Summarizing in style: Exploring summarization with Large Language Models

Evgeny Pavlov
epavlov@mozilla.com

Abstract

Large Language Models (LLMs) recently became a universal tool for solving many NLP problems. One of them is text summarization. In this work, we are exploring how Large Language Models handle summarization of web articles and whether we can use their unique properties to produce summaries in a particular style to make them more engaging for a reader. We demonstrate that styled summaries are written in a richer language, have more positive sentiment and do not lose significantly in summarization accuracy.

1 Introduction

We are interested in summarization in the context of the Pocket product by Mozilla, a bookmarking tool where users save web articles to read them later. Summarization might help users to minimize their "Pocket guilt" (forever growing list of unread articles) by going through the list faster and returning to the application more often. They can either read a summary and realize that it's not something they would spend time reading or be inspired to read the full article sooner after looking at an engaging summary. Regular automatic summaries often feel hard to read. Compared to summaries, excerpts are often more engaging but they are designed to trick the user to open and read an article.

The goal is to provide a user with a comprehensive context of what the article is about to make it easier to decide whether to read the full article. The summary should include all the main points of the article in contrast to typical article excerpts. At the same time, we want the summary to be written in such a way that it inspires the user to read the full text if they are interested.

The main hypothesis is that we can generate a brief and engaging web article summary using modern LLMs. Another hypothesis is that the style of writing plays an important role in the readability, sentiment and other properties of the summary that help a user to engage with the content.

Generative abstractive summarization can produce comprehensive short summaries and make tuning of styling possible. We are skipping extractive summarization entirely as too limited for this use case. Modern LLMs are famous for their ability to generate text in different styles. We explore the "engaging" writing style of LLMs using prompt engineering. At the same time, we are interested in the feasibility of model inference or API usage in a cost-efficient way, so we use only a basic GPU machine and easily accessible APIs for our experiments.

We demonstrate that powerful LLMs available with APIs can produce styled summaries for web articles. We evaluate them on news datasets and compare them with unstyled versions and strong summarization baselines. We use traditional summarization quality metrics along with the metrics that allow us to estimate the styling effect: readability, coherence and sentiment. We show that styled summaries are just slightly less accurate but produce richer language and are more positive. Those findings are a good indicator that users would enjoy such summaries more and likely would engage with the content after reading them. A broader discussion is how tunable LLM styling is to tailor summarization to different use cases in general, for example sharing on social networks, specifying target audience etc.

2 Related work

2.1 Length of content

Web content can be of arbitrary length and it is important to take into account the limitation of the number of tokens when doing summarization. The paper "An Empirical Survey on Long Document Summarization: Datasets, Models and Metrics" (Koh et al., 2022) provides a great overview of existing summarization methods, specifically in the context of long documents. The paper makes it clear that longer documents require a different approach for summarization. For simplicity, we will consider only shorter web articles in this work.

2.2 Summarization type

Summarization can be extractive when the pieces of content are used in the summary or abstractive when the summary text is generated from scratch. We are interested in abstractive summarization to be able to experiment with the style of writing.

"BRIO: Bringing Order to Abstractive Summarization" (Liu et al., 2022) introduces a SOTA abstractive summarization transformer-based model. It can be used as a strong baseline for summarization quality.

2.3 Evaluation

Evaluation is the main area of uncertainty that is visible across various summarization publications. It is clear that summarization is very subjective and there is no one perfect way to evaluate it. "SummEval: Re-evaluating Summarization Evaluation" (Fabbri et al., 2022) tried to address this by creating a framework for all popular evaluation metrics. The authors performed a study with carefully selected human annotators to estimate the correlation of different evaluation methods with human judgement.

Some papers use classic similarity approaches like ROUGE score and base their research and findings exclusively on those metrics.

Others claim that popular metrics, especially string-matching ones are outdated and do not correlate with human judgement. Most of the papers define their variation of the evaluation metrics, so we can conclude that the choice of metrics is very problem specific.

2.4 Summarization with LLMs

"Language Models are Few-Shot Learners" (Brown et al., 2020) demonstrates the impressive capabilities of Generative Pretrained Transformers on a variety of generic NLP tasks. OpenAI API documentation specifically states summarization as one of the tasks their GPT models are capable of.

