THE PERILS OF GENERATIVE MODEL INBREEDING: EVALUATING THE CONSEQUENCES OF CROSS-MODEL TRAINING IN LARGE LANGUAGE MODELS

by

Gabrielle Stein

May, 2024

Director of Thesis: Nic Herndon, PhD

Major Department: Computer Science

What happens when the output of generative AI models is included in the training data of new models? With the rise of generative AI content online, and considering that most training data for AI models is sourced from the Internet, concerns have arisen about how this generated content might taint future training datasets. Existing research has evaluated the effect of models consuming *their own output*, and has shown that the output of self-consuming models degrades with each successive generation of re-training, a phenomenon coined as "model collapse." This degradation takes the form of a loss of diversity in the output of the model. Currently there is limited research on the impact of models consuming *other models' output*, specifically large language models. In this study we aimed to determine the effect of training a model on a different model's output. Additionally, we developed a potential solution to prevent "model collapse." Guaranteeing the majority of training data is guaranteed to be human-generated (non-synthetic) data has been shown to mitigate the loss of diversity caused by "model collapse." Given that AI models are here to stay, the methods for developing new models will need to evolve to address this issue, ensuring that AI development can continue to progress and improve.

THE PERILS OF GENERATIVE MODEL INBREEDING: EVALUATING THE CONSEQUENCES OF CROSS-MODEL TRAINING IN LARGE LANGUAGE MODELS

A Thesis

Presented to The Faculty of the Department of Computer Science

East Carolina University

In Partial Fulfillment of the Requirements for the Degree

Master of Science in Computer Science

by

Gabrielle Stein

May, 2024

Director of Thesis: Nic Herndon, PhD

Thesis Committee Members:

Rui Wu, PhD

David Hart, PhD

Table of Contents

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

The prevalence of artificial intelligence generated content (AIGC) across the internet has exploded, with nearly every online service incorporating some artificial intelligence (AI) feature or application. With this, the amount of AIGC on the internet is also increasing at a rapid pace. Recently, a study has proposed the concept of "**model collapse**" which states that generative models trained on AIGC will experience a degradation in their results, which manifests as a decrease in the diversity of the generated material [24].

Shumailov et al.[24] posits that model collapse is unavoidable without access to non-synthetic (human generated) data. To address this, we propose actionable steps that can be taken during model training to mitigate model collapse. The simplest solution is guaranteeing that the majority of the training data is verified to be human generated and not AIGC. Additionally, we evaluated the effect of a model being trained on data generated by a different model, which we refer to as **cross-training**.

With the introduction of Transformers architecture by Google in 2017, the development of generative AI, specifically large language models (LLMs), has leapt forward [27]. This new architecture has been used to create models which are trained on huge datasets. This gives the models the ability to handle a variety of prompts and have diverse capabilities. These LLMs produce text that is extremely human-like. How-

ever, this new advancement in AI model development has the potential to lead to issues with future models in the form of model collapse.

Research into model collapse is novel, but Shumailov et al.[24] has shown that it can be detrimental to generative models, and that it affects many types of AI models. Therefore, action needs to be taken to prevent model collapse as soon as possible. For generative models to be able to interact with up-to-date, contemporary knowledge, they need to be trained on up-to-date material, which is sourced from the internet. However, the amount of AIGC on the internet has dramatically increased, especially with easy accessibility to models like ChatGPT. Therefore, the likelihood that data pulled from the internet will contain AIGC will only increase as more time passes. The more AIGC there is online, the more likely this content will be included in training sets and lead to model collapse.

Current research by Shumailov et al.[24], as well as an additional study on model collapse that came out concurrently [2], has shown that model collapse is avoided by training on human-generated (non-synthetic) data. However, given that large training sets of LLMs contain billions of examples, and hundreds of billions of tokens, it would be extremely difficult to identify and remove AIGC. In addition to being completely infeasible to do this manually given the large size of these datasets, it is also difficult for both humans and other models to detect AIGC. There is also little to no regulation requiring the AIGC to be labeled as such online, so it cannot be removed this way either.

Recently there have been pushes to develop technologies made to detect AIGC. But the effectiveness and accuracy of these tools has been called into question [8, 29]. Therefore, model developers must act under the assumption that there is a chance when scraping a large amount of internet data, this data will contain AIGC and currently the available tools are not sufficient to detect and remove it.

A potential solution would be to only train models on internet data collected before AIGC started to proliferate on the internet. However, this would not be viable as this would limit the amount of contemporary knowledge an AI would have, which is not desirable for LLMs as it limits their ability to answer questions about contemporary topics. Another solution is that in the future there could be a large, open-source push to collect a large amount of human-generated content into training sets for model-training purposes. However, this would take a large amount of time and resources, so it seems not feasible to deal with the immediate threat of model collapse.

Considering that existing solutions are non-viable, we aimed to develop simple, actionable steps that can be taken to diminish the risk of model collapse immediately. We propose that by guaranteeing a certain percentage of training data is human generated. This means that the rest of the training data can be sourced from the internet and might include AIGC, but this effect of model collapse will be diminished compared to using all training data sourced from the internet. This can be done easily when sourcing training data and doesn't incur additional cost to modify the architecture of the generative model.

Additionally, we aimed to answer two questions with our experiment:

1. What happens when a model is trained on another model's output? We refer to this concept as cross-training. We aim to determine if cross-model training lessens the effect of model collapse.

2. Is there an inflection point where the percentage of generated data compared to human data leads to a decrease in the accuracy of the model? We aim to determine if there is a specific percentage of human data that can be included in the training data that will lessen the impact of model collapse.

Chapter 2

Related Works

There is currently a dearth of research investigating model collapse. The concept of model collapse was first introduced by Shumailov et al. [24], in which they proposed that training a model on generative data misrepresents the true distribution of the data and leads to a degradation in the output of the model. They state the importance of human (non-synthetic) data in the training sets of models.

In this section we will evaluate current research into model collapse, as well as other factors that contribute to model collapse including AI content on the internet, AI detection and watermarking.

2.1 Model Collapse

Shumailov et al. [24] stated in their study that model collapse is inevitable without access to non-synthetic data. They also showed that it affects many kinds of generative models and model collapse has two stages: "early stage" and "late stage."

"Early stage" model collapse involves a model trained on data that was produced by a model that has only gone through a few cycles of generative-model inbreeding. This means that maybe it is trained mostly on human-generated data instead of AIGC. It could also mean it was trained on model-generated data but the model that generated the data was trained on human data, so it has only done through a single

cycle of inbreeding.

"Late stage" model collapse involves a model trained on data produced by a model that has gone through the many cycles of re-training on AIGC. Shumailov et al. [24] stated that this can lead to the output of the model zeroing in on a single point in the data distribution.

Their experiment focused on "early stage" model collapse because generative models have only just achieved wide-spread prominence. With the recent release of generative models like Chat-GPT and MidJourney, it is unlikely for there to have been an opportunity to reach "late stage" model collapse at this point. In other words, it is unlikely that AIGC from models that have been trained on AIGC will appear in large training datasets sourced from the internet. GPT-3 was trained on multiple large training sets, the largest of which is Common Crawl[1] [3]. The internet data contained in the Common Crawl was collected when generative models weren't as prevalent and thus these datasets were less likely to include AIGC. Therefore, we think that "early stage" model collapse is the most pressing concern currently. If we develop tactics to mitigate early stage" model collapse, "late stage" model collapse could be avoided.

