Should ChatGPT and Bard Share Revenue with Their Data Providers? A New Business Model for the AI Era

Dong Zhang [1]

***Abstract*—With various AI tools becoming increasingly convenient and popular, we are ushering in a true AI era.**

CONTENTS

I. INTRODUCTION

ChatGPT (Chat Generative Pre-trained Transformer) has recently become highly popular, and its popularity is continuing to rise. It is a class of advanced machine learning Natural Language Processing models developed by OpenAI, using both supervised machine learning and reinforcement learning techniques. ChatGPT was initially built on top of OpenAI's GPT-3 family, then GPT-4, designed to answer questions, generate text, summarize information, and perform other language-related tasks. It has the capability to understand and respond to various types of inputs, making it a powerful tool for conversational AI applications.

As a large language model (LLM), ChatGPT is trained on a large and diverse dataset of text obtained from various sources on the internet, but the specific sources and details of the training data are still not publicly available. Despite this, we can still gain some understanding from its previous version GPT-3 [1]. As of 2020, GPT-3 used a broad range of data sources, including the CommonCrawl data, WebText, two internet-based books corpora, as well as English-language Wikipedia. The CommonCrawl data contains 410 billion tokens originally from billions of web pages collected by the CommonCrawl Foundation. The WebText2 dataset, which comes from high quality webpages scraped from Reddit links, contains over 49 billon tokens. The two book sources include about 67 billion tokens, while Wikipedia contributes at least 3 billion tokens. GPT-3 API was opened to the public in 2021, and InstructGPT, which used human feedback and reinforcement learning to fine-tune GPT-3, was released in 2021 and upgraded in 2022 [2]. ChatGPT, which is a fine-tuned version of a model from the GPT-3.5 series and a sibling model to InstructGPT, trained to respond to a prompt with a detailed response.

ChatGPT was made available for public testing on November 30th, 2022. Just one week after the launch, Sam Altman, CEO of OpenAI announced on Twitter that ChatGPT has crossed one million users. It was reported that ChatGPT has more than 100 million users with the first two months of its launch, and has more than 13 million daily visitors as of 2023. On February 1, OpenAI announced that it is rolling out ChatGPT Plus pilot plan, a premium subscription service for $20 per month, only in the US marketplace, to have faster ChatGPT access and response times, as well as priority access to new ChatGPT features and improvements. On March 14, OpenAI launched GPT-4, which limited version became public available in ChatGPT Plus. OpenAI predicts that ChatGPT will generate $200 million in revenue by the end of 2023, and $ 1 billion in 2024. Of course OpenAI will distribute its revenue from ChatGPT to its stakeholders. For example, Microsoft will reap significant profits from OpenAI, since Microsoft Azure has supported OpenAI and ChatGPT model training, and it has invested $10 billion in OpenAI[1]. It was reported that Microsoft may gain 75% share of OpenAI's profits until it makes back the money on its investment, after which 49% of stake ownership of it[2]. Meanwhile, Google also launched its

[1] Amazon , dongzhn@amazon.com

[1]See report by Reuters.
[2]Sources include CNBC and Reuters.

own chatbot Bard in March 2023, attempting to compete with ChatGPT.

But there is a question here: does OpenAI only need to share its ChatGPT revenue and profits with its stakeholders, or does it also need to share them with all of its data providers? We know that OpenAI has expended a great deal of effort collecting the training dataset for its GPT families, and has consumed a significant amount of time developing and training the models, but the success of ChatGPT is largely due not only to its state-of-the-art models, but also to the huge amount of data it has collected and used for training. Wouldn't it be fair to acknowledge the contribution of all the providers of ChatGPT's different data sources to its success? It is worth noting that even if the data sources are in public domain, it is still common practice to give proper attribution when using them in one's work, so is ChatGPT. But how can an Artificial Intelligence tool weight the importance of its data providers and acknowledge them? In this paper, we will see that discussing these questions is necessary and essential for building a good AI ecosystem.

Same question for Bard and Google. Of course the above questions are not only for ChatGPT, or for all LLMs, but for almost all AI tools that are emerging. We have AI tools in various fields. For example, OpenAI also has a popular image generation tool called Dall-E 2. Although the training dataset of Dall-E 2 is still not yet public, we know that Dall-E 2 at least used Google's Conceptual Captions dataset, which has more than 3 million images, paired with natural-language captions. Should the commercialized Dall-E 2 share the revenue with Google, or even the original image creators from Conceptual Captions? Other AI image generators, such as MidJourney, Stability AI and DeviantArt, have detailed their compliance with Digital Millennium Copyright Act (DMCA) on their official websites, with the aim of avoiding any infringement of someone's intellectual property rights. But some artists have initiated lawsuits against these AI image generators because they believe that these companies have violated the DMCA[3]. For the upcoming AI era, is it better for AI tools to strictly adhere to existing copyright laws such as DMCA, or is sharing revenue with data providers such as artists a win-win way to go? Let us take another look at the field of healthcare. Medical imaging is another area that utilizes AI tools. Public and private medical imaging datasets are used to compare with images uploaded by customers on health platforms such as Google Cloud's Medical Imaging Suite or Amazon Web Service (AWS)'s HealthLake, so that customers can do modeling and data analyze for their own images. Can we create a new mechanism where health platforms share revenue with the providers of the medical imaging data? And this new mechanism may also encourage more medical institutions or personals to join and share their medical data, and eventually better improve the health AI tools.

Currently, there are multiple ways for companies to distribute revenue with their data and content providers. Some companies pay contributors through a royalty-based system, where contributors earn based on the usage of their content and the licensing agreement between them and companies. For example, Getty images contributors can earn 20% and 25% of consumers' payment for each image and video respectively, while iStock exclusive contributors can earn between 15% and 45% loyalties for each piece of content they license[4]. In the Web 2.0 era, a more sophisticated and internet way to share revenue is based on cost per action (CPA). Google has developed a revenue sharing mechanism through a platform called AdSense, which is an advertising platform that enables website owners to display targeted ads on their websites and earn money for every click on those ads. This sharing is based on some CPA models, which are popular today, but not good enough for the new coming AI era. It is very likely that we can have a more "AI way" to ask emerging big AI tech companies to share their revenue, and to encourage more people and organizations to participate in the AI era.

II. Why Sharing Revenue is a good idea

In the following discussion, I will often refer to ChatGPT. But I am not just targeting ChatGPT, which is just an example. The discussion in this paper is essentially applicable to all large AI models and tools.

A. Data vs. Models

We must know that various AI tool using deep learning models are not omnipotent, as they all require a large amount of data support. Without data, AI tools cannot make any progress. Peter Norvig, Google's research director once said, "We don't have better algorithms. We just have more data. More data beats clever algorithm, but better data beats more data." Although there is still debate about whether data or machine learning algorithms are more important for the coming AI era, it is undeniable that machine learning model performance highly depends on the quantity and quality of its training data. This can be seen from the evolution of GPT models. The first generation model, GPT-1 [3], was trained on the BooksCorpus dataset of 5 GB with 117 million parameters [4]. The second generation model, GPT-2 [5], used a 40 GB of WebText training dataset to feed a larger architecture with the number of parameters increasing to 1.5 billion. The following GPT-3 had a 45 TB training dataset across five different corpora including Common Crawl, WebText2, Books1, Books2 and Wikipedia, feeding an even larger architecture with 175 billion parameters [1]. The training dataset for ChatGPT has not been released, but comparable to GPT-3. If we consider that ChatGPT and other LLMs sparked an AI revolution, then this revolution is undoubtedly supported by massive amount of data.

B. Still need more and better data

But ChatGPT is not perfect. Contrary to what many people imagine, the current generation of ChatGPT has many

[3]Lawsuit against some AI image generators was reported by multiple media outlets, such as CBS News, The Verge, Bloomberg.

[4]The rates are from iStock and Getty Images Contributor Earnings.

limitations. One of the major problems is that many of the answers it gives are incorrect. ChatGPT itself acknowledged that it "sometimes writes plausible-sounding but incorrect or nonsensical answers". The question and answer website Stack Overflow banned the use of ChatGPT for generating answers to questions, since ChatGPT "produces have a high rate of being incorrect"[5]. ChatGPT also generates fake scientific abstracts and research papers[6], making the academic community very cautious about using it. Not only ChatGPT, but also other advanced chatbots made errors. Meta's Galactica model [6], which was trained on 48 million examples of a variety of sources, was offline after experts found it to be biased and generating false information. Microsoft's GPT-powered new Bing made mistakes in demo, and Bard, Google's ChatGPT competitor, Bard, gave wrong answer related to James Webb Space Telescope in an initial promotional demo.

One way to address the limitations of today's chatbots is to restrict their use cases, such as making ChatGPT only an auxiliary tool for writing, drafting documents, summarizing meeting records and giving limited recommendation. But this is clearly not what we want to see in the coming AI era, because even today's digital assistant tools such as Siri and Alexa can provide more information services. The ChatGPT website discusses the reasons for its incorrect answers, in addition to model-related reasons, a major factor is the lack of ground truth data. Obviously, the next generation of ChatGPT needs more and better data to train a better version. Here, **"more and better data"** not only means having a large training dataset for the model, just as some paper discussed [7], [8], but also has at least four manifold meanings:

1. Enhancing the diversity and amount of training data can improve models' learning ability and decrease the chances of errors.
2. Using data specific to a certain task of domain knowledge can fine tune the model for the task or domain.
3. Many models, such as ChatGPT, are using outdated data. It is crucial to keep the data up-to-date to produce more precise and recent results.
4. Incorporating more human feedback to establish better ground truth. So far chatGPT has done better job to collect human feedback than other LLMs.

So far, all the data used by large AI tools such as ChatGPT comes from publicly available sources (there have been some lawsuits but the situation is tricky). However, not all important data is publicly available, and ChatGPT, as well as any AI tool that wants to be sufficient intelligent, must have the willingness and methods to purchase data from diverse sources. But how can an AI tool purchase data outside of the public domain? Before discussing this topic, let us take a look at data ownership and copyrights.

C. Data privacy and copyright

Can ChatGPT use recent articles from New York Times, Wall Street Journal or BBC for training? I am afraid not. Most of the articles from these media outlets are not legally available for use without permission. The robots and crawlers policies of many media do not allow for free scraping of their content. Although whether media outlets used for training are not publicly disclosed by OpenAI, but there are now some concerns that ChatGPT may have used copyrighted materials from certain media outlets[7]. If that is the case, ChatGPT may face legal challenges, just like some AI image generators mentioned in Section I. This is certainly something that ChatGPT wants to avoid as much as possible. Since it is too expensive to use traditional way to purchase copyrighted data, this has resulted in many important news knowledge being excluded from the current training corpus of ChatGPT. Other AI tools also face similar issues.

Moreover, books are invaluable for long-range context modeling and coherent storytelling. ChatGTP has also used a huge number of books for training, but book selection is limited to public domain only. We know that the Project Gutenberg is a library of over 60,000 free books with more than 10 GB data and 3 billion tokens, GPT-3 used Books1 and Books2 datasets with a total of over 60 billion tokens. However, keep in mind there are ~ 130 million books published since the invention of Gutenberg's printing press in 1440 A.D.[8], a majority of books are copyrighted. Should ChatGPT use a traditional way to spend \$10 or \$20 to buy each book to feed the model? Absolutely not.

ChatGTP also uses Wikipedia and gives it a high weight in the training dataset to emphasize its importance. However, Wikipedia is not a reliable source for citations, as it states itself: "As a user-generated source, it can be edited by anyone at any time, and any information it contains at a particular time could be vandalism, a work in progress, or simply incorrect."[9] In contrast to Wikipedia, Encyclopedia Britannica is a well-known, centuries-old English-language encyclopedia that seems to have a reputation for scholarly authority. Why cannot ChatGPT use Encyclopedia Britannica or other encyclopedias as a more reliable source? Again, the reason is that other encyclopedias are copyrighted sources and may not be available for unrestricted use. Even if ChatGPT buys some encyclopedias, it still cannot freely display the knowledge from them to the public, according to current copyright laws.

To summarize, in the current situation, ChatGPT can only use public domain data for training, but it is obvious that public domain data is not enough. Just like a student needs to buy some textbooks, if ChatGPT wants to become an excellent student, it must spend money to purchase "textbooks", which are copyrighted or even private data. However, how can ChatGPT purchase copyrighted data and distribute it to the public

[5]See this Stack Overflow announcement for more details.

