AI21 SUMMARIZE API: TECHNICAL EVALUATION

AI21 Labs
studio@ai21.com

April 2023

ABSTRACT

This study examines the performance of AI21 Summarize API, powered by a task-specific summarization model, and compares it to general-purpose Large Language Models (LLMs), specifically davinci-003 and gpt-3.5-turbo available via OpenAI API. We apply both human evaluation and automatic metrics to evaluate the quality of summaries generated by different models. We use source texts from an established academic benchmark (XSum) as well as proprietary real-world data. Sensitivity of different prompting methods for LLMs is also investigated. We find that AI21 Summarize API generally performs better or on par with both OpenAI LLMs, across different tests. For real world data, human evaluation shows a preference for AI21 Summarize API over OpenAI LLMs, regardless of prompting method. In particular, AI21 Summarize API exhibits a significantly lower rate of unreliable summaries with incorrect information ("hallucinations") and/or misleading re-arrangement of source facts ("reasoning violations"). AI21 Summarize API also outperforms OpenAI LLMs in terms of automatic metrics on the same data, irrespective of prompting method.

1 Introduction

Recent advances in natural language processing (NLP) have made accurate text summarization possible, offering an effective tool for combating information overload in today's information age.

AI21 Summarize API[1] is provided by AI21 Labs to help developers leverage these advances in their own products and services. It is powered by a task-specific model: a language model specialized to one particular task, in this case targeting summarization. The AI21 Summarize model was trained on a proprietary dataset of real-world documents and corresponding high-quality summaries.

In this report, we compare AI21's task-specific approach with a prompting-based approach leveraging general purpose Large Language Models (LLMs). In particular, we evaluate the industry-leading davinci-003 and gpt-3.5-turbo models by OpenAI.

Section 2 lays out the evaluation process and Section 3 describes the main results. Additional details about the experiments we performed can be found in the appendix.

2 Methods

In this section we describe the methods of evaluation, including the datasets and metrics we used.

2.1 Using the APIs

AI21 Summarize API is designed to address summarization with a simple interface[2]. The request contains a source (input) text and the response contains its summary, generated by the underlying language model.

[1] https://docs.ai21.com/docs/summarize-api

[2] https://docs.ai21.com/reference/summarize-api-ref

Davinci-003 and gpt-3.5-turbo are general-purpose LLMs trained to produce a completion that follows natural language instructions in a given prompt. Applying LLMs to accomplish summarization requires crafting suitable instructions in the prompt, a process known as "prompt engineering". To unsure the fairness of our experiments, we followed OpenAI's own prompt engineering best practices[3]. To estimate the impact of well-designed prompts, we used two kinds of prompts:

- **Simple Prompt**, conveying a high-level instruction to "summarize the text below".
- **Detailed Prompt**, containing explicit instructions to respect constraints on length, style and consistency with facts in the original text.

The full prompts appear in Appendix A.

2.2 Datasets

We tested all models on two kinds of data.

Academic data: We used the XSum dataset (Narayan et al., 2018), an established academic benchmark for abstractive single-document summarization systems.

Real-world data: We compiled a dataset of publicly-available source documents that users submitted for summarization through various AI21 Labs products. This dataset is balanced in terms of length, sources and domains and reflects real-world usage patterns.

2.3 Metrics

2.3.1 Human evaluation

We sampled two similarly distributed datasets of 100 source texts from the real-world data. Two separate human evaluation experiments were conducted using these two sets of sources:

1. Comparing AI21 Summarize API to OpenAI models with a **simple** prompt.
2. Comparing AI21 Summarize API to OpenAI models with a **detailed** prompt.

In both experiments, each input passage was summarized using both methods (AI21 and OpenAI). Each experiment involved two participants, linguistics students who were trained to evaluate summaries. The participants labeled each summary with one of four labels: Good, Okay, Bad and Very Bad. The evaluation process was blind and special care was taken to ensure fairness, eliminate bias and reduce random noise. A full description of our methods can be found in Appendix B.

According to guidelines given to participants, Good and Okay summaries correspond to usable ("passing") results. We therefore define the **pass rate** for a given model as the fraction of Good or Okay summaries in the sample.

2.3.2 Automatic metrics

We sampled a total of 500 source texts from each dataset and fed them through each of the evaluated models, using their respective APIs as described above in Section 2.1. We applied the following automatic metrics to evaluate the results.