Shumailov et al. [24] also showed that including a small percentage of human data in training set in addition to the AIGC they obtained through recursive training was able to lessen the effect of model collapse.

Another paper about model collapse was published roughly concurrently, which called model collapse MAD (Model Autophagy Disorder) [2]. Alemohammad et al. [2] agreed with the conclusions reached by Shumailov et al. [24], that model collapse is inevitable without access to non-synthetic data. They also state that models have a threshold for how much synthetic data they can be trained on before it impacts the

[1] https://commoncrawl.org/overview

model's output, stating that in specific instances, a small amount of synthetic data obtained through biased sampling can improve a model's performance, yet strongly decrease the diversity of the models' output. They state that the loss of diversity caused by model collapse can be observed as a loss of less-likely outcomes over successive training cycles, and supported the conclusion that model collapse can be seen in all types of generative models. In addition to these conclusions, they showed that watermarking in models, which they refer to as "architectural fingerprints" of models are amplified by model collapse.

A new study proposed metrics specifically meant to measure the diversity of output in LLMs to help measure an LLM experiencing model collapse [9]. They looked at ways to measure the three cornerstones of language: lexicality, semantics, and syntactically. Additionally, they proposed that tasks that provide less context and expect more variability in the response, which they refer to as "high entropy" tasks, are more susceptible to model collapse. This essentially states that tasks asking an LLM to be more"creative" will experience model collapse faster over recursive retrainings and will experience a loss in linguistic diversity more acutely.

The conclusions reached in these studies establish that model collapse exists and has detrimental effects, provide general guidelines on how to avoid model collapse by using non-synthetic data, and ways to measure model collapse's effects. However, Guo et al. [9] didn't investigate the sensitivity of the proposed linguistic diversity metrics. Additionally, these studies only evaluated models trained on their own output. Our study aims to evaluate the effect of an LLM trained on the output of a different LLM.

2.1.1 AI Generated Content on the Internet

The amount of AIGC on the internet has increased in the last few years, but it is difficult to measure or approximate the amount, as it is rarely labeled, and much

of it is intentionally presented as created by humans instead of AI. While the exact amount is difficult to measure, there has been a marked increase.

From an ethical perspective it is undesirable to have unlabeled AIGC on the internet. LLMs have already been shown to be used for nefarious purposes, such a powering bot networks on social media [31]. In addition to this being an ethical concern, it can also be seen as detrimental to the long-term goal of developing and innovating LLMs. There is now a likelihood that data obtained from social media will contain responses from bots that made those responses using AI, and if you were to train/fine-tune a model using this data your model could begin to exhibit symptoms of model collapse. Social media is the basis for communication across the world and responsible for much of modern pop culture, so there will have to be decision made on whether the information obtained from social media data will be worth the risk of model collapse.

A fortunate change some companies are implementing is requiring AI content to be labeled. This includes social media companies like Meta[2] and art-sharing platforms like DeviantArt[3], which have updated their terms of service to require AI content to be labeled. This can help to filter out synthetic content from web-scraped datasets. However, content moderation to enforce this rule would be extremely challenging as AIGC is designed to appear human-generated, and the accuracy of AI detection tools has been called into question. Additionally, these new policies are still in the process of being introduced and not every company has such policies, so they cannot be relied on currently to prevent model collapse.

[2] https://about.fb.com/news/2024/02/labeling-ai-generated-images-on-facebook-instagram-and-threads/

[3] https://www.deviantart.com/about/policy/service

2.1.2 Watermarking and AI Detection Tools

We have established the importance of using non-synthetic training data. When collecting extremely large datasets from the internet that have the potential of containing AIGC, there are a few proposed solutions. One solution is detecting and then removing such content using AI detection tools. For such large amounts of content this would have to be automated rather than manually sorted, requiring the use of machine-learning driven tools. However, there is ongoing debate among researchers on the efficacy of AI detection tools.

In general, it is hard for humans to identify AI generated text. One study found that participants were only able to correctly identify AI generated text 24% of the time [14]. LLMs are designed to generate text that emulates human text that they are trained on. As these models improve their accuracy, they will become harder for humans to detect.

Even if humans were better at detecting AIGC, it is not feasible to try to manually detect and remove it. For instance, the training sets used to train GPT-3 were approximately 500 billion tokens, with 60% of the dataset being sourced from Common Crawl, which many would consider takes "snapshots of the entire internet" for internet archiving purposes [3]. The amount of data is impossible for humans to manually sort into synthetic and non-synthetic categories for training purposes. For that reason, a proposed solution is to have AI detection tools that can automate the process.

There is a multitude of viewpoints among researchers about the efficacy of AI detection, both theoretically and in practice.

Some researchers purported that the AI detection is theoretically possible because AI generated texts and human-generated texts have different distributions [4]. Others

say that paraphrasing AI generated texts can make them significantly harder to detect using AI detection tools [13]. This has concerning consequences not just for AI detection to avoid model collapse, but also for education. There are already concerns about being able to reliable detect whether an LLM wrote parts of education and academic research [5]. The fact that paraphrasing can be used to avoid detection could compound this problem. Another study showed that this effect can be compounded using recursive paraphrasing, which can make the detection rate of an AI text drop to under 60% [23].

How accurate are AI detection tools in general? One study that did a review of 16 available AI detection tools found that while some tools were acceptably accurate when detecting GPT-3 texts, the tools struggled to detect GPT-4 texts [8, 29]. A different study of available tools showed that some tools had high accuracy for detection, other tools struggled to detect AIGC, and all the tools had a significant percentage of false positives [1].

One proposed solution to improve AI detection is watermarking models. For text generation models, this consists of formatting models so that their responses are always recognized as generated by the model, which is referred to as semantic-watermarking. There are currently many proposed ways to watermark LLMs. However, one study has stated that using semantic watermarking for the purposes of AI detection is impossible [32]. They showcased a watermarking attack that can be used to remove multiple watermarking methods with minimal loss in quality. Additionally they showed that watermarking will inevitably lead to a reduction in the quality of the models response, currently making it a non-viable solution for model collapse. Watermarking would also require changing the structure of the model itself which is expensive.

Therefore, we propose a solution that only requires curating the training data of

the model to include datasets guaranteed to be written by humans. This can take the form of large corpus of books written by humans, text written before the prevalence of transformer models, or some other curated dataset where the contents are verified to be written by humans.

Chapter 3

Materials and Methodology

This section outlines the training and generation methodology of the experiment.

We aimed to see if the ratio of human-to-model generated data affected the rate of model collapse. Specifically, if there is a minimum amount of human generated data that needs to be in a model to lessen the effect of model collapse. Previous experiments showed that recursively training a model on its own output leads to models collapse. We wish to evaluate the effects of a model trained on a different model's output, or cross-model training.

Code for this experiment is available at https://github.com/gabriellestein/thesis.

