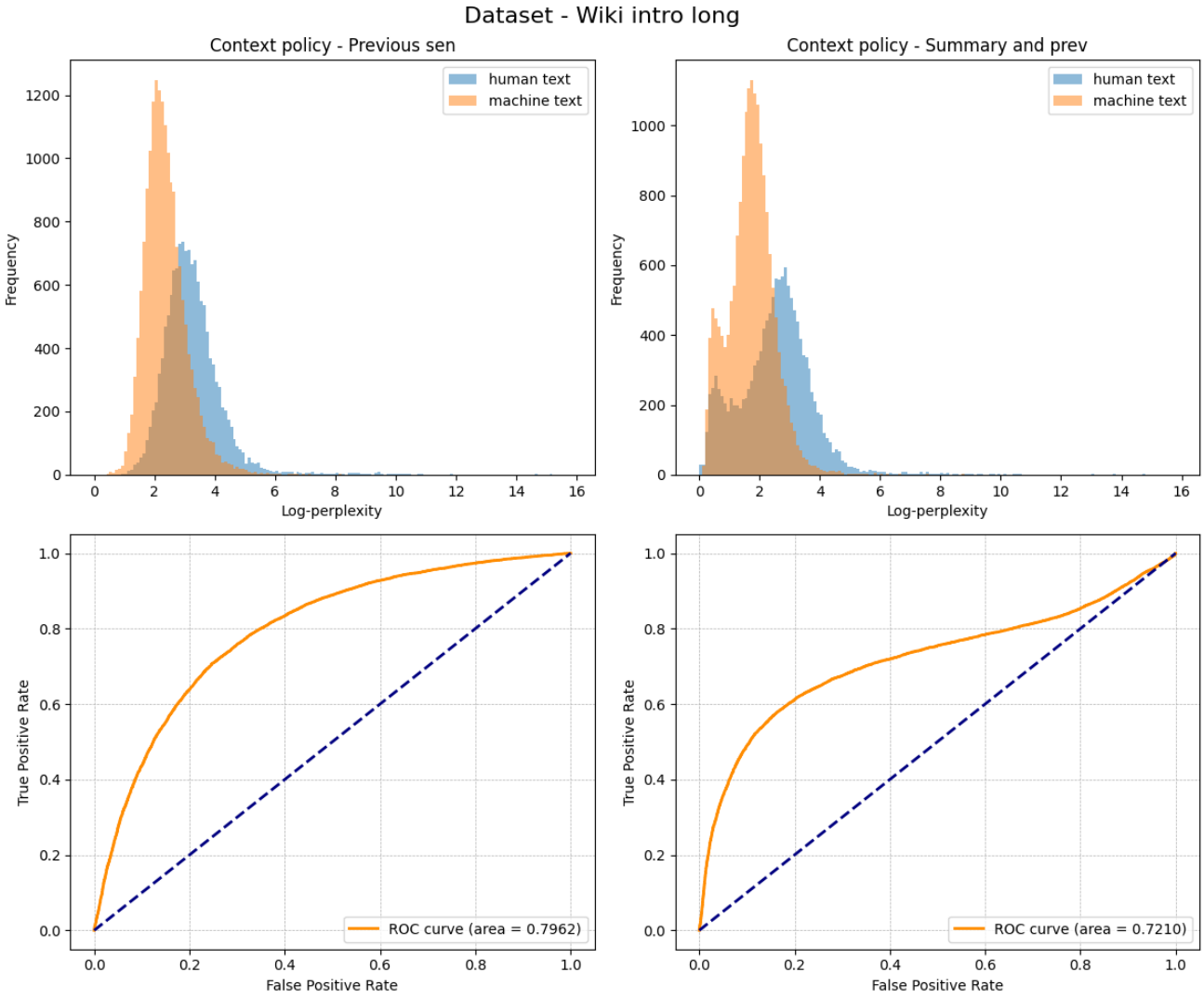

Figure 14: Prev vs Sum and prev