"Benchmarking Large Language Models for News Summarization" (Zhang et al., 2023) shows that designing the right prompts is more important than the size of the model. The authors also argue that the quality of current popular datasets does not provide enough confidence to estimate the performance of LLMs. The authors hired professional writers and annotators to compare the summarization capabilities of LLMs with summaries written by people and concluded that the performance of LLMs is on par with human writers.

"News Summarization and Evaluation in the Era of GPT-3" (Goyal et al. 2022) compares the performance of several fine-tuned summarization models to a generic GPT one (BRIO, T4 and Open AI GPT3 Davinci v2). The main finding like in the previous paper is that automatic evaluation algorithms do not correlate with human judgement when GPT models come into play. Even though summaries by GPT models were preferred by human evaluators, all automatic algorithms favor T0 or BRIO including reference-free ones that rely on question answering. General-purpose GPT3 style language models are not finetuned on popular summarization datasets and produce quite a different style of summarization that break the existing metrics but is still preferred by the users.

Overall, the research on summarization with LLMs appears to be limited. Most of the papers show that automatic methods of evaluation sometimes do not correlate with the human judgement which poses challenges in the estimation of models' capabilities. We can conclude then that automatic evaluation is not enough and some sort of human evaluation is always required for summarization by LLMs.

2.5 Styling

Publications on summarization styling are also scarce. "Inference Time Style Control for Summarization" (Cao and Wang, 2021) presents methods to generate style summaries with transformer models. Like in previous papers, one of the insights is that evaluation of styling is challenging and requires human evaluation.

3 Data

3.1 Choosing datasets

We use popular open-source evaluation datasets that often appear in research papers. For our use case of web articles, the news is the closest data source. Documents or scientific publications would not be a good choice taking into account their length and niche domain. Having the significant size of some datasets we sample a subset of examples to ensure API and inference costs are reasonable.

3.2 CNN / DailyMail

The CNN / DailyMail dataset contains over 300k unique news articles written by journalists at CNN and the Daily Mail. It supports both extractive and abstractive summarization. The original version was created for machine-reading and comprehension and abstractive question answering. The summaries of this dataset are typically 2-4 sentences. We use the test split of the HuggingFace dataset for sampling. Summary example: *"Photographer James Oatway captured a violent attack that resulted in death of a Mozambican in South Africa. Seven people have been killed in recent violence against poorer immigrants, many from South Africa's neighbors."*

3.3 Xsum

The Extreme Summarization dataset is a dataset for the evaluation of abstractive single-document summarization systems. It contains ~200k articles that are collected from BBC and cover a wide variety of domains. The summaries of this dataset are one sentence. We use the test split of the HuggingFace dataset for sampling. Summary example: *"The BBC's Global Health Correspondent Tulip Mazumdar has been investigating a new Zika vaccine which could be ready for human trials later this year."*

3.4 Newsroom

This is a large dataset for training and evaluating summarization systems (Grusky et al., 2018). It contains 1.3 million articles and summaries written by authors and editors in the newsrooms of 38 major publications. This dataset is available to download after submitting a form. The dataset provides metadata for the examples and we filter only ones that are marked as "abstractive". The summaries of this dataset are typically 2-3 sentences. Summary example: *"She has clocked up more air time on television than anyone else. But for Carole Hersee, her return to our screens after an absence of more than 10 years has come as a complete surprise."*

3.5 Dataset length comparison

Length of texts and summaries is an important factor for our experiments. We have to filter texts that are too long and design prompts to generate summaries of the length similar to reference ones.

3.6 Length issues

The specifics of Pocket are somewhat longer articles. Average length is 812 words and 95th percentile is 2174 words. The texts in evaluation datasets are a bit shorter than average Pocket articles but still should provide a good benchmark for the model performance.

Dataset	Summary length	Text length
CNN/DailyMail	55	683
Xsum	21	386
Newsroom	26	652

Table 1: Average dataset text length.