3.1 Experimental Overview

We have split the structure of the experiment into cycles, shown in Figure 3.1. Each cycle will have a LLM, referred to as Model-1, that will be fine-tuned on a dataset. This dataset is meant to simulate real-world data that has been sourced from the internet.

As more LLMs are released and used by the public, the more diverse the origins of AIGC seen online will become. Currently, most AI content on the internet is produced by GPT models, specifically from Chat-GPT. However, as the market evolves more companies will begin releasing competitors. This is already shown through the release

of LLMs such as Mistral or Gemini. This will undoubtedly increase the amount of AIGC online.

This dataset is treated as if it was scraped from the internet with unlabeled AIGC. This dataset will be referred to as Dataset-0. Each cycle of the experiment will start with 25 percent less human-generated training data than the previous cycle, moving from 100% to 0% training data.

After fine-tuning, Model-1 will generate text that will make up a new dataset, Dataset-Model-1. This would be used to fine-tune a different model than Model-1 which we will refer to as Model-2. Model-2 will then generate a second dataset, referred to as Dataset-Model-2. This is how we will implement cross-model training.

We hypothesized that a model with a different architecture would experience less of a loss of diversity caused by model collapse. Different models are trained on different-sized contexts and are shown to have different levels of accuracy. For that reason, training a model on another model's output might "trick" the model into thinking this model was generated by a human instead of another model.

We wanted to evaluate how cross-model training affects linguistic diversity com-

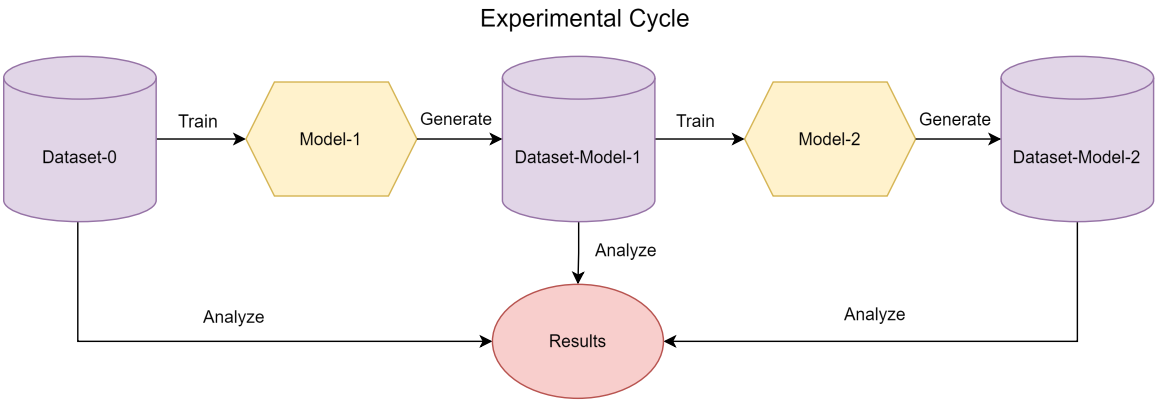

Figure 3.1: Generalized diagram showing the basic steps taken in our experimental design. Every "cycle" of the experiment, which begins with a different percentage of synthetic data in Dataset-0, follows each step listed above before data analysis.

pared to model collapse caused by self-consuming models, or models trained on their own output. To get a baseline or control for how Model-2 would respond would respond to being trained on its own output, as well as being trained on a different model's output, we also completed a series of cycles where Model-2 was trained on percentages of its own data. This cycle is shown in Figure 3.2. We began with the same dataset from the original cycles, Dataset-0. After Model-2 was trained on this dataset and then generated Dataset-Model-2-Base, percentages of this output would be injected into Dataset-0 to retrain Model-2, the same way it was for the cross-training experimental cycle.

After this has been done for all the cycles from 100% to 0% for both the baseline and cross-training experimental cycles, we will evaluate the datasets to see whether and how much each model is exhibiting symptoms of model collapse.

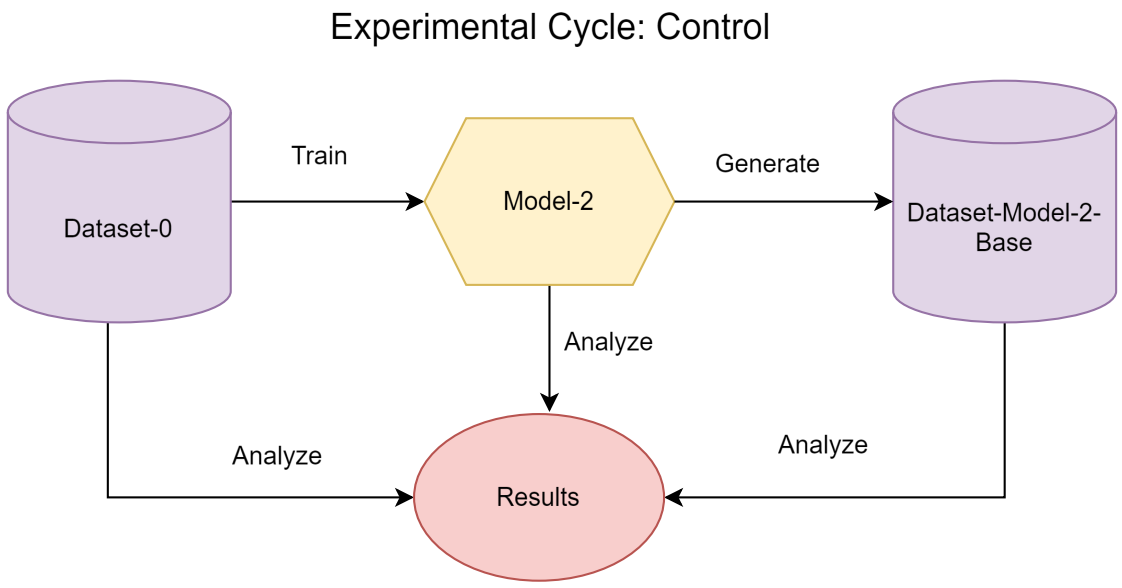

Figure 3.2: Generalized diagram of experimental cycle to determine control. This cycle is meant to establish a baseline for how a Model-1 responds to self-consuming its own output compared to the output of another model.

3.1.1 Models

Considerations when choosing the models to use in the experiment:

- The models must be open source. This is to support open-source model development and will be more cost-effective. Model architecture and troubleshooting will be easier with open-source models. Additionally, the results of this experiment will be more valuable from a replicability perspective.

- The models must have different architectures or from different model "families." This is an effort to "trick" the model into believing that text was human generated instead of machine generated. In practice, this means that both models cannot be GPT models, Llama models, Flan models, etc.

- For ease of implementation, developing the experiment so that very little changes need to be made when switching between training and generating text with both models.

To build off the experiments from both Shumailov et al.[24] and Guo et al.[9] we decided to continue to use OPT-350M [34] as Model-1. This model is open-source and easily accessible through the Transformers library provided by HuggingFace [30].