[6]See Forbes news Fake Scientific Abstracts Written By ChatGPT Fooled Scientists, Study Finds (10 Jan 2023) and Nature news Abstracts written by ChatGPT fool scientists (12 Jan 2023)

[7]For example, Bloomberg's article OpenAI Is Faulted by Media for Using Articles to Train ChatGPT (Feb 16 2023).

[8]This number was estimated by Google a decade ago.

[9]See Wikipeadia's own statement: Wikipedia is not a reliable source.

in the upcoming AI era is an unsolved problem. Obviously, traditional copyright laws are not applicable to the AI era. Commercialized ChatGPT and other AI tools cannot acquire data based on the old copyright laws.

Despite this, there is still a way for AI tools to obtain copyrighted data to improve its performance, while also providing benefits to data owners. Google's AdSense has given us some inspiration: a centralized platform created a project to connect to various websites, and distribute ads to these websites. The centralized platform gain revenue from advertisers, and share revenue with the websites that joined the project to distribute ads. This is one of the primary revenue streams for Google, and also one of the significant business models of the Web 2.0 era. The future AI era will be very different from Web 2.0, but the revenue-sharing model is still worth learning from. Whether ChatGPT's main revenue in the future comes from advertising, membership fees or other streams, it can share revenue with individuals and organizations that provide data to ChatGPT. This business model can also be adopted by other AI tools. Of course, we must design a new smart way to ensure that the centralized AI tools can share revenue fairly and transparently.

Before proposing some specific methods for revenue sharing, let us take a look at why it is a good idea for AI tools such as ChatGPT to share revenue with its data providers.

D. From AI war to AI-powered monopoly? Or revenue-shared utilitarian ecosystem?

ChatGPT is very popular, and perhaps Bard will catch up later, but we must recognize that ChatGPT and Bard are not the only LLM models available.

Currently, the competition among LLM models is intensifying. In addition to OpenAI's GPT families, other LLM include Google's PaLM [9], LaMDA [10], and Bard which is powered by LaMDA, Meta's OPT [11] and LLaMA [13], Amazon's AlexaTM [12], Microsoft and NVIDIA's jointly launched Turing-NLG [14], Baidu's Ernie 3.0 Titan [15] and recently launched ERNIE. Moreover, there are also DeepMind's Chinchilla [16] and Gopher [17], EleutherAI's GPT-NeoX [18], Anthropic's Claude [19], BigScience's BLOOM [20], and so on. Most of them are proprietary, and all of them have been based on the Transformer architecture or a variant of it.

With various LLMs and their chatbots entering the market, there is likely to be fierce competition among different LLMs in the near future. While there are academic methods for evaluating the superiority of models [21], ultimately, the market will decide which ones will succeed. In my opinion, in order to win the LLM war, a successful LLM and its products must have the following characteristics:

- **Having access to more and better quality data than other LLMs.** Just as discussed in Section II-B, if other LLMs are limited to training and finding answers only from the public domain data, and one LLM also use a more extensive knowledge from much more sources to provide better answers to a variety of questions beyond the public domain, this LLM will be definitely superior than others.
- **Having more powerful computing resources.** Massive amounts of training data required large deep learning models. As early as 2018, OpenAI found that the amount of computational power used to train the largest AI models had doubled every three to four months since 2012. Since then, the idea of LLM has been introduced and now it has become the mainstream of NLP. Currently, a couple of LLM have hundreds of billion parameters. GPT-3, which has 175 billion parameters, took 3.64×10^3 GPU PF-days to train, with an estimated cost of 4.6 million dollars for a single training cycle. If Huang's Law for GPUs[10] really replaces Moore's Law for CPUs, then we can expect that future models will become even more complex. Whoever can own more computing resources and have access to more data (including human feedback) will be able to produce better models. Given the high cost of computing resources, it is clear that only big technology companies or those with significant funding have advantage in LLM development.
- **Faster model iteration and refresh than others.** The biggest challenge for LLMs is how to update models. Transfer learning may sometimes works, but it is better to refresh the entire LLM. For example, the latest data GPT used for training was from 2021. It will take long time for openAI to collect more recent data and upgrade the model. ChatGPT is almost ignorant of most recent things and may answer incorrectly, which contradicts the public's need to know the latests news and knowledge. The academic community has a disadvantage in updating models because they are slower than the industry, while big technology companies often update models more slowly than startups due to bureaucracy issues and a greater sense of responsibility, while startups usually lack computing resources and sufficient financial support. Perhaps a good approach would be startups and big companies to collaborate, and combine LLMs with search engines that have the latest information, just as Microsoft's New Bing is currently doing.

As LLMs and AI chatbot technology become increasingly mature, perhaps people will increasingly rely on high-performing chatbots and gradually break free from their dependence on current search engines. People will use chatbots more frequently than they use Google search or any other search engines today. Just as Microsoft has integrated ChatGPT into its new Bing and 365 Copilot, Morgan Stanley has also started using an OpenAI-powered chatbot, in the foreseeable future, chatbots could generate enormous profits. Meanwhile, just as today's search engine market is dominated by Google search with over 90% of the market share, one or several best LLMs with their chatbots could potentially dominate the chatbot or

[10]Nvidia CEO Jensen Huang mentioned that in 2018 that the performance of GPUs is doubling every 18 to 24 months, which is faster than the rate predicted by Moore's Law for traditional CPUs.

even multimodal market in the future. This is called **"LLM monopoly"** that the chatbot market will be dominated by only a few tools. In addition to chatbot, other AI domains may also see the emergence of monopolistic tools in the near future. Certain multimodal AI tools may establish a monopoly that spans multiple areas.

Let us call the future most popular chatbot as "Chabot" for now. Chatbot will inevitably have a huge and growing training dataset. What would happen if Chatbot does not share revenue or profits with its data providers? Even if the data is currently in the public domain, it is highly likely that the original data owner will lose more and more visitors and customers, or lose the reputation of originality due to the impact of Chatbot. This will be a zero-sum game: the original data owner loses traffic and customers, while Chatbot uses the data to answer questions and earns revenue without given credit or quote anyone. Of course this is not acceptable. Even data that were once publicly available may start being copyrighted to protect data owners' interests, as data owners prevent Chatbot from "stealing" their data and protect their own interests. As a result, Chatbot will only be allowed to use increasingly limited data due to copyright laws, resulting in lower quality or even constantly incorrect answers for customers. If there are still a large number of people relying on Chatbot at that time, the misleading responses from Chatbot may cause serious consequences. And if Chatbot loses popularity due to a decline in quality, the AI revolution sparked by today's chatbots will suffer a serious setback, and we will return to the era of Web 2.0. This is the consequence of the zero-sum game between AI tools and data providers.

On the other hand, what if Chatbot shares revenue with its data providers, no matter they are public domain or not? Even if the data provider loses customers and traditional reputation due to Chatbot, they can still get revenue from the world's best chatbot. This is a virtuous circle: Chatbot collects more diverse, extensive and high-quality data to improve its performance, attract more customers, and gain greater revenue; while data providers can also get more revenue, and even get new reputation from sharing revenue with Chatbot. Once the shared revenue from Chatbot exceeds the loss caused by customer loss, it becomes economically good for the data owner. We can be sure that even if Chatbot is willing to share revenue with data providers, there will still be "old school" data owners who decline to share data. However, many other data owners will be encouraged to offer their data and participate in the Chatbot sharing revenue project. It is truly a win-win plan for all parties, and a necessity for building a good AI ecosystem in the future.

"Chatbot" is only an example for all popular AI tools in the future. In summary, we may have two possible paths in the upcoming AI era. The first path is that AI tools have to use "free data" only for training and profits from it. But even free data could be no longer free, if data owners consider AI tools as "data thieves" and never give the original data owners any credit, so the owners refuse to open their data any more. This creates a paradox for popular AI tools: on the one hand, they want to become the leading "move and shaker" and even monopolistic tools in the AI era, but on the other hand, the data it can obtain is constrained even dwindling. If this is true, the AI era will be a hostile and toxic era. The second path shows a much better prospect which is what we hope to see: popular AI tools are willing to share revenue with their data providers, and more and more data owners are encouraged to participate in AI tools' revenue sharing programs as new data providers. As these AI tools obtain more diverse and high-quality data everywhere, they constantly improve their performance, gains more users, earns more revenue, and shares more money with its data providers. This will be a utilitarian AI era where all parties benefit.

Once we know that sharing revenue is a must for the AI era, we can then discuss the technical issues: how to make various AI tools share revenue? Let us first take a look at how enterprises in the Web 2.0 era share revenue.

III. Today's Revenue Sharing models

Traditionally, revenue sharing refers to the distribution of a company's revenue among stakeholders, shareholders, and other contributors. In particular, some companies have adopted revenue sharing models to collaborate with public contributors such as photographers, artists, writers, and content creators for their role in the companies' success. Some companies create two-side platform to connect producers with consumers directly, while others create their own content and products with the assistance of public contributors, such as Getty images, Shutterstock, Adobe Stock, Google Adsense, Youtube, Amazon Associates and so on. Among them, Getty images and Goole Adsense are two typical models of sharing revenue with public contributors, and the current and future AI tools can also be inspired by their business models.

A. Getty Images

Getty Images is a well-known American-British visual media company that specialized in stock images, editorial photography, videos and music for business and consumers. It has a large contributor base consisting of over 488,000 photographers and videographers as well as more than 300 content partners to deliver content to Getty Images for distribution [11].

Getty images offers a revenue sharing model to its contributors. It sells stock photos and videos on-demand to consumers, then pays the original photo and video contributors royalty fees. For example, Getty images requests consumers a cost of \$375, \$1625 and \$3000 for a single, five and ten medium-size images respectively[12], then pays 20% as royalty rate to the image's provider. On the other hand, the royalty fee varies for iStock contributors. Exclusive contributors receive a default royalty of 25% for stock photos, which increases to 30%, 35%, and 45%, based on the total number of downloads.

Getty Images' revenue-sharing model encourages people to provide high-quality images and videos, thus ensuring that Getty Images has a vast stock images and videos. Based on the

[11] The numbers are from Getty images 2022 news report.

[12] More details see Getty Image Pricing.

download volume of images and videos, the revenue-sharing rate has been divided into several fixed tiers. Unfortunately, this business model does not apply to AI tools. This because while Getty images contributors images and videos can be explicitly measured by download volume to determine their popularity, the data providers for an AI tool cannot use any similar metrics to measure whose data is ”more popular” with AI tool’s customers. Although we can still assign some fixed revenue-sharing rates to different data providers, those rate will be arbitrary but not data-driven. Therefore, Getty images’ revenue-sharing model cannot be applied to AI tools in the AI era.

B. Google Adsense

In the early days of the Internet, a majority of website owners had very limited options to earn revenue from their sites. Google introduced its AdSense program in 2003 and changed the game. Adsense is an online advertising platform that allows website publishers to display ads that were tailored to their content and audience, and Google took care of matching ads with website publishers, who can earn money when audience clicked on the ads.

Figure 1 shows Google’s business model in general. Google’s advertising models, which includes Google Ads (launched in 2000) and Google AdSense (launched in 2003), are among the most successful online business models in the era of Web 2.0, and the most important source of revenue for Google. Keep in mind that Google AdSense is different from Google Ads, although both of them are advertising platforms offered by Google. AdSense is for website publishers who want to display ads from Google and earn a portion of the revenue that Google earns from the advertisers who run the ads. On the other hand, Google Ads only run ads on Google centralized properties such as Google search results pages and YouTube. While content contributors to Google properties such as YouTubers can also benefit from Google Ads, Google AdSense has a much broader range of customers – potentially any website publisher who joins the program can benefit from it. Which is better, Google AdSense or Ads? From the utilitarian perspective for Web 2.0, perhaps AdSense is better than Ads. The latter is build on Google’s centralized system, highly depending on Google, and the participants, whether they provide money or content, must contribute to Google (such as advertisers and YouTubers). The former is more like an “alliance” established between a broad range of websites and Google. Websites do not need to directly contribute to Google, but can still share revenue from Google, making it a win-win system for both websites and Google. There are other programs similar to Google AdSense, such as Amazon Associates.

How much revenue does Google AdSense share with website publishers? Different from Getty Images, the revenue-sharing rate is not simply several fixed tiers, but depends on many factors. Advertisers pay Google on the basic of ads clicks, and website publishers receive 68% of the ads revenue recognized by Google in connection with AdSense. For example, per click earnings depends on ads number of impressions, click through rate (CTR) and cost per click as

$$\text{Earnings} = \#\text{ of impression} \times \text{CTR} \times \text{CPC}$$

where CTR and CPC vary depending on a variety of factors, including ads’ categories and countries. Typically AdSense pays \$0.2 - \$2.5 per 1,000 views on average. Although the entire revenue sharing system is far more complex than Getty Images, participants in Google AdSense can clearly see their earnings on their platform, and the overall process is still transparent. Other programs may have different way to pay, but the principle of revenue sharing is more or less the same. For example, Amazon Associates allow participants to earn a commission when someone clicks on their unique affiliate link and makes a qualifying purchase on Amazon. The commission rate varies depending on the product category and ranges from 1% to 10% of the sale price.