Faithfulness: Based on a proprietary classifier developed by AI21 Labs. For a given pair of source and summary text, the classifier provides a binary "faithfulness" label indicating whether the summary is consistent with its source data. The classifier is most sensitive to factual contradictions between the summary and the source text. The mean faithfulness rate presented in results below is the average rate of faithful summaries produced by each model. We report the standard deviation by splitting the data into five random folds and calculating the standard deviation of the mean faithfulness rate in the different folds.

Compression: For a given pair of source and summary texts, we define compression as

$$1 - \frac{\text{len}(\text{summary})}{\text{len}(\text{source})}$$

where $\text{len}(x)$ is the length in characters of text x (e.g. if the summary is $100\times$ shorter than the source, the compression will be 0.99). The mean compression presented in the results below is the average compression of summaries produced by each model. We also report the standard deviation of this distribution.

[3] https://help.openai.com/en/articles/6654000-best-practices-for-prompt-engineering-with-openai-api

3 Results

3.1 Human evaluation

Figure 1 shows the pass rate for both human evaluation experiments, comparing AI21 Summarize API to davinci-003 with Simple and Detailed prompting. AI21 Summarize API outperforms davinci-003 in both experiments.

Table 1 shows the distribution of labels for each model in both experiments. In both experiments, AI21 Summarize API had significantly fewer Very Bad summaries compared to davinci-003 ($p = .05$ and $p = .008$ respectively). Very Bad labels correspond to completely unreliable summaries that contradict the source text or deviate from it substantially (see Appendix B).

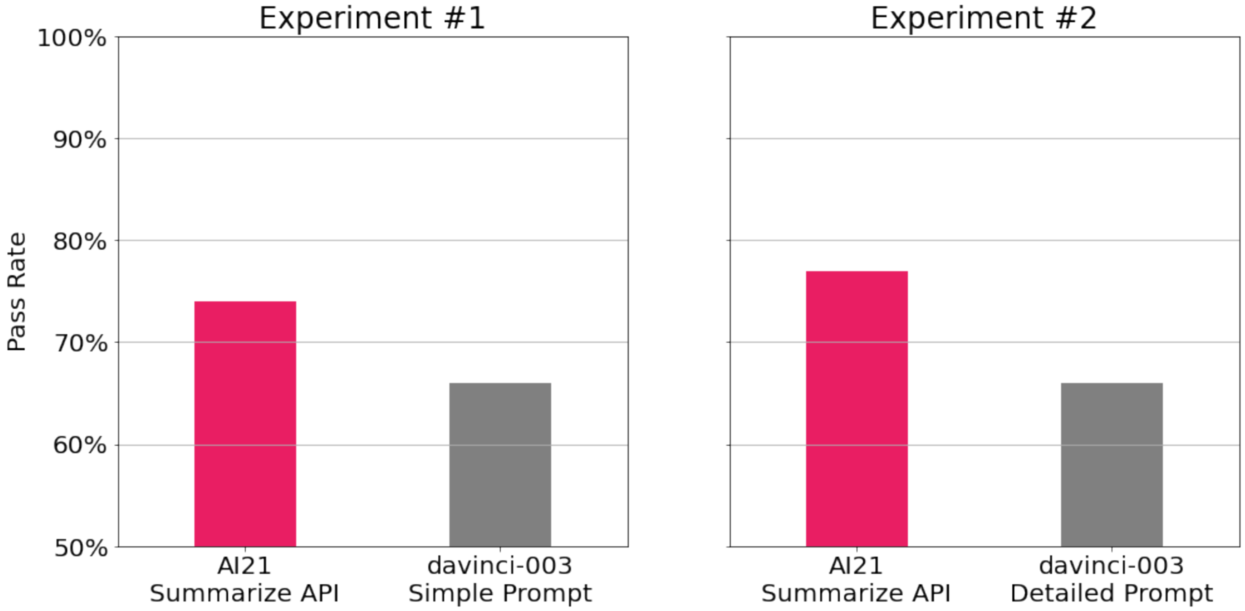

Figure 1: Human evaluation pass rates on real-world data. Left - Experiment #1 comparing AI21 Summarize API versus OpenAI davinci-003 with a simple prompt. Right - Experiment #2 comparing AI21 Summarize API versus OpenAI davinci-003 with a detailed prompt.

Experiment	Model / Method	Good	Okay	Bad	Very Bad
#1	AI21 Summarize API	40.4%	33.3%	22.2%	4%
	davinci-003 Simple Prompt	44.8%	20.4%	17.3%	17.3%
#2	AI21 Summarize API	39.8%	37.8%	10.2%	12.2%
	davinci-003 Detailed Prompt	41.4%	24.2%	8.1%	26.3%

Table 1: Distribution of human evaluation labels on real-world data.