The reason we chose to have OPT-350M act as Model-1 instead of Model-2 was so we could potentially better compare our results to that of Guo et al.[9] if we had the opportunity to expand beyond very early stage of model collapse we are evaluating in this experiment. If we were able to train and re-train the models multiple times to evaluate cross-model training for a later of stage model collapse, we could compare the

results to models experiencing later stage model collapse that were tested in previous research.

For Model-2 we wanted an open-source model also available through the Transformers library, one that is well-suited for fine-tuning research purposes. For that reason, we chose Llama-2, which is a commonly used open-source LLM which is often used in research applications [26]. We chose the smallest size of Llama-2 available, Llama-2-7B with 7 billion parameters to be comparable in size to OPT-350M. Additionally, a larger model would be difficult with time and resource constraints.

3.1.2 Dataset and Task

For the experiment we chose a single natural language processing (NLP) task. Guo et al.[9] has previously shown that many NLP tasks are affected by model collapse. They stated that "higher entropy" tasks, which essentially require more "creativity" from the model, will be more greatly affected by model collapse. We chose a task that required a smaller number of tokens to be generated. This would provide enough text to measure the linguistic diversity, while also reducing generation time for the large number of required samples.

Text summarization has a low entropy, according to Guo et al.[9] This will allow the model to save time and computing resources as the output for each sample in the dataset will only require less tokens per generation/sample in the dataset. Additionally, there are existing metrics well suited to measure the accuracy of text summarization that we can use to further evaluate the model.

When determining what dataset to use for the experiment, we wanted a well-curated large dataset which is meant to ensure high-quality data that would act as a non-confounding variable in the experiment. We chose XL-Sum, a large dataset comprised of abstractive summaries of articles in 44 languages [10]. This dataset was

used for the "Text Summarization" task from Guo et al.[9]. It was designed to provide high quality and human-curated summaries for training and research purposes. Other similar large datasets sourced a lot of their data from places that weren't guaranteed to have been written by humans. For instance, Guo et al. [9] used a story generation dataset that was sourced from Reddit, which is hard to authenticate the responses and verify the quality. XLSUM has already done this.

The dataset is already formatted and maintained on HuggingFace, making it easily accessible through the Dataset library [19]. This also has the added benefit of being designed to integrate with other components of the HuggingFace ecosystem we used in our code.

We selected a random sample of the English portion of the dataset (5%), which included approximately fifteen thousand samples for fine-tuning. This still provided a satisfactory training context while significantly cutting down on both the training and generation steps. A recent study has shown that having a small amount of quality samples provided better results than many samples that were low quality [35], so we were confident that using a portion of XLSUM would not negatively affect fine-tuning the models.

3.1.3 Model Training

We implemented instruction tuning to fine-tune the models [33]. Instruction-tuning utilizes supervised learning to train a model to perform a specific task by providing instruction-output pairs to show the model what its expected output should be. This is useful as we are only training our model to summarize text, and this specific task is an ideal use-case of instruction tuning. We gave the model the Instruction: "Summarize the following text.", followed by the article to summarize, and the summary

associated with the article. This was done using HuggingFace's TRL library[1] [28], which provides a straightforward method for automatically formatting training data for instruction tuning. We trained the models for 5 epochs.

When fine-tuning the models, we wanted to optimize, where possible, to cut down on running time. For that reason, we chose to use low rank adaptation (LoRA) to fine-tune the model [11]. This allowed us to specialize the model for text summarization without having to adjust the pre-trained model weights. LoRA greatly reduces the number of trainable parameters and therefore reduces the time spent fine-tuning, without having to sacrifice model quality. We used the PEFT (Parameter Efficient Fine Tuning) library provided by HuggingFace[2] [21].

In addition to LoRA we used the AdamW-paged-32-bit optimizer, which is an implementation of the AdamW [20] that is designed to transfer paged memory of the optimizer state to CPU when GPU memory is exhausted. This is useful as we will only be using a single RTX 4090 GPU.

When training Llama-2, we needed to quantize the model to 8-bit to fit on a single GPU [7]. We did this using the built-in 8-bit quantizer available through bitsandbytes library[3] , which provides implementation for optimizations that integrate with other HuggingFace libraries.

We used the cross-entropy loss function that is the default loss function for HuggingFace language modeling to optimize the model when training. The goal of the experiment is not to optimize the model to be more diverse to avoid model collapse, but instead to train the model like we would "normally" for the task we chose, in this case text summarization, and evaluate the diversity of the responses from there.

[1] https://github.com/huggingface/trl
[2] https://github.com/huggingface/peft
[3] https://github.com/TimDettmers/bitsandbytes

3.1.4 Text Generation: Inference Strategy

When generating text for both Model-1 and Model-2, we limited the number of new tokens generated to 50 in line with Guo et al.[9]. We used top_p (nucleus) sampling set to 0.9 to slightly limit the token included in responses, and a temperature of 0.7 to create slightly more diverse responses.

For OPT-350M (Model-1), we used a 3-beam search when generating the model's responses. This improved the quality of the responses while still generating at a speed that remained within time constraints.

For Llama-2 (Model-2), we used a different decoding approach due to time constraints. Because of the large parameter size of Llama-2, we took a combination of approaches to speed up the time taken to generate responses. The model is quantized to 8-bit to fit on the GPU and increase inference speed. Secondly, we used Flash Attention-2 [6], which improves upon Flash Attention-1 and increases inference speed. The final way we increased speed was using n-gram based assisted decoding[4], which builds on the concept of speculative decoding [15]. This strategy greatly increased inference speeds, cutting generation time in half for Llama-2. It works on the assumption that many tokens in the prompt will appear in the output, which is well suited for text summarization. This tells the model that tokens that appear in the article should be higher priority in the summary. With these optimizations, single GPU inference was possible and more efficient.

We applied post-processing to the text that was returned by models. This meant removing incomplete sentences or extraneous text segments as seen in Figure 3.3. Only what the model considered the summary of the text was included. If the summary ended before the 50 token limit was reached the model would search for what

[4] https://github.com/apoorvumang/prompt-lookup-decoding

to include until the 50 token limit was reached. In practice, occasionally the article for the summary would be re-printed, or the summary itself would repeat.

3.1.5 Evaluation Metrics

The levels of both linguistic diversity and accuracy can be compared between all the datasets (Dataset-0, Dataset-Model-1, Dataset-Model-2) to see the effects of both the percentage of human training data and cross-model training.

Guo et al. [9] proposed a series of novel metrics to measure the linguistic diversity of generated text. The three categories of metrics determine lexical, semantic, and syntactic diversity.

We used preexisting metrics to measure lexical diversity, such as Type-Token Ratio (TTR) [12, 25], Distinct-N [16], and SELF-BLEU [17]. However, as Guo et al. [9] state, only looking at lexical diversity is not sufficient to measure the linguistic diversity of a text. For that reason, we also use metric measure syntactic (structure/-grammar), and semantic (meaning) diversity.

To measure syntactic diversity, Guo et al. [9] propose creating graphs using Universal Dependencies formalism to represent the structure of the text, then vectorize the graphs so that graphs closer to each in the embedding space are more alike. Then calculate the average pairwise distance (D_syn_p) or the average centroid distance (D_syn_c) among all graph embeddings.