In the upcoming AI era, there may be new forms of advertising, or we can assume that a large source of revenue for various AI tools will come from membership fees for premium customers. Let us consider whether the principle of AdSense can be borrowed by AI tools. While there is no click and click-through-rate in the entire process of AI tool customers inputting prompts and tool outputting responses, we can still analogize every single prompt input as a click. For AI tools, **Cost Per Action (CPA)** in the Web 2.0 era can be replaced by **Cost Per Prompt (CPP)**. Although the customers of AI tools do not pay based on the number of prompt inputs, we can still establish a scoring system for AI data providers by calculating the connection between each prompt input and all data providers, and then share revenue among data providers based on the scoring system. In the next chapter, we will examine how to establish such a CPP scoring system for data providers in detail.

IV. A New Revenue Sharing Model for AI data providers

Figure 2 demonstrates the new revenue-sharing business model including data providers for an AI tool. Similar to current Google’s business model, all parties (users, data provides and AI tool) can benefit from this model. However, the difference between this model and Google’s model is obvious: most metrics used for Web 2.0 business model need to be replaced by new metrics. Meanwhile, there is a separated revenue-sharing model, which is crucial to calculate how to distribute revenue to the AI tool’s all data providers.

A. Old Metrics and New Metrics

In the Web 2.0 era, people used to ask the question: “how to measure online ads’ importance and success?” In the AI era, the corresponding question is not how to measure the importance of an AI tool, but how to measure the importance of each data provider for an AI tool. We can see that the metrics used to measure ads importance are essentially ineffective for AI tools.

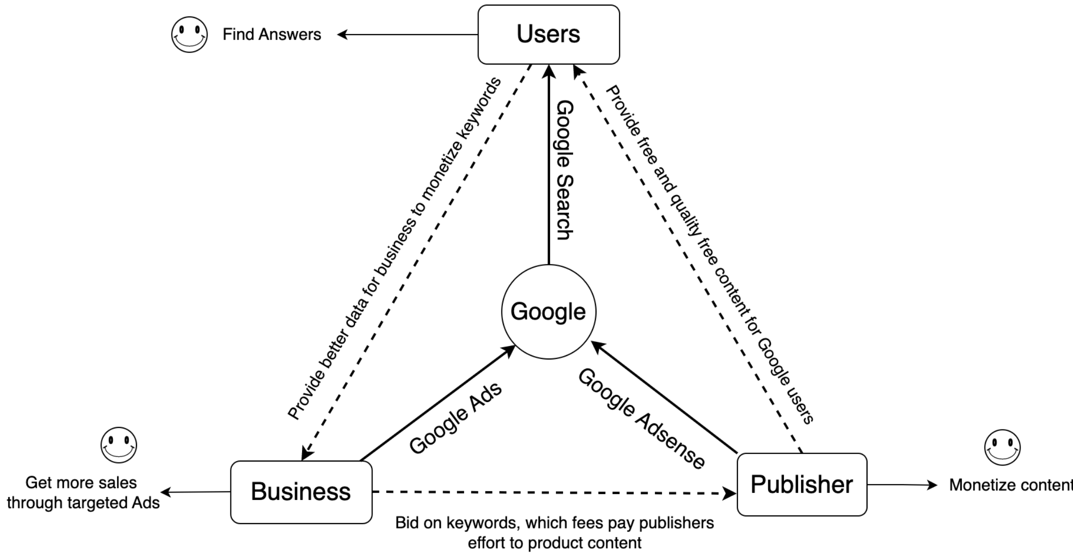

Fig. 1: Google's Business Model, adapted from Google Business Model Analysis. Business join Google via Google Ads, while website publishers join Google by AdSense. Both ordinary users, business and publishers can benefit from this business model. However, this model cannot be used in the AI era for AI tools.

(1) **Page views**. AI tools may still be integrated into various websites, but the count of web page views no longer matches the usage of AI tools. For example, the web-based ChatGPT can be used for a long time without refreshing the page. The relevant "bound rate" and "average view duration" will be less important for AI tools.

(2) **Clicks**. The click that originated from hypertext markup language (HTML) may no longer be applicable in the AI era, as AI tools no longer reply on clicks to function. Consequently, many important metrics associated with clicks, such as Click Through Rate and Cost Per Click, are no longer applicable for calculating the importance of AI products as well as data provided for AI training.

(3) **Conversion rate**. The conversion rate of an advertisement is closely related to clicks and purchases. However, for an AI tool, although it is still possible to calculate the proportion of free users and premium users, such calculations do not help in understanding how the AI training data is used and being utilized.

(4) **Engagement rate**. Similar to the conversion rate, the old method of calculating the user engagement rate is no longer applicable for evaluating the importance of data in AI tools.

So what are the most important metrics for an AI tool? These metrics as follows:

(I) **Prompts**. It is highly likely that prompts will replace clicks as the most important metric in the AI era. Each input prompt from a user contains certain information, triggering AI tools to response. The traffic and information of prompts will become the most important measurement of an AI tool in the visible future.

(II) **Cost Per Prompt**. As mentioned at the end of the previous section (III-B), Cost Per Click will be replaced by Cost Per Prompt. How does an AI tool call its pre-trained or fine-tuned model to generate a response for each prompt, and how are prompts and generated responses related to the AI tool's training dataset? These are crucial questions.

(III) **Data Engagement Rate**. Let us qualitatively image the concept of "data engagement". Consider an AI chatbot that has been trained on a large amount of text data about physics and astronomy. This chatbot is likely to be able to answer many questions on physics and astronomy, but it may not know anything about Shakespeare's plays. Therefore, if a prompt is related to physics or astronomy, we can say that the training data of this chatbot engaged, or has a high "degree of engagement". On the other hand, if a prompt is to ask about Shakespeare, we say that the chatbot's data is not engaged, or has a very low degree of engagement for this prompt. If an AI tool has may classes of training data across a wide range of topics and areas, the engagement degree of each class is obviously different for an input prompt. By quantifying and normalizing the engagement degree, we can calculate he data engagement rate.

B. *Scoring System and Requirements*

In order to calculate the new metrics mentioned above such as Data Engagement Rate for AI tools, we must build a scoring system, where each data provider for an AI tool can have a score. Using this scoring system, we can calculate the Cost Per Prompt for each data provider and the engagement rate of each data provider for prompts. Then, we can use the engagement scoring system to calculate the AI tool's share of revenue for each data provider.

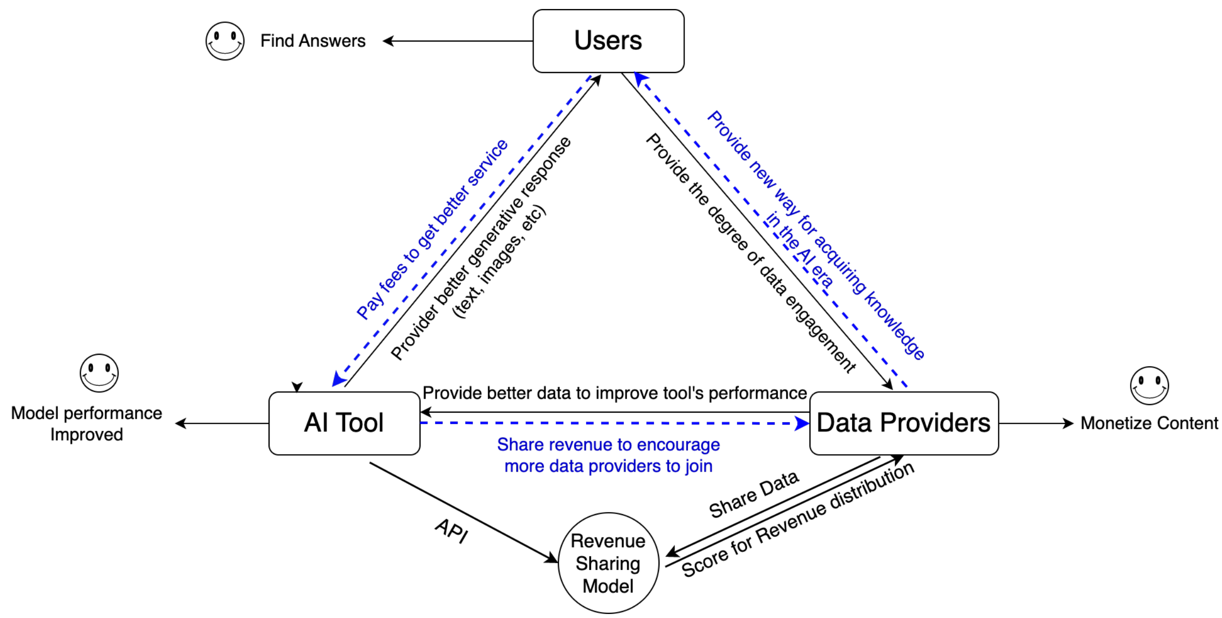

Fig. 2: Proposed business model for AI tools in the AI era. It includes four key elements: the AI tool/model itself, the users, the providers of data for the AI tool, and the model for revenue sharing. If there is a continuous inflow of data providers into the AI tool's ecosystem, similar to Google's business model (compare to Figure 1), users can receive better generative responses due to AI model improvement, and data providers can earn revenue. However, there are significant difference between this business model and Google's model. For instance, users may need to pay fees to access better services instead of using them for free. Also, data providers cannot directly show their content to users, the content is transformed by the AI model and displayed to users after AI model's training. Additionally, the revenue-sharing model is independent of the AI tool itself. See Section IV-B for more discussion.

To build a scoring system, there are several crucial prerequisites:

- **The scoring system is based on prompts.** As mentioned in Section IV-A, for AI tools, the most important metric is prompts rather than clicks. Therefore, the scoring system is based on each prompt, not each click. Specifically, when a user inputs a prompt for an AI tool, each data provider of the tool will receive a score for that prompt. These scores for all prompts assigned to data providers are linearly combined to produce a final score for providers, which is used to measure the engagement of the data they provid.
- **The scoring system should be as simple as possible** – DONOT use neural network to build it. Nowadays, there are more and more AI tools, a considerable number of which employ deep learning models. Various LLMs and large computer vision models have reached the scale of tens and hundreds of billions of parameters, with complex deep neural network architectures. Such large deep learning models are often business secrets and not open to the public, and difficult for the public to understand. The scoring system is for general general data providers, we must ensure that the scoring system can be understand by the public, which means the scoring system as a model itself, should be as simple as possible. For example, the scoring system can be based on a tree-based model or vector computations rather than relying on the AI model architectures.
- **The scoring system is build for all data providers of an AI tool, we need to ensure that the AI tool's entire training dataset is open and transparent, allowing third party verification.** Many large deep learning models have publicly released their training datasets, but there are also many models which datasets are still publicly available. There are various reason for this, such as some datasets include proprietary content, meaning that the companies that developed the models have exclusive rights to use them. Some datasets may contain personal information or other sensitive data that could be used to identify individuals or compromise their privacy. Moreover, legal restrictions or concerns about data bias may also cause companies and organizations to hesitate in opening the entire training datasets of their models. To establish a scoring system that applies to all data providers while avoiding the issues mentioned above, the following solution can be employed: the dataset used by an AI tool is not made public, but only shared with a reliable third party that is responsible for building the scoring system. Brining in a third party ensures a certain level of transparency for the AI model's dataset, while also avoiding the dataset being made public to its competitors.

- **The scoring system must be as independent from the AI tool as possible.** Whether the scoring system is built by a third party or by the company or organization that created the tool, we do not want the scoring system to be too closely related to the tool's model itself. This is because the more they are related, the more the scoring system depends on the model itself, the harder to explain the scoring system to the public. We need to ensure that if we treat the AI model as a black box, the scoring system can still be established and explained. However, when I say "as independent as possible", it means that the scoring system and the AI tool can also have some connection, but the connection can only be at the API level. That is to say, the AI tool itself provides API-level information to the scoring system, such as doing text embedding through API (refer to the later Section IV-D), then the scoring system builds independently based on this.

Once the scoring system is established, we can use it to evaluate the engagement of each data provider and distribute revenue accordingly. Let us now take a look at the possible mechanisms for building the scoring system.