To overcome this issue in a real production system we might explore in future whether the models are capable of summarizing parts of the articles to preserve most of the details and then doing the final summarization of summaries with style tuning or using prompt chaining. This is out of the scope of this work.

4 Models

We use the pre-trained language models to perform summarization. Easily available Eleuther models are a good baseline for LLMs. Specialized BRIO serves as a strong baseline for summarization quality.

4.1 Eleuther

Eleuther family of models (125M-6B parameters) (Black et al., 2021) - there are smaller models that can be inferred on a CPU and are convenient to set up experimentation pipelines and larger ones that can serve as an LLM baseline for prompt-based quality evaluation. We use gpt-neo-1.3B for our experiments as large enough to produce something

useful and the one that we were able to run on a 12GB GPU.

4.2 BRIO

BRIO - a task-specific summarization model that demonstrates SOTA results for abstractive summarization. It is useful as a baseline for automatic quality metrics. We use two BRIO variations. The first is fine-tuned on CNN/DailMail and the second is fine-tuned on the Xsum dataset. Both models are accessible with the HuggingFace library.

4.3 OpenAI API

OpenAI models are the most interesting ones because very powerful LLMs are packed behind a simple API and are easy to use. We use those models to explore styled summaries.

- **Open AI API Davinci** - the best model from InstructGPT family (GPT3.5, max 2K tokens)
- **Open AI API Curie** - faster and cheaper model, officially recommended for summarization (GPT3, max 2K tokens)

5 Experiments

5.1 Evaluation data preprocessing

Having high variability in length for both texts and summaries of evaluation news datasets we apply filters to make the distribution of length more predictable and aligned with the input and output of LLMs we want to test.
We use the following filters for test splits of evaluation news datasets:

1. 500 <= Text lenth <= 1000 tokens
2. 10 <= summary length <= 60 tokens
3. Use only "abstractive" examples for Newsroom

Then we sample 100 examples for each dataset. The total evaluation corpus is 300 examples. The main consideration for the data size is to be able to run evaluation quickly having the limited resources but still have enough examples to observe meaningful results.

5.2 Running experiments

We used a basic 12GB GPU provided by Stanford. GPU memory limits usage of larger LLMs. We were able to run inference only for Eleuther 1.3B. Using V100 or A100 would help to experiment with larger models.

We run evaluation experiments on pre-trained language models. For summarization-specific models (BRIO) we just feed text, for prompt based we design the input that includes the text and the prompt by telling the model what kind of summary we expect.
We assume that by tuning the prompts of GPTs we'll be able to produce summaries in a desired style.

5.3 Eleuther

We experimented with different prompts. The smaller Eleuther 1.3B model is very sensitive to the prompt and we could make it produce proper summaries only with the simplest prompt *"<text>\nSummary:"*. Even adding another *"\n"* makes the results worse. We tried to specify the style in the prompt but this didn't work with Eleuther. Using large models would probably solve this.

It is important to direct the model to produce the desired number of sentences and length of the summary, so we used different *"max_new_tokens"* parameter for different datasets.

It is worth noting that the raw response from Eleuther is pretty challenging to process, the model tends to repeat a phrase multiple times. We used code from the Stanford course notebooks to handle this and get clean summaries.

5.4 OpenAI models

For the non-styled prompts we used:
"<text>\n\nSummarize the above article briefly in 2 to 3 sentences." for CNN/DailyMail
"... in 3 to 4 sentences." for Newsroom
"... in 1 sentence." for Xsum

For the styled models:
"<text>\n\nGenerate a brief and engaging 2 to 3 sentence summary for the article above." for CNN/DailyMail and similar to non-styled ones, *"1"* and *"3 to 4 sentences"* for Xsum and Newsroom.

Observations:

- word "*brief*"/"*briefly*" is important to make summaries shorter and to the point.
- Task phrasing matter a lot for styled summaries. We experimented with it and found that this variant works best

Parameters:

- *Temperature=0.1* for non-styled models and *0.9* for styled ones. The idea is to add more randomness and make the models more creative with styling.
- *TopP=0.95*

5.5 Evaluation

We are interested in how comprehensive the summaries are and whether they provide the main points of the article. Automatic similarity-based metrics can provide some approximation here.