Similarly, to measure semantic diversity, Guo et al. [9] propose creating sentence embedding vectors that are dispersed across semantic space. Then calculate the average pairwise cosine-distance (D_sem_p) and the mean cosine distance of each embedding vector to the centroid (D_sem_c).

In addition to the metrics proposed by Guo et al. [9] we added several metrics to the measure model accuracy. These metrics include perplexity which was calculated

from eval_loss, ROUGE [18], BLEU [22], and an F1 score, which ROUGE acts as recall and BLEU acts as precision.

Figure 3.4 shows the specifications chosen for the experiment, referencing the specific models and dataset chosen, as well as the evaluation metrics chosen.

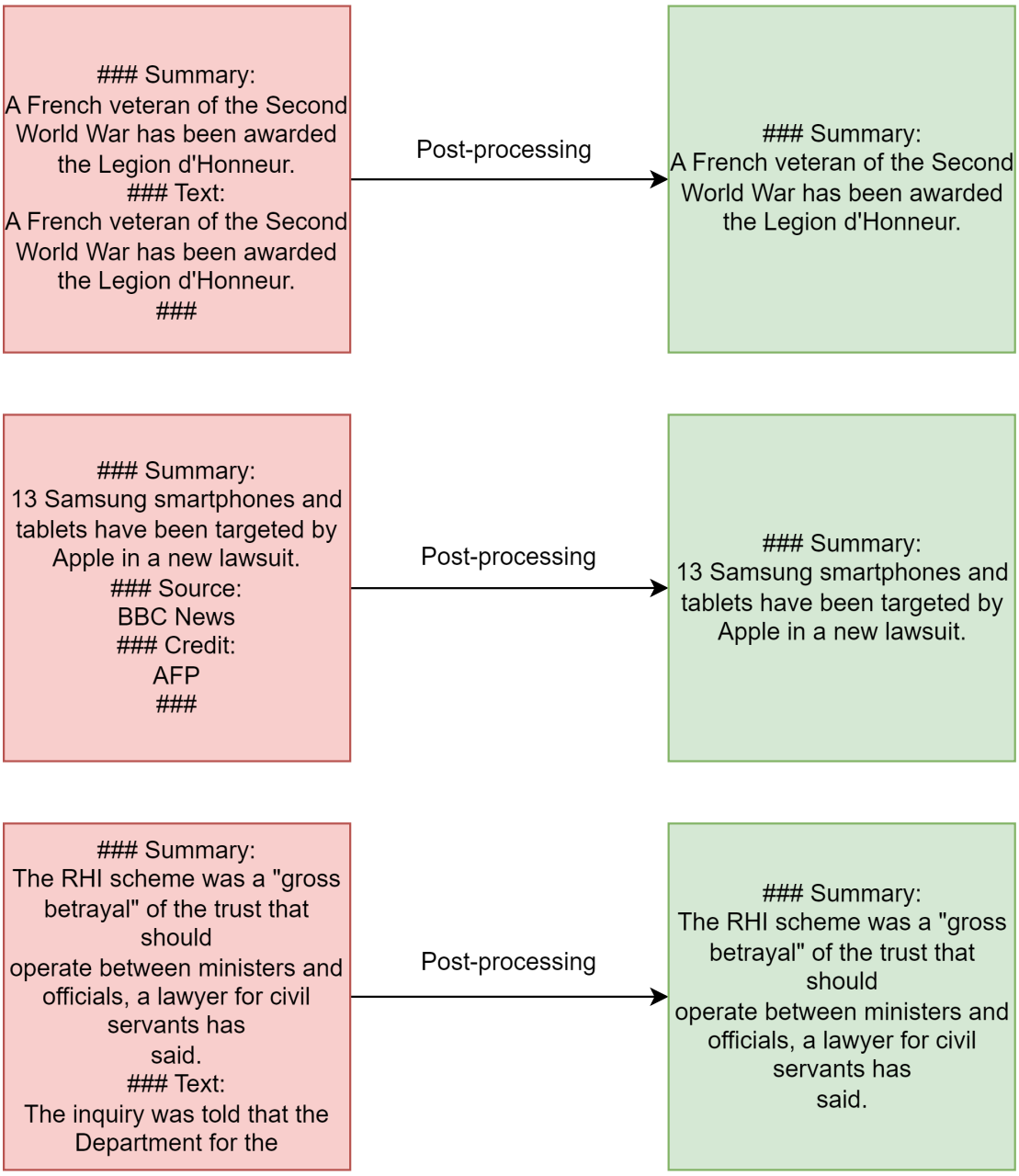

Figure 3.3: Examples of extraneous text returned by the models if the summary completed before the 50 token generation limit was reached. Before analyzing the text this response was post-processed to remove this extraneous text.

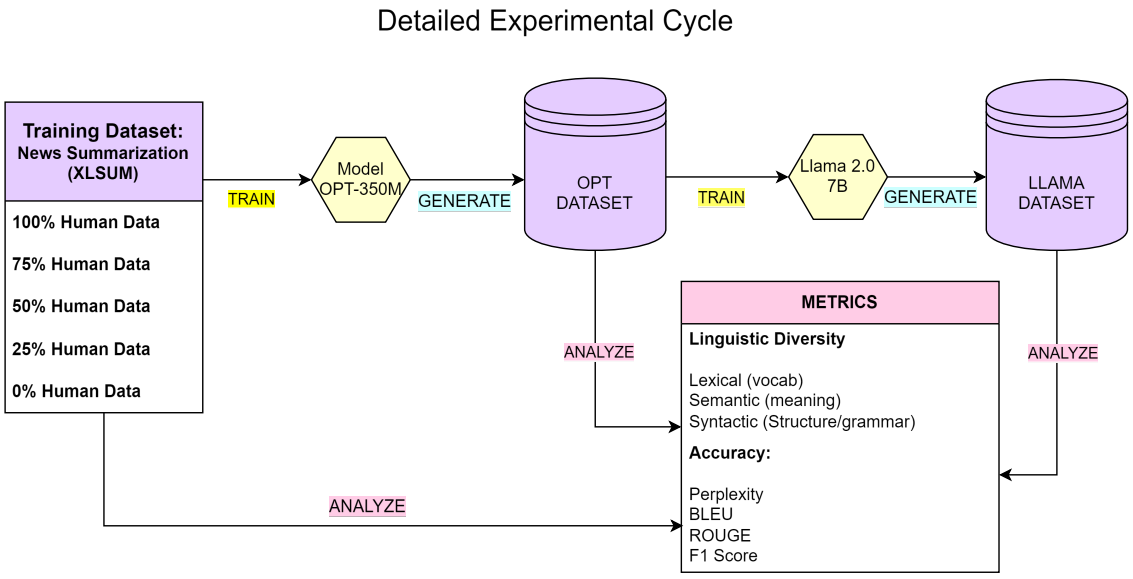

Figure 3.4: In-depth diagram of the experimental design. This diagram references specific experimental elements such as the models and dataset chosen. It shows each step of the experiment before data analysis.