C. Classification

The most straightforward approach to create a scoring system is to build a supervised classification model, which can generate probability scores. For the training dataset of an AI tool, if we consider each data provider as an independent class, or a combination of data providers as a class, we can build a pre-trained classification model on this training dataset based on the class (provider) to which each document belongs. Once a user inputs a prompt and receives a generative response from the tool, we can use the pre-trained classifier mentioned above to assign each class a probability score based on the prompt or the response, so the probability scores reflect the degree of engagement of each class for this prompt. The summation of probability scores for all prompts can be used to establish a scoring system.

*1) **Newsgroup20 Demonstration**:* To better illustrate how it works, we can use the benchmark dataset Newsgroup20 to demonstrate how to build a classifier on a series of text documents and calculate the probabilities that any text document belongs to each class. **Newsgroup20**, which was originally collected in 1995-1996 from the "Newsgroups" project, is a widely used dataset for text classification tasks, consisting of around 20,000 documents distributed among 20 distinct newsgroups. The dataset has been split into training and testing sets, where training set contains $\sim$11.3K documents and testing set contains $\sim$7.5 K documents. We can view each newsgroup as a data provider – imagining that Newsgroup20 is composed of data provided by 20 different data providers. If we treat each provider as a class, we can use the training set to built a text classifier for these 20 classes, and calculate the probabilities of any text document belonging to each class.

The first step is to do text embedding and vectorize documents. Text embedding is more sensitive for calculating text similarity, which will be discussed in Section IV-D. As a baseline model, we utilize the widely-used TF-IDF technique, which includes including `CountVectorize` and `TfidfTransformer` to tokenize and vectorize each text document, and use `LinearSVC` and `CalibratedClassifierCV` to train and build the classifier. Next, we can apply the text classifier to any text inputs. Let us randomly pick up a document in the testing set. For example, we have a document which talks about Hubble Space Telescope (HST) in testing set (hereafter Document HST) :

```
From: henry@zoo.toronto.edu (Henry Spencer)
Subject: Re: HST Servicing Mission Scheduled for 11 Days
Organization: U of Toronto Zoology
Lines: 12

In article 1993Apr27.094238.7682@samba.oit.unc.edu
Bruce.Scott@launchpad.unc.edu (Bruce Scott) writes:

If re-boosting the HST by carrying it with a shuttle would
not damage it, then why couldn't HST be brought back to
earth and the repair job done here?

The forces and accelerations involved in doing a little bit
of orbital maneuvering with HST aboard are much smaller
than those involved in reentry, landing, and re-launch.
The OMS engines aren't very powerful; they don't have to be.

SVR4 resembles a high-speed collision | Henry Spencer
@ U of Toronto Zoology between SVR3 and SunOS.
- Dick Dunn | henry@zoo.toronto.edu utzoo!henry
```

Newsgroup20	Score 1	Score 2	Score 3
sci.space	0.9527	322.2	0.6710
rec.autos	9.05e-3	2.859	1.77e-3
sci.electronics	5.16e-3	10.10	1.67e-3
comp.sys.mac.hardware	4.77e-3	4.48	0.200
misc.forsale	4.06e-3	3.91	6.07e-3
soc.religion.christian	3.29e-3	2.78	4.88e-3
rec.sport.hockey	2.92e-3	1.22	8.50e-3
rec.sport.baseball	2.87e-3	1.26	3.30e-2
comp.graphics	2.43e-3	10.2	2.13e-3
alt.atheism	2.17e-3	2.51	6.56e-4
comp.sys.ibm.pc.hardware	2.10e-3	4.08	1.25e-2
comp.windows.x	1.56e-3	4.10	1.84e-2
comp.os.ms-windows.misc	1.54e-3	2.34	6.11e-3
talk.religion.misc	1.50e-3	2.18	1.46e-2
talk.politics.misc	1.30e-3	4.00	3.41e-3
sci.med	1.10e-3	7.81	2.95e-3
talk.politics.guns	5.37e-4	3.01	1.48e-3
talk.politics.mideast	4.45e-4	1.04	4.78e-3
rec.motorcycles	2.72e-4	1.90	5.33e-3
sci.crypt	1.90e-4	1.99	3.34e-4

TABLE I: Distribution of Newsgroup20 probabilities given by the text classifier. Score 1 are the probability scores of one document from Newsgroup testing set, Score 2 are the accumulated probability scores of all document labeled as "sci.space" from Newsgroup testing set, while Score 3 are the score of a given prompt input "Why the sky is blue?".

We use the text classifier to show the probabilities of the Document HST belongs to each class. Score 1 in Table I gives the result: this document has 0.9527 probability of belonging to the class "sci.space", which is consistent with its real labeled class. On the other hand, the document has lower probabilities of belonging to other classes, such as 0.009 for “rec.auto”, and 0.005 for “sci.electronics”. We can consider the twenty probability scores as scores for a single document belonging to each of the twenty classes. Just for the Document HST, “sci.space” has the highest engagement due to its highest probability score.

The probability scores can be summed up for many documents. For example, if we collect all the documents labeled as “sci.space” in the Newsgroup20 testing dataset, which totally has 394 such documents, the summed up score for “sci.space” is 322.23, and for “comp.graphics” is 10.19. Score 2 in Table I provides the score corresponding to each class. Note that the sum of all these scores is 394, which is the number of documents. If we consider that each document weights equally, we can normalize Score 2 by dividing it by 394 to obtain the normalized probability scores. Again, the class “sci.space” has the highest degree of data engagement, followed by “sci.electronics” and “comp.graphics”. It is easy to understand why the class “sci.space” has the highest degree of engagement, since all documents used in this case are from this class. However, for other classes, the degree of engagement is not so intuitive and evident. For instance, why does the degree of engagement of “sci.electronics” exceed that of “rec.autos”? To answer such questions, Explainable AI (XAI) is a commonly used method to explain machine learning decisions such as text classification and probability score calculation. In particular, SHAP can be used for NLP model explainability [22], [23]. However, XAI is beyond the scope of this paper, we expect to discuss it further in future work.

Next, let us take a look at the probability scores assigned by the aforementioned text classifier to any given text input. Score 3 in Table I gives the score distribution for the given input prompt: `Why the sky is blue?` We can see that the highest probability is still from “sci.space”, which is understandable, followed by the probability of “comp.sys.mac.hardware”, which is less obviously. As mentioned before, we may need XAI to better understand the probability scores, but at least the text classifier indeed give a prompt-based scoring system to measure the degree of engagement of each class for any documents.

*2) **Extend to General Case**:* In principle, we can apply this idea to any AI tools as well. Figure 3 shows the basic pipeline for establishing a scoring system. Each data provider, or a group of data providers can be treated as a class. With each document assigned a class, we can use AI tool’s all training datasets to train a text classifier, which can be used to establish a score system for any text documents (including paragraphs and sentences). From the perspective of revenue sharing, if each class represents a data provider, who are able to share revenue from the AI tool with other providers, then the revenue that one provider can share can be calculated as:

$$\mathscr{R}_i = \mathscr{R}_{\text{tot}} \times P_i, \tag{1}$$

where $\mathscr{R}_{\text{tot}}$ is the total shared reveue, $\mathscr{R}_i$ is the revenue shared by $i-$th provider, and P_i is the provider’s normalized score

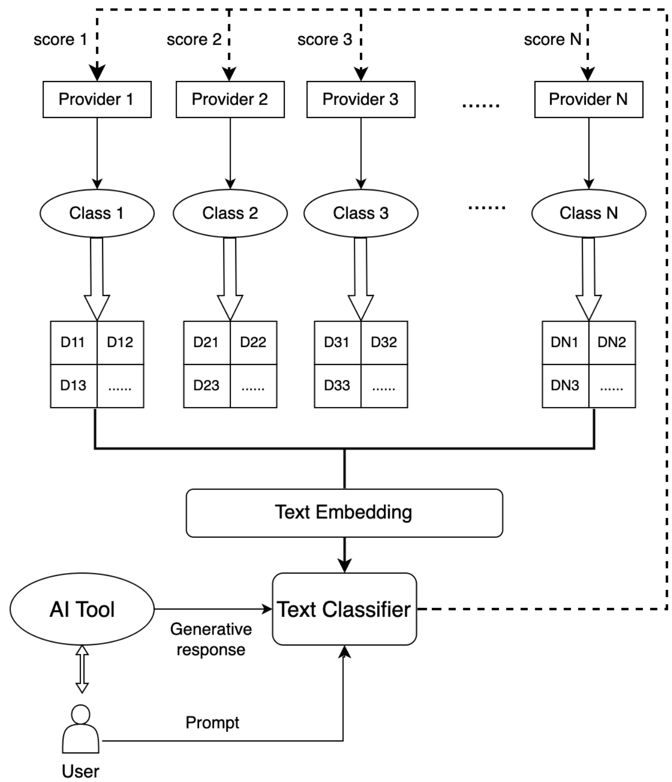

Fig. 3: The revenue-sharing scoring system based on text classification. Each data provider can be considered as one class (or a cluster of data providers as a class), with a series of documents in each class (D$_{i1}$, D$_{i2}$, D$_{i3}$ for $i-$class). Documents are vectorized using some text embedding techniques (see Section IV-D and Figure 4 for more discussion on text embeddings), then A text classifier can be built for all documents. A user inputs a prompt in the AI tool and receives generative response, both the prompt and response can go through the text classifier to generate a series of probability scores for each data providers.

calculated by

$$P_i = \sum_n p_{in} / \sum_{i,n} p_{in} \tag{2}$$

and p_{in} is the probability of the $n-$prompt input or its response belonging to the $i-$th provider’s dataset.

*3) **GPT as Another Case** :* The full training datasets for ChatGPT and GPT-4 have not opened to the public. Let us look into GPT-3. We can consider the training dataset of GPT-3 comes from the following five providers: Common Crawl, WebText2, Books1, Books2, and Wikipedia, and treat each provider as one class to build a text classifier on the training dataset. A somewhat tricky question is how to decide on the size of each text document for text classification training. This is also a general question not only for GPT. For example, if one data provider offers three documents, each of which has millions of tokens in length, while another data provider offers hundreds of documents, each with only one sentence and a piece of information, it is clear that the sizes of the documents are very different. One approach is to consider using truncated documents to roughly equalize their length. For GPT models themselves, each input document is typically a sequence of contiguous text tokens. The original model used a fixed-length input sequence of 1024 tokens, while GPT-3 uses a more flexible input format. For text classification, we can truncate text to make a single document to have a size that is roughly equal to the average length of generative responses.

For any prompt input by a customer, the pre-trained text classifier can assign scores to each of the aforementioned five classes. The accumulation of these scores can indicate the degree of engagement of the five data providers in all chat responses, thus determining how much revenue each provider can share – if ChatGPT would like to share revenue with them.

*4) **Large Number of Data Providers and Classes**:* Of course, dividing GPT-3 into five data providers is just for brainstorming. In the future, if we have hundreds or thousands or even more data owners providing training data for GPT, we can still label these large number of data providers as different classes to build a text classifier, and calculate each provider's degree of data engagement by classifier probability scores.

Reuters-21578	Score 1	Score 2
earn	780	–
acq	7.16	0.538
money-fx	3.84	1.52e-2
ship	1.74	–
grain-corn	1.66	–
crude	–	2.37e-2
interest	–	1.21e-2
grain	–	7.10e-3

TABLE II: Top five probability scores for all documents in the Reuters-21578 testing set labeled as "earn" (Score 1), and for the prompt int "Why the sky is blue?" (Score 2).

We can use another benchmark dataset, Reuters-21578, to demonstrate for many-class problem. The Reuters-21578 dataset consists more than 21K newswire articles from the Reuters news agency, which were collected in 1987 and used for text classification research. The articles are classified into 90 different topics, but we can further divide the classes to create more specific ones. For example, if a document belongs to both the "grain" and "wheat" classes, we can assign it to a new class called "grain-wheat". Similarly, if a document belongs to the "interest", "retail" and "ipi" classes, we can create a new class called "interest-retail-ipi". As a result, the Reuters-21578 training set is divided into a total of 464 classes. We can build a text classifier on the training set for these classes using the same method mentioned above for the Newsgroup20 example.