3.2 Automatic metrics

Figure 2 and Table 2 show the compression and faithfulness scores of AI21 Summarize API and OpenAI's two models (davinci-003 and gpt-3.5-turbo), with two prompting methods for each (Simple Prompt and Detailed Prompt). For real world data, AI21 Summarize API outperforms all other models on both compression and faithfulness. On the academic dataset, the AI21 Summarize API scores are comparable with OpenAI's. For both datasets, AI21 Summarize API has the lowest standard deviation on compression, meaning it most reliably produces shorter summaries with predictable length.

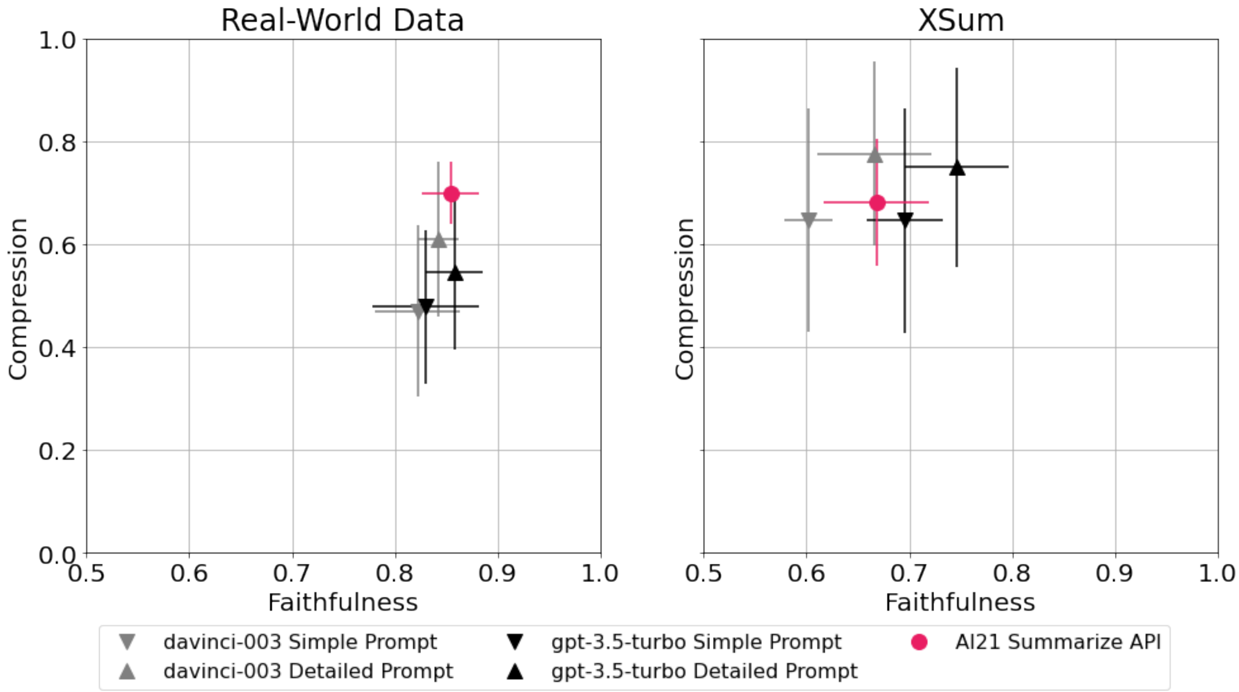

Figure 2: Automatic evaluation (faithfulness and compression scores) for different models. Error bars correspond to standard deviations as defined in Section 2.3.2.

	Real-World Data		XSum	
Model / Method	Faithfulness	Compression	Faithfulness	Compression
davinci-003 Simple Prompt	0.82 ± 0.04	0.47 ± 0.17	0.60 ± 0.02	0.65 ± 0.22
davinci-003 Detailed Prompt	0.84 ± 0.02	0.61 ± 0.15	0.67 ± 0.06	0.78 ± 0.18
gpt-3.5-turbo Simple Prompt	0.83 ± 0.05	0.48 ± 0.15	0.70 ± 0.04	0.65 ± 0.22
gpt-3.5-turbo Detailed Prompt	0.86 ± 0.03	0.55 ± 0.15	0.75 ± 0.05	0.75 ± 0.19
AI21 Summarize API	0.85 ± 0.03	0.70 ± 0.06	0.67 ± 0.05	0.68 ± 0.12

Table 2: Automatic evaluation metrics for different models with corresponding mean and standard deviations on real-world and academic data.