Chapter 4

Results and Discussion

4.1 Results

We can see from Table 4.1 that the self-consuming Llama-2 models have higher diversity scores for almost all cycles and metrics compared to Llama-2 cross-model trained on the output of OPT-350M. The only type of diversity that the cross-model trained models performed better than either human data for self-consuming models was semantic diversity, where it scored higher for all percentages. However, for most cycles,

		Syntactic		**Semantic**			**Lexical**		
	Perplexity	D_syn_p	D_syn_c	D_sem_p	D_sem_c	Self-BLEU	TTR	Distinct-2	Distinct-3
100 Percent Human	–	10.09	7.169	1.359	0.962	**0.556**	0.061	**0.442**	**0.783**
Llama-2 Trained on Self	5.55	9.705	6.902	1.354	0.958	0.487	0.064	0.394	0.717
Llama-2 Trained on OPT-350M	5.46	9.365	6.69	**1.364**	**0.965**	0.38	0.061	0.317	0.58
75 Percent Human	–	**10.101**	**7.187**	1.36	0.962	0.54	0.062	0.435	0.772
Llama-2 Trained on Self	5.53	9.833	6.992	1.349	0.955	0.47	**0.066**	0.383	0.687
Llama-2 Trained on OPT-350M	5.47	9.362	6.697	1.363	0.965	0.393	0.065	0.337	0.601
50 Percent Human	–	9.886	7.034	1.359	0.962	0.524	0.063	0.425	0.759
Llama-2 Trained on Self	5.51	9.666	6.871	1.348	0.954	0.456	0.065	0.375	0.68
Llama-2 Trained on OPT-350M	5.47	9.133	6.523	1.363	0.965	0.388	0.065	0.33	0.591
25 Percent Human	–	9.873	7.014	1.358	0.961	0.507	0.063	0.412	0.741
Llama-2 Trained on Self	5.48	9.633	6.847	1.347	0.954	0.453	0.064	0.373	0.677
Llama-2 Trained on OPT-350M	5.47	9.38	6.71	1.364	0.965	0.392	0.065	0.337	0.601
0 Percent Human	–	9.77	6.956	1.355	0.959	0.491	0.064	0.394	0.717
Llama-2 Trained on Self	**5.45**	10.006	7.112	1.348	0.954	0.457	0.065	0.374	0.679
Llama-2 Trained on OPT-350M	5.46	9.413	6.727	1.362	0.964	0.395	0.064	0.337	0.603

Table 4.1: This table shows the scores for all the diversity metrics plus perplexity for all baseline/control cycles of the experiment, where each cycle had 25% less human data in the training set for the model. In this table we compare the original human data combined with different percentages of generated data from Llama-2, output from Llama-2 which was trained on this combined data, and output from Llama-2 which was trained on the output of OPT-350. The best scores for every metric are bolded.

the self-consuming models scored higher than both human data and cross-trained model data for all other metrics.

When comparing human data, OPT-350M data, and cross-model trained Llama-2 data in Table 4.2, we can see that a lower percentage of human data in training data lead to lower linguistic diversity scores overall for each cycle. However, Llama-2 models overall still scored lower or the same for almost all diversity metrics. The only exception to this is that Llama-2 scored higher than OPT-350M and human data for the 0% human data cycle.

Table 4.1 and Table 4.2 display the perplexity of the models trained in addition to the linguistic diversity metrics proposed by Guo et al. [9]. We can see that adding synthetic data produced by the model to to the model's own training set has decreased the perplexity of the model. This is true both when OPT-350M was trained on it's own output, and when Llama-2 was trained on its own output. However, the perplexity of the Llama-2 models trained on the output of OPT-350M stayed the same with a variation of about 0.01 across all percentages of human data.

		Syntactic		**Semantic**			**Lexical**		
	Perplexity	D_syn_p	D_syn_c	D_sem_p	D_sem_c	Self-BLEU	TTR	Distinct-2	Distinct-3
100 Percent Human	–	**9.999**	**7.098**	1.36	0.962	**0.552**	0.061	**0.442**	**0.783**
OPT-350M	11.55	9.341 (-0.066)	6.669 (-0.061)	1.362 (0.001)	0.964 (0.001)	0.388 (-0.297)	0.06 (-0.006)	0.32 (-0.277)	0.592 (-0.244)
Llama-2	5.46	9.365 (0.003)	6.69(0.003)	**1.364 (0.001)**	**0.965 (0.001)**	0.38 (-0.022)	0.061(0.007)	0.317 (-0.007)	0.58 (-0.019)
75 Percent Human	–	9.99	7.104	1.361	0.963	0.52	0.062	0.424	0.752
OPT-350M	11.26	9.376 (-0.061)	6.701 (-0.057)	1.362 (0.001)	0.964 (0.001)	0.414 (-0.205)	**0.066 (0.063)**	0.343 (-0.189)	0.617 (-0.179)
Llama-2	5.47	9.362 (-0.002)	6.697 (-0.001)	1.363(0.001)	0.965(0.001)	0.393 (-0.05)	0.065 (-0.005)	0.337 (-0.02)	0.601 (-0.027)
50 Percent Human	–	9.688	6.904	1.362	0.964	0.472	0.062	0.398	0.711
OPT-350M	11.26	9.4 (-0.03)	6.722 (-0.026)	1.364 (0.001)	0.965 (0.001)	0.406 (-0.141)	0.065(0.05)	0.343 (-0.139)	0.617 (-0.132)
Llama-2	5.47	9.133 (-0.028)	6.523(-0.03)	1.363 (0)	0.965(0)	0.388 (-0.045)	0.065 (-0.008)	0.33 (-0.037)	0.591 (-0.042)
25 Percent Human	–	9.465	6.752	1.364	0.965	0.438	0.062	0.366	0.659
OPT-350M	11.1	9.406 (-0.006)	6.727 (-0.004)	1.363 (0)	0.965 (0)	0.409 (-0.065)	0.066 (0.055)	0.344 (-0.061)	0.617 (-0.064)
Llama-2	5.47	9.38 (-0.003)	6.71(-0.003)	1.364 (0)	0.965(0)	0.392 (-0.042)	0.065 (-0.004)	0.337 (-0.02)	0.601 (-0.026)
0 Percent Human	–	9.273	6.623	1.363	0.965	0.382	0.06	0.32	0.592
OPT-350M	10.92	9.159 (-0.012)	6.558 (-0.01)	1.363(0)	0.964 (0)	0.403(0.055)	0.064 (0.069)	0.339(0.06)	0.612 (0.034)
Llama-2	**5.46**	9.413(0.028)	6.727(0.026)	1.362 (-0.001)	0.964 (-0.001)	0.395 (-0.019)	0.064 (-0.003)	0.337 (-0.006)	0.603 (-0.014)

Table 4.2: This table shows the scores for all the diversity metrics plus perplexity for all cross-model training cycles of the experiment, where each cycle had 25% less human data in the training set for the model. In this table we compare the original human data combined with different percentages of generated data from OPT-350M, output from OPT-350M which was trained on this combined data, and output from Llama-2 which was trained on the output of OPT-350. The best scores for every metric are bolded.