Next we test the text classifier probability scores by taking all 1041 documents in the Reuters-21578 testing set that are labeled as "earn", seeing how the classifier assigns probability scores to these documents. Score 1 in Table II gives the result for five classes with highest scores. The summed up probability score of "earn" for these documents is 780, which is the highest score, followed by significantly lower scores from other classes. When we test another single prompt input `Why the sky is blue?` , the text classifier assigns the highest probability score to the class "acq", followed by "money-fx", "crude", "interest" and so on. It may be difficult to understand why Reuters-21578 would classify a question about the sky as being highly related to corporate acquisitions (acq). One main reason is that the training dataset of Reuters-21578 is incomplete, which means the training data does not cover enough topics, for example, topic related to "space". When there is a lack of training data for a specific topic, the classifier can only assign the input text to the most similar class. If this is the case for a real AI tool, We do not expect the AI tool to have a suitable response for this prompt.

Therefore, building a scoring system based on probability scores of a text classifier on the training dataset of an AI tool can be comprehensible, but its interpretation could be challenging if the training data is incomplete or poorly labeled. In such scenarios, an alternative scoring system based on unsupervised techniques like text similarity can be developed for any text.

D. Text Similarity

Tex similarity is an unsupervised technique to measure how similar two or more pieces of text are to each other in terms of their meaning or content. For example, let us consider the similarity among the following three sentences:

- `Why the sky is blue?` (Sentence I)
- `Why the space is dark?` (Sentence II)
- `The sky is blue due to a phenomenon called Raleigh scattering.` (Sentence III, answer to Sentence I).

The commonly used method to compare text similarity is to use some text embedding techniques to vectorize all texts, and measure the distance between vectors in a high-dimensional space. The measurement techniques including **cosine similarity**, **word embedding-based similarity**, and **Latent Semantic Analysis**. Using TF-IDT method to vectorize and cosine similarity to measure the distance for the above three sentences, we have the similarity matrix:

$$\begin{bmatrix} 1 & 0.452 & 0.424 \\ 0.452 & 1 & 0.138 \\ 0.424 & 0.138 & 1 \end{bmatrix}$$

which means the first sentence is more similar to the second sentence than the third sentence, while the second and third sentences have the lowest similarity.

*1) **Universal Text Embedding Method and Cosine Similarity**:* Next, let us use Newsgroup20 again to demonstrate how to establish similarity between any document and the various classes in a dataset. The first thing is always to find an embedding method. Images and videos may need some other techniques, let us focus on text embeddings in this section.

We still use Newsgroup20 to demonstrate how to embed and vectorize documents in a dataset. In principle, the TF-IDF method provides a technique to perform tex embedding

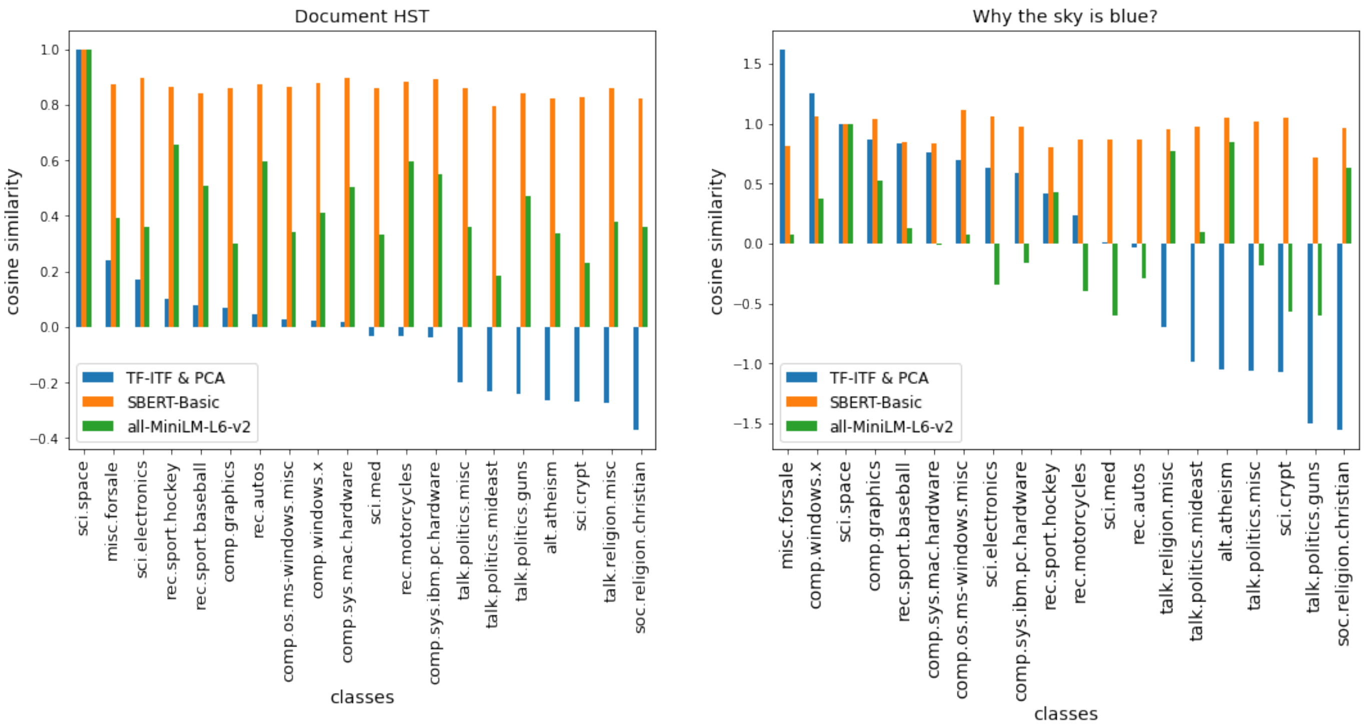

Fig. 4: Cosine Similarity of between Newsgroup 20 classes and Document HST (left), and one sentence `Why the sky is blue?`. Multiple text embedding methods were used to vectorize text documents: directly TF-ITF and PCA to reduce vector dimensions, pre-trained models including SBERT-Basic and all-MiniLM-L6-v2.

for documents in the Newsgroup20 training dataset and convert documents to vectors. As a result, each training document in Newsgroup20 is vectorized into a 130093-dimensional vector. And we can vectorize any document into the same dimensional vector, and calculate its cosine similarity between vectors to show the degree of similarity between any documents. However, since the vectors generated by TF-IDF are of high dimensions and are also very sparse, such calculations are usually time and energy consuming. To simplify the calculations, a commonly used method is to reduce the dimensions. As an example, I used Principle Component Analysis (PCA) to reduce each vector in the training set from 130093 dimensions to 768[13], and used the same method to reduce any vectorized document to 768 dimensions.

Let us look at the Document HST mentioned in Section IV-C about the Hubble Telescope. After text embedding, we calculated the cosine similarity between each document in the training set and the Document HST, and the averaged cosine similarity between the Document HST and the $i-$th class can be calculated by

$$\langle \mathbf{V}_{\mathrm{HST}} \rangle = \sum_{j=1}^{K} \mathbf{V}_{\mathrm{HST}} \cdot \mathbf{V}_{ij} \tag{3}$$

where $\mathbf{V}_{\mathrm{HST}}$ is the embedded Document HST as a vector, V_{ij} is embedded documents in $i-$th class with $j = 1, 2, 3...K$ and K is the number of documents in that class. This gives the similarity between each class in the training set and Document HST. The left panel of Figure 4 shows the cosine similarity values normalized by "sci.spacce" similarity value. We see that the similarity between "sci.space" and Document HST is the highest, while the similarities between "comp.windows.x", "comp.sys.mac.hardware", and Document HST are almost zero. The similarities between 7 classes including "talk.politics.mideast". "soc.religion.christian" and Document HST are all negative, which indicates that these classes have completely different topics, contexts, or meanings from Document HST.

It should be noted that the TF-IDF text embedding technique and PCA relies solely on the documents contained within a dataset, independent of any other models. This approach provides a universal standardized way to measure the similarity between each class in the training dataset and any document, so it can be called as a form of "universal text embedding" methodology.

*2) **Model-Based Particular Embedding Method**:* Universal text embedding methods are often computationally expensive. For example, the TF-IDF method is clearly an expensive method that requires processing the entire training dataset. If the training set is very large, TF-IDF will produce a huge number of dimensions, making dimensionality reduction such as PCA a time-consuming and laborious task.

A more convenient method for calculating text similarity

[13] The number 768 is the dimensions of text embedding from base-Bert. I use the same dimension for a later comparison with BERT.

is to use pre-trained models to directly implement text embedding. For example, BERT provides several methods for vectorizing any text document. For demonstration, we used Sentence-BERT (SBERT) to embed and vectorize documents in Newsgroup20 training dataset and Document HST. The left panel of Figure 4 shows the results provided by SBERT-Basic, which are obviously different from those provided by TF-ITF and PCA. Although "sci.space" still has the highest similarity with Document HST, other classes also have similar cosine similarities with Document HST. There is no negative similarity between any class and Document HST. This is because BERT-Basic was pre-trained on the SNLI dataset, which is a quite different dataset from Newsgroup20. As a result, the text embeddings generated by SBERT-Basic and those obtained by directly applying TF-IDF and PCA on the Newsgroup20 training dataset are quite distinct from each other. In addition, we also used text embedding provided by another pre-trained model called all-MiniLM-L6-v2, which used different training dataset, to calculate the similarity between each class and Document HST, and although "sci.space" still shows the highest similarity, it can be seen that it provides different result from TF-ITF and SBERT-Basic.

The right panel of Figure 4 shows the results of another demo, which is the similarity between the prompt `Why the sky is blue?` and each class of Newsgroup20. It can be seen that using the TF-IDF and PCA embeddings directly built from the Newsgroup20 training dataset and the embeddings from other two pre-trained models provide significantly different similarity distributions. In Section IV-C , `Why the sky is blue?` has the highest probability for "sci.space", but in text similarity analysis, the class most similar to this sentence is other class such as "misc.forsale", which means the results of classification and text similarity may be contrast with each other.

Clearly, different text embedding methods will provide different results for text similarity. Which method should we use to calculate text similarity and creating a scoring system?

*3) **Scoring System based on Text Similarity**:* If we aim to set up an alternative prompt-based scoring system using text similarity, there are two feasible approaches. Figure 5 shows the two possible approaches.

The first approach is to use a universal text embedding technique to vectorize all documents. For example, TF-IDF and dimensionality reduction can be considered as one of such techniques. Moreover, while some text embedding techniques rely on models, we can still use them for text embedding if publicly available pre-trained models gain widespread acceptance among the public.

The second approach is closely related to the special AI tool we want to build the scoring system for. The AI tool may provide its own text embedding technique, which is created by its own training dataset and model architecture. For example, GPT also provides several pre-trained embedding models, including Ada (1024 dimensions), Babbage (2048 dimensions), Curie (4096 dimensions), Davinci (12288 dimensions), we can use GPT API perform text embeddings and convert documents to vectors, calculate the similarity between any documents and GPT, then build a scoring system via text similarity specifically for GPT.

For any input prompt, the score of text similarity between the prompt and the $i-$th class can be calculated by (similar to equation [3], see also Figure 5):

$$s_{\mathrm{p}i} = \sum_{j=1}^{K} \mathbf{V}_{\mathrm{p}} \cdot \mathbf{V}_{ij}, \tag{4}$$

where $S_{\mathrm{p}i}$ gives the score, and $\mathbf{V}_{\mathrm{p}}$ is the prompt embedded as a vector. To sum up $S_{\mathrm{p}i}$ for all prompts, we can get the raw score for $i-$th class:

$$s_i = \sum_{\mathrm{p}} s_{\mathrm{p}i} = \sum_{\mathrm{p}} \sum_{j=1}^{K} \mathbf{V}_{\mathrm{p}} \cdot \mathbf{V}_{ij}, \tag{5}$$

and similar to equation (1), the revenue shared by the $i-$th class, i.e., the $i-$th data provider can be calculated by

$$\mathscr{R}_i = \mathscr{R}_{\mathrm{tot}} \times S_i, \tag{6}$$

where $\mathscr{R}_i$ is the revenue shared by $i-$th provider, and S_i is the provider's normalized score calculated by

$$S_i = s_i / \sum_i s_i \tag{7}$$

Keep in mind that, as shown in Figure 4, different text embedding techniques can yield vastly different text similarity computations, and thus result in different scores for each data provider. Currently, there is no general answer as to which text embedding technique is best suited, we can choose the text embedding between universal and model-based methods based on business requirements or other particular purposes.

Furthermore, when considering not just text embeddings but also more general embeddings such as image embeddings, although the techniques for embeddings and computing similarity can differ, the general approach for establishing a scoring system via document similarity should be similar.