ROUGE - (Recall-Oriented Understudy for Gisting Evaluation), measures the number of overlapping textual units (n-grams, word sequences) between the generated summary and a set of gold reference summaries. There are also many variations of the ROUGE metrics, we use four of them (ROUGE-1, ROUGE-2, ROUGE-L,)

BertScore - computes similarity scores by aligning generated and reference summaries on a token level. Token alignments are computed greedily to maximize the cosine similarity between contextualized token embeddings from BERT

Automatic evaluation of style is challenging. We calculate some metrics based only on output text that provide some insights into how different the style and unstyled versions are.

Readability - calculates different metrics that show how readable the text is for a human. We calculate the following metrics: Gunning-Fog index, SMOG index, Flesch reading ease, Flesch-Kincaid grade, Automated Readability Index, Coleman-Liau index, Lix score, and Rix score. They are mostly convertible to the education grade required to read the text. The higher the score, the higher the grade.

Coherence - calculates the coherence of the document, based on word embedding cosine similarity between sentences. First-order coherence: the cosine similarity between consecutive sentences. Second-order coherence: the cosine similarity between sentences that are two sentences apart. These metrics cannot be calculated for all examples because some summaries are just one sentence.

Sentiment - we calculate positive-rate which is the percentage of examples from a dataset that were classified as positive by the binary sentiment classifier.

Automatic metrics are imperfect for the evaluation of style, so we perform a subjective human evaluation on a limited sample of produced summaries.

Human evaluation of style - how engaging the generated text is from a human perspective.

6 Analysis

Looking at the evaluation results for summarization quality in Table 2 and Table 6 BRIO is a clear leader which is not surprising taking into account that this is a specialized summarization model that was fine-tuned on CNN/DailMail and Xsum datasets. Eleuther model is a clear outlier because of its smaller size. GPT3.5 based Davinci

model	rougeL	BERT score f1
eleuther1b	0.183	0.857
curie	0.215	0.874
style_curie	0.214	0.876
davinci	0.229	0.880
style_davinci	0.217	0.878
brio	0.310	0.876

Table 2: Summarization quality, CNN/DailyMail

performs better than other LLMs which is expected. It is interesting that styled version of Davinvi performs worse than the unstyled one consistently across all metrics and datasets, unlike Curie which shows some variation depending on the dataset. We think this is because of the higher language styling capabilities of GPT3.5. The model is more creative and uses richer language.

Readability scores provide interesting insights into the complexity of the generated texts. BRIO

model	Flesch Kincaid grade
eleuther1b	8.883
curie	9.826
style_curie	9.987
davinci	9.661
style_davinci	10.745
brio	6.661

Table 3: Readability, CNN/DailyMail

shows significantly simpler texts than LLMs, however, we think it's because it generates shorter and more concise summaries. LLM size and styling appear to have a negative effect on readability. Davinci produces less readable text than Curie and

model	First order coherence	Positive rate
eleuther1b	0.799	0.42
curie	0.800	0.41
style_curie	0.808	0.46
davinci	0.791	0.49
style_davinci	0.810	0.50
brio	0.753	0.42

Table 4: Coherence and sentiment, CNN/DailyMail

the styled version of Davinci consistently generates less readable text than the unstyled one. We think that by getting instructions to produce "engaging" style the models try to use more creative phrasing that have a negative effect on readability. Also, a higher temperature setting most likely affected the results.

The coherence metric is distorted by the number of sentences the model produces. However, we can see that styled versions of models mostly produce texts with higher coherence. Looking at CNN as a dataset where most of the examples are multi-sentence, BRIO produces the lowest coherence, which is surprising. Probably LLMs are just better at generating text as a related sequence of sentences.

Looking at the final metric - positive-rate we can conclude that Davinci is a consistent leader in producing text with more positive sentiment. Styled versions of models also mostly produce texts with higher rates except for Davinci on Xsum. This metric is pretty tricky for the news datasets, because a lot of news describe negative events. However, we think this is a good indicator that more powerful LLMs and the usage of styling can increase the engagement of users with summaries.