We can see that there is an overall decrease from 100% to 0% human training data for all linguistic diversity metrics aside from the semantic diversity metrics, which experienced very minor changes for both D_sem_p and D_sem_c for all percentages. While there is a significant increase between the values 0% and 100%, there is a more gradual change between each percent group. For instance, between 100% to 75%, the rate of change is not as significant. Additionally, the rate change seems consistent across all percent groups. This indicates there is not an inflection point associated with a certain percentage of synthetic training data the model will experience a sharp decrease in diversity.

The metrics that seem to be most greatly affected by the addition of synthetic training data were the syntactic diversity metrics and Distinct-N, where the greater the N value the more variation there was across the percent groups. This could indicate these metrics are more sensitive to model collapse compared to the semantic diversity metrics which does very sensitive, specifically when looking at early stage model collapse as we did in this experiment.

Interestingly, the semantic diversity scores for Llama-2 models cross-model trained on the output of OPT-350M where higher than the scores of Llama-2 models trained on their own output.

Table 4.4 shows that the ROUGE, BLEU, and F1 scores all displayed a similar trend to that seen by the perplexity scores. The ROUGE, BLEU, and F1 scores improved with more synthetic data added, appearing to show that the model was performing better. This was shown for the models trained on their own output: OPT-350M models in 4.4 and the Llama-2 models in 4.3, as well as the Llama-2 cross-trained models in 4.4. However, this is contradicted by the decreasing in the linguistic diversity metrics.

	ROUGE	BLEU	F1
100 Percent Human	0.325	0.076	0.062
75 Percent Human	0.349	0.099	0.077
50 Percent Human	0.38	0.131	0.097
25 Percent Human	0.403	0.157	0.113
0 Percent Human	0.428	0.185	0.37

Table 4.3: This table shows the performance metrics that were measured in addition to linguistic diversity metrics. This table shows the performance scores for the Llama-2 models that self-consumed their own output.

	Model	ROUGE	BLEU	F1
100 Percent Human	OPT-350M	0.298	0.052	0.044
	Llama-2	0.469	0.221	0.15
75 Percent Human	OPT-350M	0.351	0.108	0.083
	Llama-2	0.509	0.278	0.18
50 Percent Human	OPT-350M	0.419	0.174	0.123
	Llama-2	0.508	0.271	0.177
25 Percent Human	OPT-350M	0.493	0.257	0.169
	Llama-2	0.51	0.275	0.179
0 Percent Human	OPT-350M	0.568	0.353	0.218
	Llama-2	0.511	0.281	0.181

Table 4.4: This table shows the performance metrics that were measured in addition to linguistic diversity metrics. This table compares the performance scores for the OPT-350M models and the Llama-2 models which were cross-model trained on the output of OPT-350M.

4.1.1 Discussion

Our results reaffirm the conclusion reached by Guo et al. [9] that there was a need for new metrics to specifically measure linguistic diversity in AI generated text. We noticed that perplexity and other existing performance metrics that we measured were insufficient to determine if a model is experiencing model collapse. While the change across cycles was small, the perplexity did decrease overall when a model was trained on more AIGC, which would indicate that a model was more accurate. However, this is deceiving because we known based on the diversity metrics that the diversity in the text is also decreasing. This shows that perplexity as a measure of accuracy is not

enough on it's own to paint the full picture of the quality of the model in the context of model collapse.

We see similar results when looking at other model performance metrics. ROUGE, BLEU, and F1 scores are increased as more synthetic training material was added to the models' training sets. Without interpreting this in the context of model collapse these scores would indicate that the models are performing better. However, we can see that with the increases in performance metrics comes a marked decrease in the diversity of the generated text.

In evaluating the Llama-2 models that were cross-model trained using the output of OPT-350M and the Llama-2 models trained on their own output, we can see that for many metrics the models trained their on their own outputs performed better. With the exception of semantic diversity, there is a notable discrepancy between the cross-trained models and the model trained on their own output.

This would indicate that not only is cross-training not a sufficient solution to avoiding model collapse, cross-training can decrease the lexical and semantic diversity compared to models trained on their own output.

Semantic diversity improved with improved with cross-model. However it did not change or barely decreased when the model was only train on its own output indicates that semantic diversity might not be as susceptible to model collapse compared to syntactic and lexical diversity. Considering that semantics is meant to measure the meaning of words, this could indicate that models might retain an understanding of word meaning even while experiencing early stage model collapse.

We were interested in determining if there was a significant inflection point in the data that showed that a certain percentage of synthetic data in the training set would lead to a sharp decrease in diversity. This experiment did not show a sharp inflection point, but instead showed gradual decrease in diversity through the different

percentage groups.

For that reason, to avoid model collapse we recommend guaranteeing a "majority" of non-synthetic, human generated data in the training set. The results of this experiment show that for early stage model collapse, the more human data you can guarantee is in your training data the less loss of diversity you will experience. It would be ideal to guarantee 100% of the training data is human-generated, and it would be detrimental to not guarantee that any of it is human-generated. Therefore we propose the compromise that guaranteeing a majority of the training data is human-generated. In practice this could look like using more corpora of books/media which is guaranteed to be written by humans, or internet data from before the widespread use of generative AI in the early 2020's. This would be proportional to the amount of more recent internet data that is included from sources such as Common Crawl.

While this is not an ideal solution to model collapse in the long run as over time the AIGC on the internet will increase and perhaps outpace human generated data available, it is an actionable solution that can be taken currently to prevent model collapse.

Chapter 5

Conclusions and Future Works

5.1 Conclusions

We have provided actionable steps that model developers can take today to slow down the rate of model collapse. This can be done by guaranteeing that the majority of training data in the training set is human-generated, which can slow down the rate of model collapse and loss of diversity in the model's output. This data can include things like large corpora of books, and internet data from before the widespread use of model collapse.

These steps are desirable because they don't require changes to the architecture of large models, which are costly. Instead, we suggest that in addition to large amounts of internet content sourced from places like Common Crawl, one can also include large datasets that have guaranteed non-synthetic material, like large corpora of books or other media. If these datasets were developed before the wide-spread use of generative AIs, it would diminish the risk of reaching a later stage of model collapse and compromising your model.

This solution still provides the opportunity for large models to learn about current/up-to-date information from the internet with a lessened risk of model collapse, though with this approach the internet data will now make up the minority of training data. Many of these corpora are also open-source and are cheaper than web-scraped

datasets.

Additionally, we have determined that cross-model training with a model of a different architecture/model "family" is not a sufficient solution to preventing early stage. Within the scope of this experiment, self-consuming models performed better overall than cross-trained models.

5.1.1 Future Work

In the future we would be interested in expanding the experiment in various ways to gain a deeper understanding of the effect of models trained on other output. To expand this experiment we would like to train for more epochs than 5 to see if this affects the diversity of the models. We would also like to develop a way to optimize for linguistic diversity when training models as a way to prevent model collapse.

We would like to test multiple instances of training and re-training the models on AIGC to simulate late stage model collapse. This experiment was limited in that we only only aimed to test early stage model collapse. We only tested the models being trained on AIGC once and generating output instead of training and re-training on AIGC. In the future we would like to evaluate if cross-model training has an impact on late-stage model collapse that we did not see for early stage mode collapse.