Figure 6 represents a modified version of Figure 3, illustrating the relation between an AI tool, its users, data providers, and a scoring system which is the model for revenue sharing. This figure serves as a summary of Sections IV-B, IV-C, IV-D. The scoring system is prompt-based, which each data provider's final score being a combination of the pre-trained classifier's probability score and the text similarity calculator's similarity score. It is worth nothing that the scoring system is independent of the AI tool/model, although it may utilize the embedding technique provided by the AI tool.

E. Complexity of Scoring Systems

Now let us discuss the complexity and cost of the two scoring methods: (1) a pre-trained text classifier, or (2) calculating text similarity with some text embedding techniques.

The first scoring system requires training a classifier model on the entire training dataset, which can be time-consuming. However, training a tree-based classifier is always less expensive than a semi-supervised deep learning model using the

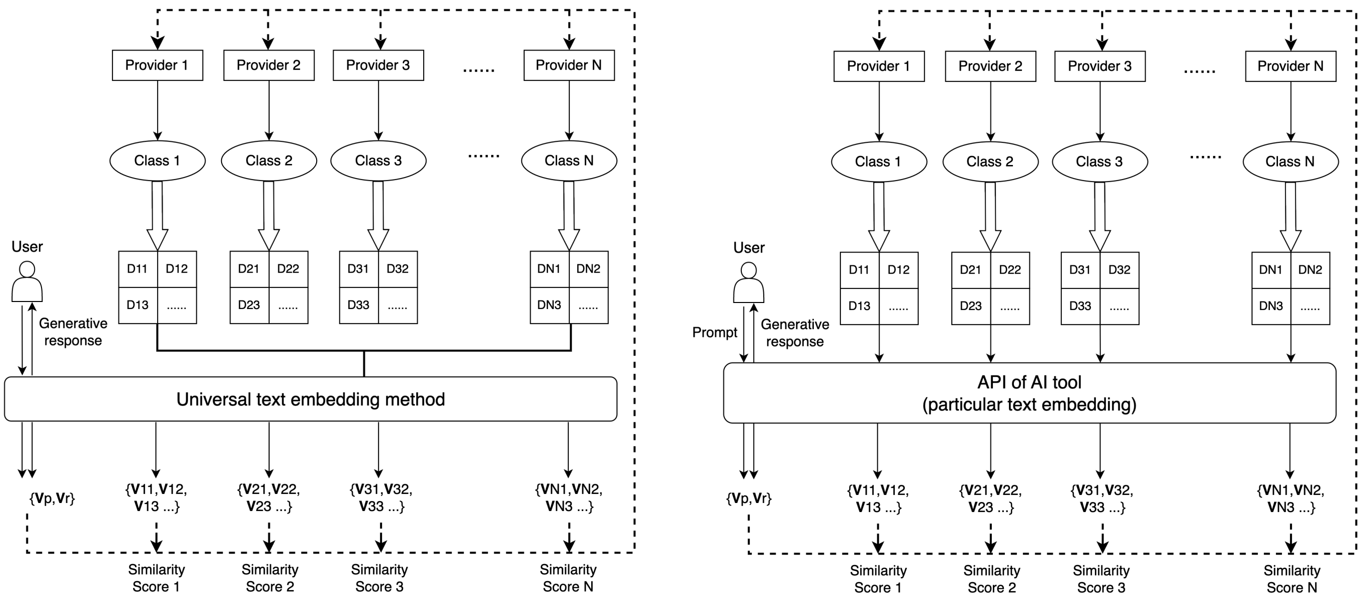

Fig. 5: The scoring system based on text similarity. Documents can be vectorized using two methods of text embedding: universal text embedding method (left panel) and tool-based particular text embedding method (right panel). A user's prompt and the generative response from the AI tool can be vectorized using the same methods, and calculate the average similarity between documents in each class and prompt/response. The average similarity scores are used to establish the scoring system.

same data size. For example, the preprocessed training data for GPT-3 is 570 GB, we expect that combination of text embedding and classification training will cost approximately $\sim$\$1000 -\$2000. Once the classifier is completed and becomes a pre-trained model, each prompt input and the response from the AI tool can quickly generate probability scores for all data providers. The complexity of adding up all probability scores given by prompts is linearly related to the number of prompts $\propto O(P)$, where P is the number of prompt inputs.

The scoring system based on text similarity is more complex. First, if we decide reduce dimensions after text embedding, the work is more complex than a supervised classification model.[14] For each prompt input and its corresponding response, we need to go through all the embedded documents as vectors, to calculate the similarity between each class in the training set and the prompt input with its response. The complexity of adding up the text similarity provided by a large number of prompts given by equations (5) and (6) is $\propto O(NP) \gg O(P)$, where N and P are the number of documents in training set and the number of prompt inputs respectively. For large AI models such as LLMs, the training datasets may have millions or even billions of documents, and the number of prompts from customers on a daily basis is at least in the billions. Therefore, using text similarity scores to build a scoring system is significantly more complex and expensive than the scoring system based on a classifier.

However, we can reduce the complexity of text similarity calculation through some methods. For example, by performing certain weighted averages, we can condense each class of the AI tool's training dataset into a single vector, instead of having a large number of vectors corresponding to its documents. Assuming there are K documents in the $i-$th class of the training dataset, and the documents are embedded as vectors $\{\mathbf{V}_{ij}\}$ with $j \in [1, K]$, the characteristic vector for the $i-$th class can be averaged by

$$\begin{aligned}\langle \mathbf{V}_i \rangle &= \frac{1}{P}\sum_{j=1}^{K}\sum_{p=1}^{P}\mathbf{V}_{ij}\cdot\frac{\mathbf{V}_p}{|\mathbf{V}_p|} \\ &= \frac{1}{P}\sum_{p}\frac{s_{\mathrm{p}i}}{|\mathbf{V}_p|}, \end{aligned} \tag{8}$$

which is the averaged raw score per prompt for the $i-$th class. If P is sufficient large, we can use $\langle \mathbf{V}_i \rangle$ as a characteristic vector for the $i-$th class. For new coming prompts we only needs to calculate $s_{\mathrm{p}i}^{\mathrm{new}} \approx \mathbf{V}_p^{\mathrm{new}} \cdot \langle \mathbf{V}_i \rangle$, thereby reducing the computational complexity to $\propto O(C)$ per input, where C is the number of classes. The overall computational complexity would then be $\propto O(CP) \ll O(NP)$.

F. Real-time Scoring System

Retrain a large AI model can be challenging and time-consuming. It could take months to collect new data, design new architectures and retrain large models. For instance, the first generation pre-trained model of the GPT family, GPT-1, was launched in 2018, while GPT-2 was released in 2019,

[14] Consider a tree-based classifier, the time complexity of the training is $\sim O[KFN\log(N)]$, where K, N and F are the number of trees, the number of documents, and the number of the features (dimensions) respectively, while the space complexity is $\sim O(KNF)$. On the other hand, the text embedding gives a time complexity of $\sim O(F^3)$, where $F \gg K$ and $F \gg N$ if F is obtained from word count and TF-IDF, which means building text embedding models are much more time consuming than supervised classification.

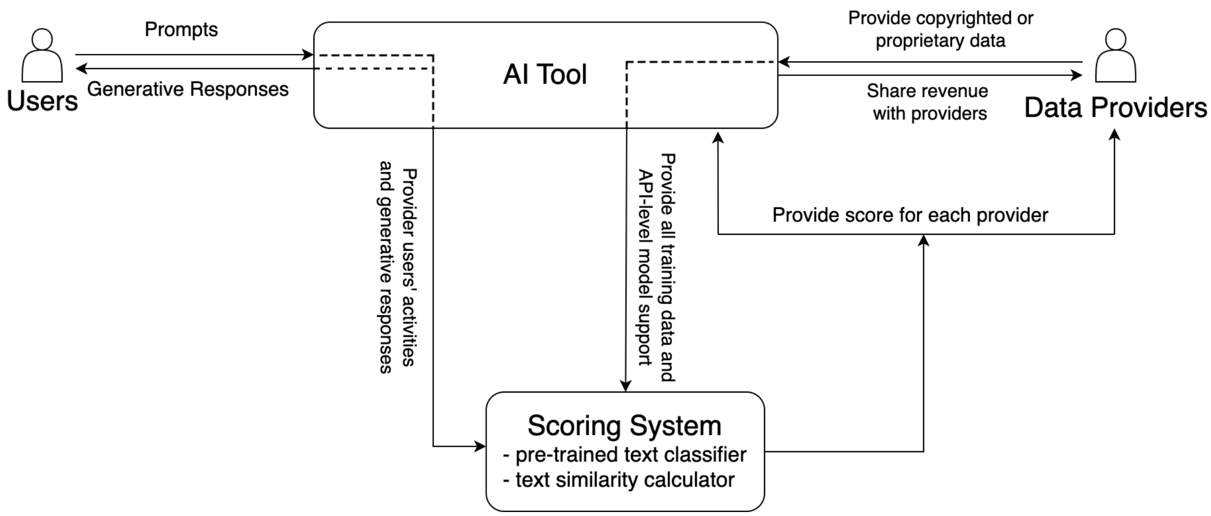

Fig. 6: Illustration of the prompt-based scoring system model and the relation between the AI tool, its users, data providers and the scoring system. Users input prompts and receive generative responses from an AI tool. In order to effectively evaluate data providers and allocate revenue accordingly, the AI tool must provide all of its training data with possibly API-level support to the scoring system, so the scoring system can use this information to build a pre-trained text classifier to obtain "probability score" (Section IV-C), or text similarity calculator for "similarity score" (Section IV-D), while text embeddings are based on model API support. Furthermore, the users' prompts and the AI tool's responses must also be sent to the scoring system, allowing the system to calculate a score for each data provider for each prompt. The sum of these scores across all input prompts will be used to measure the degree of data engagement of each provider. Revenue sharing will then be based on this scoring system, ensuring that data providers are fairly compensated based on their data contributions.

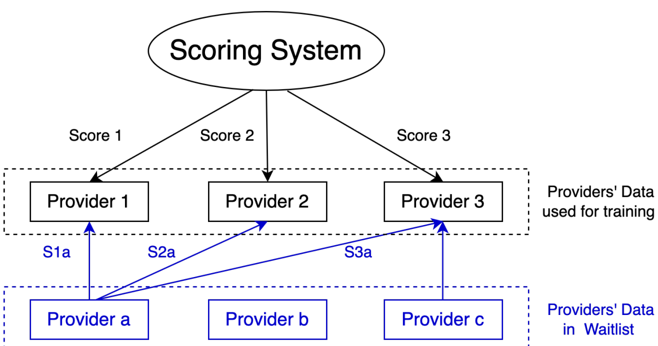

Fig. 7: The AI tool's revenue sharing program includes data providers on the waitlist (blue providers) who have joined but whose data has not yet been used for training. These providers can compare their text similarity with providers whose data has been included in the training dataset (black providers), and use equation (9) to generate a score.

GPT-3 in 2020, ChatGPT (GPT3.5) in 2022, and GPT4 in 2023. This shows that we cannot have real-time models, at least with current technological capabilities.

If a data owner is willing to provide data for a large AI tool, it is likely that the provider must wait at least several months until the next training cycle of the AI model, when the provider's data is truly included in the training dataset of the refreshed AI model. Before that, the provider's data is only on a "waitlist" waiting to be used.

Can the data providers on the waitlist also share the revenue generated by the AI model?

The answer is yes. Just like website owners who join Google AdSense can immediately share revenue from Google ads, there is no reason why providers who have submitted data to an AI tool should have to wait until the next training cycle to officially join the revenue sharing project. Here, we provide an idea how to calculate the score of a data provider before their data is officially used. Figure 7 gives the illustration. Assuming the i-th class in the formal training dataset has an normalized score of P_i, then the $j-$th data provider on the waiting list can be calculated by

$$W_j = \sum_i P_i S_{ij}, \tag{9}$$

where S_{ij} is the similarity between $j-$th provider's data on the waiting list and $i-$th class d8ata in the training dataset, and the scores are normalized by

$$\sum_i P_i = 1, \sum_i S_{ij} = 1. \tag{10}$$

Note that P_i is a prompt-based score, so equation (9) can also provide a prompt-based score for data providers in the waitlist. If the AI tool is willing to share revenue with the data providers on the waitlist, W_j shows a method for revenue distribution.

G. Third-Party Transfer Learning vs. Centralized Dataset

Due to data privacy or other reasons, some data owners may want to use a certain AI tool, but are not willing to provide data to the company or organization who owns the AI tool. These data owners may have a small amount of data that is not enough to train a model from scratch, but they can use a technique called transfer learning to leverage the knowledge that a large AI tool has already learned from its massive training data. Transfer learning uses a data owner's special data to fine tune the parameters of the pre-trained AI tool, so that the tool can be adjusted and used for the data owner's specific task.