We also looked visually at the summaries to better understand the differences in styled and unstyled versions. It feels like styled summaries use richer language and evoke stronger emotions and desire to engage with the content. However such evaluation is very limited and subjective. It would be interesting to perform a large-scale experiment to get feedback on how styled summaries effect user experience.

7 Limitations

Evaluation is performed using a limited set of easily accessible automatic metrics. There are many more ways to evaluate summarization quality including referenceless and QA-based metrics.

Using smaller open-source LLMs. There is a lot of room for exploration of larger LLMs. More expensive hardware is required for this. However, those larger models might be able to compete in quality with popular APIs. More tuning and prompt engineering would be likely required.

Length of texts. We specifically left lengthy articles out of scope. Understanding how to incorporate longer articles with LLMs would be another interesting direction for exploration.

Small dataset size. Some results might not be statistically significant having a small size of the dataset.

Lack of proper human evaluation. Most of the papers claim how important human feedback is in the evaluation of LLMs. Authors hire people to perform such an evaluation.

8 Conclusion

We explored a promising direction of styled summarization using modern Large Language Models. The larger models appear to have pretty good quality compared to the specialized solutions like BRIO. At the same time, those models generate summaries in richer language and with a more positive sentiment which might be important factors to make the summaries more engaging for users. More experiments with larger open-source models and high scale user experiments are required to make more confident conclusions about cost efficiency and user impact of styled summarization. For regular summarization, specialized models would probably work well and be cheaper. However, if we want summaries to be tailored for specific use cases LLMs appear to be the right tool for the job because of their flexibility.

Acknowledgments

We would like to acknowledge the great work of the Stanford NLP group and Prof. Christopher Potts for providing high quality learning materials and guidance during the course.

Authorship statement

The work was performed by Evgeny Pavlov. The ideas for the work were discussed with the colleagues on the Pocket team at Mozilla. Prof. Potts helped to direct this work to be more focused during the Q&A session.

References

Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, Dragomir Radev. 2022. *SummEval: Re-evaluating Summarization Evaluation*. Linguistics 2021; 9 391–409, https://doi.org/10.1162/tacl_a_00373

Huan Yee Koh, Jiaxin Ju, Ming Liu, Shirui Pan. 2022. *An Empirical Survey on Long Document Summarization: Datasets, Models and Metrics* ACM Computing SurveysVolume 55 Issue 8 Article No.: 154pp 1–35 https://doi.org/10.1145/3545176

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. *Newsroom: A Dataset of 1.3 Million Summaries with Diverse Extractive Strategies*. NAACL 2018, https://aclanthology.org/N18-1065

Shuyang Cao and Lu Wang. 2021. *Inference Time Style Control for Summarization*. Arxiv, https://doi.org/10.48550/arXiv.2104.01724

Sid Black, Leo Gao, Phil Wang, Connor Leahy, Stella Rose Biderman. 2021. *GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow.* Zenodo, https://doi.org/10.5281/zenodo.5297715

Tanya Goyal, Junyi Jessy Li, Greg Durrett. 2022. *News Summarization and Evaluation in the Era of GPT-3*. Arxiv, https://doi.org/10.48550/arXiv.2209.12356

Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown and Tatsunori B. Hashimoto. 2023. *Benchmarking Large Language Models for News Summarization*. Arxiv, https://doi.org/10.48550/arXiv.2301.13848

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. *Language models are few-shot learners*. In Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS'20). Curran Associates Inc., Red Hook, NY, USA, Article 159, 1877–1901.

Yixin Liu, Pengfei Liu, Dragomir Radev, Graham Neubig. 2022. *BRIO: Bringing Order to Abstractive Summarization.* In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2890–2903

A Appendices

All collected metrics during the evaluation are listed in Table 5, Table 6 and Table 7.