We would want to test more tasks, as this experiment only tested a single task: text summarization. We would like to determine if certain tasks increase or decrease have an affect on how cross-model training affects the models. We would also like to replicate the experiment with other models, including the output of multiple models in the training set instead of one, to see if this lessens or exacerbates the effects of cross-model training.

In the future we would also be interested in furthering existing research on measuring linguistic diversity but developing a metrics to measure prosody in LLM generated

texts. Prosody refers to the stress and intonation of words and how that affects their meanings.

BIBLIOGRAPHY

[1] Akram, A. An empirical study of ai generated text detection tools. *Advances in Machine Learning & Artificial Intelligence* (9 2023). Orignality best AI detection tool they tested. Noticied many false positives.

[2] Alemohammad, S., Casco-Rodriguez, J., Luzi, L., Humayun, A. I., Babaei, H., LeJeune, D., Siahkoohi, A., and Baraniuk, R. G. Self-consuming generative models go mad.

[3] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. *Advances in Neural Information Processing Systems 2020-December* (5 2020).

[4] Chakraborty, S., Bedi, A. S., Zhu, S., An, B., Manocha, D., and Huang, F. On the possibilities of ai-generated text detection.

[5] Dalalah, D., and Dalalah, O. M. The false positives and false negatives of generative ai detection tools in education and academic research: The case of chatgpt. *The International Journal of Management Education 21* (7 2023), 100822.

[6] Dao, T. Flashattention-2: Faster attention with better parallelism and work partitioning.

[7] Dettmers, T., Lewis, M., Shleifer, S., and Zettlemoyer, L. 8-bit optimizers via block-wise quantization.

[8] Elkhatat, A. M., Elsaid, K., and Almeer, S. Evaluating the efficacy of ai content detection tools in differentiating between human and ai-generated text. *International Journal for Educational Integrity 19* (9 2023), 17.

[9] Guo, Y., Shang, G., Vazirgiannis, M., and Clavel, C. The curious decline of linguistic diversity: Training language models on synthetic text.

[10] Hasan, T., Bhattacharjee, A., Islam, M. S., Samin, K., Li, Y. F., Kang, Y. B., Rahman, M. S., and Shahriyar, R. Xl-sum: Large-scale multilingual abstractive summarization for 44 languages. *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021* (2021), 4693–4703.

[11] Hu, E., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. Lora: Low-rank adaptation of large language models.

[12] Johnson, W. I. a program of research. *Psychological Monographs 56* (1944), 1–15.

[13] Krishna, K., Song, Y., Karpinska, M., Wieting, J., and Iyyer, M. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense.

[14] Kumar, R., and Mindzak, M. Who wrote this? detecting artificial intelligence-generated text from human-written text. *Canadian Perspectives on Academic Integrity 7* (1 2024), 1.

[15] Leviathan, Y., Kalman, M., and Matias, Y. Fast inference from transformers via speculative decoding, 7 2023.

[16] Li, J., Galley, M., Brockett, C., Gao, J., and Dolan, B. A diversity-promoting objective function for neural conversation models. *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016 - Proceedings of the Conference* (2016), 110–119.

[17] Liang, P., Bommasani, R., Lee, T., Tsipras, D., Soylu, D., Yasunaga, M., Zhang, Y., Narayanan, D., Wu, Y., Kumar, A., Newman, B., Yuan, B., Yan, B., Zhang, C., Cosgrove, C., Manning, C. D., RÃ©, C., Acosta-Navas, D., Hudson, D. A., Zelikman, E., Durmus, E., Ladhak, F., Rong, F., Ren, H., Yao, H., Wang, J., Santhanam, K., Orr, L., Zheng, L., Yuksekgonul, M., Suzgun, M., Kim, N., Guha, N., Chatterji, N., Khattab, O., Henderson, P., Huang, Q., Chi, R., Xie, S. M., Santurkar, S., Ganguli, S., Hashimoto, T., Icard, T., Zhang, T., Chaudhary, V., Wang, W., Li, X., Mai, Y., Zhang, Y., and Koreeda, Y. Holistic evaluation of language models.

[18] Lin, C. Y. Rouge: A package for automatic evaluation of summaries. *Proceedings of the workshop on text summarization branches out (WAS 2004)* (2004).

[19] Liu, Y., Cao, J., Liu, C., Ding, K., and Jin, L. Datasets for large language models: A comprehensive survey.

[20] Loshchilov, I., and Hutter, F. Decoupled weight decay regularization. *7th International Conference on Learning Representations, ICLR 2019* (11 2017).

[21] Mangrulkar, S., Gugger, S., Debut, L., Belkada, Y., Paul, S., and Bossan, B. Peft: State-of-the-art parameter-efficient fine-tuning methods. https://github.com/huggingface/peft, 2022.

[22] Papineni, K., Roukos, S., Ward, T., and Zhu, W. J. Bleu: A method for automatic evaluation of machine translation. vol. 2002-July.

[23] Sadasivan, V. S., Kumar, A., Balasubramanian, S., Wang, W., and Feizi, S. Can ai-generated text be reliably detected?

[24] Shumailov, I., Shumaylov, Z., Zhao, Y., Gal, Y., Papernot, N., and Anderson, R. The curse of recursion: Training on generated data makes models forget.

[25] Templin, M. C. *Certain language skills in children; their development and interrelationships.* University of Minnesota Press, 1957.

[26] Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., RoziÃ¨re, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., and Lample, G. Llama: Open and efficient foundation language models.

[27] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. *Advances in Neural Information Processing Systems 2017-December* (6 2017), 5999–6009.

[28] von Werra, L., Belkada, Y., Tunstall, L., Beeching, E., Thrush, T., Lambert, N., and Huang, S. Trl: Transformer reinforcement learning. https://github.com/huggingface/trl, 2020.

[29] Walters, W. H. The effectiveness of software designed to detect ai-generated writing: A comparison of 16 ai text detectors. *Open Information Science 7* (1 2023).

[30] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., Platen, P. V., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M.

Transformers: State-of-the-art natural language processing. *EMNLP 2020 - Conference on Empirical Methods in Natural Language Processing, Proceedings of Systems Demonstrations* (2020), 38–45.

[31] Yang, K.-C., and Menczer, F. Anatomy of an ai-powered malicious social botnet.

[32] Zhang, H., Edelman, B. L., Francati, D., Venturi, D., Ateniese, G., and Barak, B. Watermarks in the sand: Impossibility of strong watermarking for generative models.

[33] Zhang, S., Dong, L., Li, X., Zhang, S., Sun, X., Wang, S., Li, J., Hu, R., Zhang, T., Wu, F., and Wang, G. Instruction tuning for large language models: A survey.

[34] Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, V., Mihaylov, T., Ott, M., Shleifer, S., Shuster, K., Simig, D., Koura, S., Sridhar, A., Wang, T., Zettlemoyer, L., and Ai, M. Opt: Open pre-trained transformer language models.

[35] Zhou, C., Liu, P., Xu, P., Iyer, S., Sun, J., Mao, Y., Ma, X., Efrat, A., Yu, P., Yu, L., Zhang, S., Ghosh, G., Lewis, M., Zettlemoyer, L., and Levy, O. Lima: Less is more for alignment.