If the performance of transfer learning is good enough, a data owner may be able to leverage the AI tool to achieve a specific task without submitting the data to the AI tool. In this case, the data owner is obviously not a data provider of the AI tool, and we cannot use the scoring system mentioned above to score such a data owner. So, which is better? Transfer learning or data owners providing their data to the AI tool and directly obtaining generative responses from the tool?

In fact, this is a comparison between two scenarios: the scenario that data owners remain third-party, and the scenario that the centralized AI tool has access to more and more data and possibly become "big brother". Clearly, from a technical standpoint, the latter scenario will achieve at least no worse model performance than the first scenario[15] So from the perspective of model performance alone, the latter scenario is definitely better. Moreover, if a third party data owner does transfer learning alone, it can only benefit itself, while giving data to the AI tool can also benefit everyone who uses the tool. From the perspective of AI utilitarianism, the latter scenario is also better. Therefore, a good plan for a third party data owner is that the owner can submit data to the AI tool to become the tool's data provider, and join the revenue sharing program. While the provider is still on the waitlist (Section IV-F), it can temporarily use transfer learning technique to obtain generative responses based on the AI tool, and when the provider's data is officially used for AI tool's next iteration of training, the provider can obtain updated responses with better performance, benefit from the revenue-sharing program, and also benefit the public.

On the other hand, if the third party has data privacy or other legal considerations, and how to prevent the AI tool which acquires more and more data from becoming a "big brother", these topics need to be discussed from a legal perspective, which is beyond the scope of this paper.

V. Discussion: Revenue-sharing model for AI Tools other than LLMs

The above discussion regarding the revenue-sharing business model is generally applicable to AI tools, but it has been primarily discussed in the context of LLMs, particularly using the GPT family as case studies. Now let us look into some AI tools in the field of computer vision.

A. AI Text-to-Image Generators

As mentioned in the Introduction (Section I), currently there are a couple of popular "text to image" AI image generators. Dall-E was launched in January 2021, which can be considered as a milestone event. Although its training dataset was not publicly available, we know that the training data for text-to-image models typically consists of text-image pairs. OpenAI has provided an algorithm called CLIP (Contrastive Language-Image Pre-Training), which can embed both text and images simultaneously [24].

Following Dall-E, the year 2022 could be marked as the rise of AI text-to-image generators. In that year, OpenAI released Dall-E 2 in April, Midjourney launched the first version in July, followed by Stability AI's Stable Diffusion released in August, and Google's Imagen in November. In 2023, new launched AI text-to-image generators so far included Google's Parti, Starryai, Dream by WOMBO, and so on.

As of today, many AI text-to-image generators have not publicly opened their training datasets. We know that the training dataset for Dall-E 2 was composed of 12 million image-text pairs. The more recently text-to-image generators used billions even trillions of image-text pairs. There are various reasons why these companies have not publicly opened the datasets. One reason is that the datasets may contain proprietary data. Another reason may be that those companies are concerned about the potential for misuse of the data. There is also a reason that companies may believe that keeping the training dataset private gives them a competitive advantage over others.

Stable Diffusion is one of the few text-to-image generators that has transparent model with opened its training dataset, which was taken from LAION-5B, a publicly available dataset derived from Common Crawl consisting of 5.85 billions CLIP-filtered pairs of images and caption [25]. The dataset was created by LAION projects, and Stable Diffusion was trained on at lease three subsets of LAION-5B including laion-high-resolution, and laion-aesthetics v2 5+. An analysis conducted by a third party, which indexed a sample of 12 million images, revealed that almost half of the images ($\sim 47\%$) were obtained from around 100 domains. The largest number of images came from Pinterest, followed by user-generated content platforms such as WordPress, Smugmug, Blogspot, Flickr, DeviantArt, Wikimedia Commons and Tumblr [16].

There is an important issue: whether AI image generators use copyrighted images in their training datasets without the owners' permission? Although many image generators' training datasets are still a black box, people can still use some methods to check the similarity of images and detect if their copyrighted work has been used for training. An interesting

[15] Because after receiving the data, the AI tool can first do transfer learning for that special task, which ensures that to provide same performance as that done by the data owner itself. Then the AI tool can do something more powerful than the third party owner: it can add the data to its whole training dataset to do various training including semi-supervised and reinforcement learning to further improve the tool's performance on the task.

[16] The third party analysis is from Exploring 12 Million of the 2.3 Billion Images Used to Train Stable Diffusion's Image Generator, and the 1.2 million images with domains can be seen here laion-aesthetic-6pls.

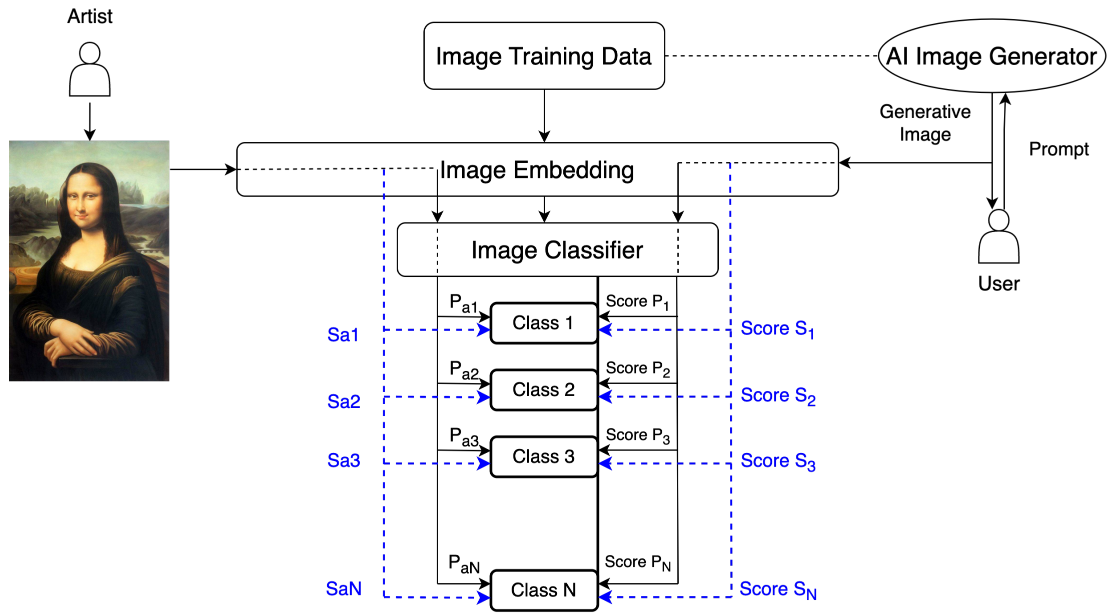

Fig. 8: The prompt-based score systems are used by image providers for an AI image generator, including the score system based on image classification and probability scores (black lines under "Image Classifer") and image similarity scores (blue lines under "Image Embedding"). Each artwork/image is given a score to highlight its contribution. To train a particular AI image generator, we can utilize an image embedding technique (e.g., CLIP) to convert images into vectors, and then build an image classifier based on all training images, where the classes can be based on artistic styles, themes, genres, and so on, and can be multi-task. Suppose an artist provides an artwork (Mona Lisa in this Figure) to the training dataset, the image classifier gives a series of probability scores for this artwork (P_{a1}, P_{a2}, P_{a3}...), and the averaged similarity between each class and the artwork (S_{a1}, S_{a2}, S_{a3}...). For a user input prompt, an image is generated based on this prompt. We can also have a probability score set for this generative image (P_1, P_2, P_3...), and a similarity score set (S_1, S_2, S_3...). The final score of the artwork can be calculated based on these scores (see Equation [13]), and is still prompt-based.

website called "Have I been trained"[17] is designed to check whether your artwork has been included in the LAION-5B dataset. Currently, AI image generators, as a new technology, have already faced some legal disputes. As mentioned in Section I, a group of artists have sued Stability AI, Midjourney and DeviantArt for copyright infringement. In addition, Getty images has filed a lawsuit against Stability AI, accusing Stable Diffusion of "brazen infringement of Getty Images' intellectual property on a staggering scale" and misusing more than 12 million Getty photos to train its Stable Diffusion AI image-generation system[18]. Faced with the issue of copyright infringement, AI image generator companies have emphasized that they will comply with the Digital Millennium Copyright Act (DMCA) and protect the copyrights of image owners.

Meanwhile, the U.S. Copyright Office has taken the position that AI-generated images do not qualify for copyright protection, as they are not the result of human authorship and therefore do not meet the definition of originality[19]. We hope that AI image generators will be better regulated in the near future. To ensure that every artist and copyrighted work to be respected, AI image generators must to be more transparent to disclose their training datasets. However, at the same time , we also expect the revenue-sharing business model discussed earlier based on LLMs to be applied to the filed of AI-generated images. Therefore, the connection between human artists, copyright owners, and AI image tools should not be viewed as a hostile, zero-sum game but rather as a collaborative and mutually beneficial relationship.

Similar to the revenue-sharing business model of LLMs, we can also establish a scoring system for image providers of AI image generators, which is based on some image classification models and image similarity calculators. In principle, as long as we replace "text" with "image" in Figures 3, 5, 6 from Section IV, we can use some image embedding techniques (such as CLIP) to build an image classifier and image similarity calculator on the image databases for a certain AI image generator, and establish a scoring system to measure the degree of engagement for each image provider's images. However, different from LLMs, image owners/providers may be more concerned about the engagement of each of their artwork in the image generator, rather than just the overall engagement of their all works. Additionally, it may be more reasonable to

[17]This is the link of this tool: https://haveibeentrained.com

[18]See the report Getty Images sues AI art generator Stable Diffusion in the US for copyright infringement.

[19]US Copyright Office decided that AI generated works are not eligible for copyright, see the statement of policy, but AI-assisted work may still be copyrighted.

classify the image database by art style, genre, topic rather than by image providers. Therefore, the image classifier may be multi-tasking.

Figure 8 shows the scoring systems that AI image generators can reply on for their revenue-sharing business model. In general, image scoring systems are very similar to the text scoring systems discussed in Section IV, but we highlight just one artwork by an artist in Figure 8, and provide a method for measuring the importance of that particular artwork. Assuming the images used for a certain AI image generator can be divided into N categories to build an image classifier, the probability scores of an artist's artwork (e.g., Mona Lisa in Figure 8, hereafter ML) for each class of the image classifier are $\{P_{ai}\}$ $(i = 1, 2, 3...N)$, the averaged image similarity score between each class and the artwork (ML) is $\{S_{ai}\}$. For one generative image created by the AI image generator for a user's text prompt, the probability scores of all classes for this generative image are $\{P_{ij}\}$, and the averaged similarity with each class is $\{S_{ij}\}$, where j means the score sets are prompt-based and for the $j-$th prompt. Here we assume all score sets are normalized. Then, based on the image classifier, the probability score of this artwork (ML) is calculated by:

$$P_{\mathrm{ML,j}} = \sum_{i=1}^{N} P_{ai} P_{ij}, \tag{11}$$

and the similarity score of ML is

$$S_{\mathrm{ML,j}} = \sum_{i=1}^{N} S_{ai} S_{ij}, \tag{12}$$

And the final score of this artwork ML is calculated by

$$P_{\mathrm{ML}} = \sum_{j} \sum_{i=1}^{N} P_{ai} P_{ij}, \quad S_{\mathrm{ML}} = \sum_{j} \sum_{i=1}^{N} S_{ai} S_{ij}. \tag{13}$$

Note that image similarity scores are more straightforward if the image embedding method has been selected. For image classifier, if the classifier is multi-task (for example, Mona Lisa can be multi-labeled by Renaissance, oil painting style, Da Vinci, female artwork, etc.), we can change probability scores for each class $\{P_{ai}\}$ and $\{P_i\}$ mentioned above be a set of two-dimension scores $\{\mathbf{P}_{ai}\}$ and $\{\mathbf{P}_i\}$[20]. and calculate the scores as

$$P_{\mathrm{ML}} = \sum_{j} \sum_{i=1}^{N} \mathbf{P}_{ai} \cdot \mathbf{P}_{ij}. \tag{14}$$

Although the U.S. Copyright Office has taken the position that AI-generated images do not qualify for copyright protection, we can still calculate the contributions of human artists, the AI tool, and user who generates an AI-generated image, and allocate the proportion of copyright accordingly, if copyright can be quantified.