dataset	model	rouge1	rouge2	rougeL	rougeLsum	BERT score precision	BERT score recall	BERT score f1
cnn	eleuther1b	0.265	0.094	0.183	0.183	0.853	0.862	0.857
	curie	0.321	0.127	0.215	0.215	0.861	0.888	0.874
	style_curie	0.317	0.110	0.214	0.214	0.870	0.882	0.876
	davinci	0.343	0.130	0.229	0.229	0.867	0.894	0.880
	style_davinci	0.328	0.116	0.217	0.217	0.864	0894	0.878
	brio	0.435	0.210	0.310	0.311	0.864	0.888	0.876
newsroom	eleuther1b	0.174	0.032	0.125	0.125	0.829	0.834	0.831
	curie	0.219	0.040	0.144	0.144	0.845	0.850	0.848
	style_curie	0.223	0.045	0.146	0.147	0.852	0.851	0.851
	davinci	0.238	0.047	0.159	0.159	0.850	0.854	0.852
	style_davinci	0.233	0.042	0.150	0.150	0.847	0.855	0.851
	brio	0.246	0.066	0.170	0.170	0.875	0.850	0.862
xsum	eleuther1b	0.192	0.037	0.142	0.142	0.852	0.858	0.855
	curie	0.231	0.045	0.165	0.164	0.866	0.871	0.868
	style_curie	0.239	0.056	0.171	0.171	0.869	0.874	0.871
	davinci	0.260	0.074	0.185	0.185	0.868	0.880	0.874
	style_davinci	0.250	0.057	0.173	0.173	0.862	0.881	0.871
	brio	0.418	0.194	0.331	0.331	0.912	0.904	0.908

Table 5: Summarization quality

dataset	model	Flesch reading ease	Flesch Kincaid grade	smog	Gunning fog	Automated readability index	Coleman liau index	lix	rix
cnn	eleuther1b	70.364	8.883	9.834	11.778	11.439	10.016	44.222	4.914
	curie	66.894	9.826	10.578	12.647	12.671	10.637	47.359	5.622
	style_curie	65.181	9.987	10.916	12.817	12.887	11.105	48.346	5.807
	davinci	65.485	9.661	10.853	12.427	12.422	11.192	47.125	5.530
	style_davinci	62.004	10.745	11.451	13.515	13.728	11.441	50.265	6.333
	brio	74.993	6.661	9.358	9.589	8.615	10.034	39.006	3.638
newsroom	eleuther1b	69.494	9.008	9.736	12.114	10.782	9.049	44.080	4.943
	curie	63.752	10.141	11.105	13.250	12.834	11.063	48.303	5.800
	style_curie	63.584	10.113	10.977	13.008	12.701	11.060	48.063	5.694
	davinci	57.214	11.687	11.270	14.567	14.740	12.011	53.529	7.149
	style_davinci	55.139	12.403	12.348	15.384	15.698	12.258	55.556	7.649
	brio	66.432	10.203	0.000	13.246	12.561	9.711	46.460	5.485
xsum	eleuther1b	73.021	7.715	8.157	10.466	9.405	9.220	40.323	4.028
	curie	66.626	9.456	10.662	12.233	12.033	10.720	46.395	5.338
	style_curie	63.131	10.146	11.262	13.077	13.135	11.649	49.383	6.122
	davinci	51.530	14.157	0.000	17.141	18.120	12.284	59.180	8.585
	style_davinci	41.701	17.026	0.000	20.083	21.645	13.091	66.590	10.770
	brio	67.200	9.333	0.000	12.029	11.577	10.279	44.470	4.850

Table 6: Readability

dataset	model	First order coherence	Second order coherence	Positive rate
cnn	eleuther1b	0.799	0.753	0.42
	curie	0.800	0.786	0.41
	style_curie	0.808	0.802	0.46
	davinci	0.791	0.792	0.49
	style_davinci	0.810	0.810	0.50
	brio	0.753	0.745	0.42
newsroom	eleuther1b	0.855	0.838	0.44
	curie	0.812	0.774	0.44
	style_curie	0.826	0.800	0.49
	davinci	0.841	0.808	0.63
	style_davinci	0.835	0.819	0.64
	brio	0.860	0.000	0.51
xsum	eleuther1b	0.759	0.677	0.47
	curie	0.794	0.789	0.44
	style_curie	0.797	0.789	0.49
	davinci	0.604	0.536	0.54
	style_davinci	0.591	0.548	0.51
	brio	0.541	0.654	0.48

Table 7: Coherence and sentiment