4 Conclusions

In this report, we study the quality of summaries produced by AI21 Summarize API and OpenAI LLMs. Based on human evaluation and automatic metrics on two different kinds of datasets, we find that AI21 Summarize API performs on par or better than LLM based solutions. Crucially, AI21 Summarize API produces significantly fewer summaries rated by humans as Very Bad (meaning they contradict or fundamentally diverge from the source text). Moreover, leveraging prompt engineering to improve the results of OpenAI LLMs achieves only marginal improvement and does not eliminate Very Bad results (Figure 1 and Table 1).

A Evaluation

With OpenAI models, we tested several prompt options and chose the one that performed best. The following prompts were used:

Simple prompt:

> Summarize the text below.
> ##
> Text: [INPUT_TEXT]
> ##
> Summary:

Detailed prompt:

> Summarize the paragraph below. The summary should be up to 2 sentences. Be consistent with the original text, make sure you cover the main points of the original paragraph, and produce a coherent summary.
> ##
> Text: [INPUT_TEXT]
> ##
> Summary:

B Human Evaluation - Detailed

Below is a detailed overview of the human evaluation studies conducted for this report.

B.1 Methods, design and procedure

Participants were second and third year linguistics students working on summarization evaluation projects for the past year. These individuals are highly experienced in categorizing the quality in relation to this specific task, based on the following definitions of quality labels:

- **Good:** A reliable and comprehensive report of the information in the source text. The focus and gist of the source are clearly communicated in the summary. Importantly, the summary does not introduce details not mentioned in the source text ("hallucinations") or distort the actual text's meaning by rearranging pieces of information ("reasoning violations").
- **Okay:** A summary that is somewhat successful in communicating the essence of the source, but only partially reflects it. Typically, this is a result of not mentioning a vital detail or mentioning an insignificant detail. Nevertheless, the summary still reports the source's essence and only reports information supported by it.
- **Bad:** A summary that is extremely partial or grossly not on point due to severe information coverage issues and does an extremely partial job at communicating the sources's gist. Importantly, the summary does not add information to what the source reported. Rather, it misses crucial details or focuses on marginal ones.
- **Very Bad:** An unreliable summary. It either distorts the meaning conveyed by the source by misleadingly rearranging the details the source mentions (reasoning violations), reports information not supported by the source (hallucination) or features noticeably awkward, repetitive or nonsensical language (incoherence).

In our analysis, these four labels are further clustered into two categories: Pass (Good and Okay) and Fail (Bad and Very Bad).

Each experiment was completed by two individuals. In each experiment, we used a Latin Square method to distribute the text-summary pairs to two experimental lists. In each list, input text appeared only once but the number of items from each compared summarization method was equal. Item order was then randomized. Each experimental list was then assigned to a different individual. In order to run the experiment, we used Label Studio (Tkachenko et al., 2020-2022) as a data labeling platform. Items were presented to each participant one by one. The task was to read the text and the summary carefully, then choose the appropriate quality label (good/okay/bad/very bad). In the case of Very Bad summaries, participants also labeled whether the issue was reasoning violation, hallucination or incoherence. The evaluation process was, of course, a blind one: the interface did not feature summarization method information of any sort.

B.2 Statistical significance

To test for significance, we applied a binomial mixed-effects model (Barr et al., 2013) to each of the data sets. This type of analysis is the standard practice in psycholinguistics, and was developed to minimize random noise originating from complex differences in linguistic materials and human personal preferences and eliminate variance lost due to averaging (among other reasons). In each model, we calculated the effect of the summarization method on summary quality, while disregarding variance that can be attributed to input texts or to an evaluator.

Experiment 1: An analysis comparing "very bad" rates between the two examined summarization methods yielded a significant effect ($p = .05$), such that the AI21 Summarize API produces significantly less Very Bad summaries. All other comparisons did not yield significant results.

Experiment 2: The same analysis yielded a similar pattern of significant differences - significantly fewer "very bads" for AI21 Summarize API ($p = .008$). This pattern remained intact even when success categories were collapsed to Pass and Fail - significantly more Pass summaries for AI21 Summarize API ($p = .05$).

In summary, AI21 Summarize API was found to produce better quality summaries than both Open AI variants. Specifically, in terms of faithfulness to the source text.

References

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1206. URL https://aclanthology.org/D18-1206.

Maxim Tkachenko, Mikhail Malyuk, Andrey Holmanyuk, and Nikolai Liubimov. Label Studio: Data labeling software, 2020-2022. URL https://github.com/heartexlabs/label-studio. Open source software available from https://github.com/heartexlabs/label-studio.

Dale Barr, Roger Levy, Christoph Scheepers, and Harry Tily. Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of memory and language*, 68:255–278, 01 2013.