[20]i.e, each $\mathbf{P}_{ai}$ and $\mathbf{P}_i$ are in the form of [probability_score_belong_to_one_class, probability_score_not_belong_to_one_class], where the sum of the pair scores is one.

B. AI in Healthcare

AI may potentially have a significant impact in the healthcare field as well. The most promising applications of AI in healthcare include medical imaging, predictive analytics, drug development, personalized patient care, and so on.

However, the development of AI in healthcare has not been as fast as in other fields. For example, both the well-known IBM Watson Health and Google Health both encountered challenges. IBM Watson Health was founded in 2015 with the goal of using NLP and machine learning to "transform healthcare with cognitive computing". However, IBM Watson Health aced significant challenges soon after. In 2018, IBM Watson Health underwent a massive layoff, and eventually IBM sold Watson Health in 2022. Google launched Google Health in 2008, and a new division called Google Health AI in 2018. Similar to IBM Watson Health, Google Health also encountered a series of problems and setbacks, leading to the downgrading of Google Health AI to Google Research in 2022.

Overall, healthcare AI faces several big challenges [26]. Firstly, how to build trust between doctors/patients with AI tools' results? Explainable AI may help to make AI tools less of a black box, but there are still debates and criticisms about the role of Explainable AI in the medical realm [27]. Secondly, the lack of sufficient data for AI tools in healthcare has limited their performance. In healthcare, collecting, managing and sharing data is complicated and difficulty due to privacy concerns, regulations, technical constraints, cost and disincentives [28]. Obtaining high-quality data user regulatory constraints is still a significant challenge.

The revenue-sharing model for AI tools I proposed may have significant potential in other fields, but could we also introduce a revenue-sharing model to gather more data in healthcare? This is a question worth exploring.

C. Multimodal AI Tools

OpenAI's GPT-4 is already a large multimodal model. We can foresee that there will be more and more multimodal AI tools in the future. We will not only have tools for text-to-text, text-to-image, image-to-text, text-to-music, text-to-video and so on, but also hybrid tools, for example, a tool to input both text and image, and output text, music, and video simultaneously. For multimodal tools, how can we establish a revenue-sharing scoring system?

Following the discussion in Section IV-A, we want to ensure that the revenue sharing for multi-modal tools is still based on prompts. One approach is to examine the generative content of the AI tool, classify it into different modalities, and assign scores to each class for the corresponding modality. For example, if the generated content contains both text and images, we can use the method discussed in Sections IV-C and IV-D to score each class in the text training dataset and the method in Section V-A to score each class in the image training dataset. Finally, we can sum up the scores generated by all generative content to obtain scores for all data providers of the multi-modal tool.

VI. Conclusion

ChatGPT has caused a storm in the AI world, and a new era of AI is about to come. We can foresee that exceptional AI tools will soon reap considerable profits. However, we must ask a question: should AI tools share revenue with their data providers? The short answer is: Yes.

For deep learning models, especially large models such as Large Language Models (LLMs), we need to acquire more and better quality data to achieve better model performance. Here "more and better data" not only means collecting a large training dataset for a model, but also has at lease four meanings: (1) Enhancing the diversity and amount of training data can improve models' learning ability and decrease the chances of errors. (2) Using data specific to a certain task of domain knowledge can fine tune the model for the task or domain. (3) Keep training data up-to-date to produce more precise and recent results. (4) Incorporating more human feedback to establish better ground truth.

However, due to data privacy and copyright laws, most AI tools often can only collect training datasets in the public domain. Some AI tools may be able to obtain some copyrighted data, but not a lot, while others may have already fallen into copyright infringement issues. Even if the data of AI tools is currently in the public domain, it is highly likely that the some original data owner will lose more and more visitors and customers, and possibly refuse AI tools to use their data. The game between data owners and AI tools look like a zero-sum game. In this paper, we discuss a better way for AI tools to obtain data and benefit all parties, which is to establish a new revenue-sharing business model in the upcoming AI era.

Taking LLMs as an example, in the fierce competition among various models and tools, a successful LLM must have: (1) more and better data than other tools, (2) more powerful computing resources, and (3) faster iterations to update the model to meet customer needs. Just as today's search engine market is dominated by Google search with over 90% market share, one or several best LLMs with their chatbots could potentially dominate the language tool market or even multimodal market in the future. This is called "LLM monopoly," which could pose a great threat to various text data owners. To establish a utilitarian AI era where AI tools can keep improving performance and a wide range of data owners can benefit from it, a revenue-sharing model established by AI tools is necessary.

Of course, establishing a revenue sharing model for AI tools is very challenging. Today's revenue sharing models, such as Getty Images' revenue-sharing model and Google AdSense in the Web 2.0 era, will not work for the AI era. The existing revenue sharing models based on old metrics such as clicks, page views, and conversion rates (Figure 1) will be replaced by new metrics-based business models such as prompts and cost per prompt (Figure 2). In order to fairly allocate revenue with data providers, AI tools must establish a scoring system to score each data provider. For this scoring system, there are several crucial prerequisites: (a) The scoring system must be based on prompts. (b) Neutral networks should not be used to build the scoring system, which should be as simple as possible. (c) The AI tool's entire training dataset should be open and transparent, allowing third-party verification. (d) The scoring system must be as independent from the AI tool as possible.

Specifically, for an AI tool, we can establish a classification system for the various data provided by data providers in its training dataset. f we consider each data provider as an independent class, or a combination of data providers as a class, we can build a pre-trained classification model on this training dataset based on the class (provider) to which each document belongs. Once a user inputs a prompt and receives a generative response from the tool, we can use the pre-trained classifier to assign each class a probability score based on the prompt or the response, so the probability scores reflect the degree of engagement of each class for this prompt. The summation of probability scores for all prompts can be used to establish a scoring system. Figure 3 shows how to use a pre-trained text classifier to assign a score based on prompts to each data provider, and ultimately allocate revenue to AI tools based on such scores. In Section IV-C, we also provide some demonstrations on how to establish a text classifier and its corresponding scoring system.

Moreover, for language tools, we can score data providers by calculating text similarity. Specifically, we can use a universal or tool-based particular text embedding technique to convert documents in the training dataset into vectors, and calculate the average text similarity between each prompt, the generative response from the tool, and each data provider. Each data provider can obtain a prompt-based text similarity score, which can be summed up prompt by prompt, s we can obtain a final score for each data provider, thus establishing a scoring system based on the the text similarity calculator (see Figure 5). Note that different text embedding methods will calculate different text similarity scores. Currently, there is no general answer as to which text embedding technique is best suited, so we can choose the text embedding method between universal and model-based methods based on business requirements or other particular purposes. Additionally, this paper discussed the complexity of different scoring systems and how to simplify them to reduce their complexity (Section IV-E). Moreover, for data providers that have not yet been included in the training dataset but have already joined the revenue-sharing program, we can also establish a temporary rating mechanism for them (Section IV-F). In Section V-A, we discussed how to establish a scoring system for AI image generators and calculate engagement rate for individual artworks (see also Figure 8). Although the U.S. Copyright Office has taken the position that AI-generated images do not qualify for copyright protection, we can still calculate the contributions of human artists, the AI tool, and user who generates an AI-generated image, and allocate the proportion of copyright accordingly, if copyright can be quantified.

The revenue-sharing model used for LLMs can be applied to other AI tools as well. For instance, AI image generators can use a scoring system based on image classification models and image similarity calculators to establish revenue sharing with their image providers. However, different from LLMs, image providers may be more interested in the engagement of their individual artworks rather than their overall portfolio. Furthermore, it may be more appropriate to classify the image database by art style, genre, and topic rather than by image providers.

Finally, we discussed AI in healthcare (Section V-B) and multimodal models (Section V-C). Perhaps we could also establish a revenue-sharing model for AI in healthcare, but this issue requires careful consideration. As for multimodal models, which are expected to become more prevalent in the future, we can establish a multimodal scoring system to score data providers for each modality.

REFERENCES

[1] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, et al., "Language Models are Few-Shot Learners," In *34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, December 2020. arXiv preprint arXiv:2005.14165 (2020).

[2] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, et al., "Training language models to follow instructions with human feedback," arXiv preprint arXiv:2203.02155 (2022).

[3] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever "Improving Language Understanding by Generative Pre-Training," 2018.

[4] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, S. Fidler, "Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books", *Proceedings of the IEEE International Conference on Computer Vision* (ICCV), pp. 19-27, 2015.

[5] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, "Language Models are Unsupervised Multitask Learners," Technical report, OpenAI, 2018.

[6] R. Taylor, M. Kardas, G. Cucurull, T. Scialom, A. Hartshorn, E. Saravia, A. Poulton, V. Kerkez, R. Stojnic, "Galactica: A Large Language Model for Science," arXiv preprint arXiv:2211.09085 (2022).

[7] F. Pereira, P. Norvig, A. Halev. "The unreasonable effectiveness of data," IEEE Intelligent Systems, 2009.

[8] C. Sun, A. Shrivastava, S. Singh, A. Gupta, "Revisiting Unreasonable Effectiveness of Data in Deep Learning Era," arXiv preprint arXiv:1707.02968 (2017).

[9] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. Won Chung, et al., "PaLM: Scaling Language Modeling with Pathways," arxiV preprint arXiv:2204.02311 (2022).

[10] R. Thoppilan, D. D. Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H. -T. Cheng, A. Jin, T. Bos, et al., "LaMDA: Language Models for Dialog Applications," arXiv preprint arXiv:2201.08239 (2022).

[11] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, et al., "OPT: Open Pre-trained Transformer Language Models," arXiv preprint arXiv:2205.01068 (2022).

[12] S. Soltan, S. Ananthakrishnan, J. FitzGerald, R. Gupta, W. Hamza, H. Khan, C. Peris, S. Rawls, et al., "AlexaTM 20B: Few-Shot Learning Using a Large-Scale Multilingual Seq2Seq Model," arXiv preprint arXiv:2208.01448 (2022).

[13] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M. -A. Lachaux, T. Lacroix, B. Roziére, N. Goyal, et al., "LLaMA: Open and Efficient Foundation Language Models," arXiv preprint arXiv:2302.13971 (2023).

[14] S. Smith, M. Patwary, B. Norick, P. LeGresley, S. Rajbhandari, J. Casper, Z. Liu, S. Prabhumoye, et al., "Using DeepSpeed and Megatron to Train Megatron-Turing NLG 530B, A Large-Scale Generative Language Model," arXiv preprint arXiv:2201.11990 (2022).

[15] S. Wang, Y. Sun, Y. Xiang, Z. Wu, S. Ding, W. Gong, S. Feng, J. Shang, et al., "ERNIE 3.0 Titan: Exploring Larger-scale Knowledge Enhanced Pre-training for Language Understanding and Generation," arXiv preprint arXiv:2112.12731 (2021).

[16] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de Las Casas, L. A. Hendricks et al., "Training Compute-Optimal Large Language Models," arXiv preprint arXiv:2203.15556 (2022).

[17] J. W.. Rae, S. Borgeaud, T. Cai, K. Millican, J. Hoffmann, F. Song, J. Aslanides, S. Henderson, et al., "Scaling Language Models: Methods, Analysis & Insights from Training Gopher," arXiv preprint arXiv:2112.11446 (2021).

[18] S. Black, S. Biderman, E. Hallahan, Q. Anthony, L. Gao, L. Golding, H. He, C. Leahy, et al., "GPT-NeoX-20B: An Open-Source Autoregressive Language Model ," arXiv preprint arXiv:2204.06745 (2022).

[19] Y. Bai, S. Kadavath, S. Kundu, A. Askell, J. Kernion, A. Jones, A. Chen, A. Goldie, et al., "Constitutional AI: Harmlessness from AI Feedback," arXiv preprint arXiv:2212.08073 (2022).

[20] BigScience Workshop: T. Le Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, et al., "BLOOM: A 176B-Parameter Open-Access Multilingual Language Model," arXiv preprint arXiv:2211.05100 (2022).

[21] P. Liang, R. Bommasani, T. Lee, D. Tsipras, D. Soylu, M. Yasunaga, Y. Zhang, D. Narayan, et al., "Holistic Evaluation of Language Models," arXiv preprint arXiv:2211.09110 (2022).

[22] E. Mosca, F. Szigeti, S. Tragianni, D. Gallagher, G. Groh. "SHAPbased explanation methods: A review for NLP interpretability," In Proceedings of the 29th International Conference on Computational Linguistics, pages 4593–4603, Gyeongju, Republic of Korea (2022).

[23] W. Zhao, T. Joshi, V. N. Nair, A. Sudjianto, "SHAP values for Explaining CNN-based Text Classification Models," arXiv preprint arXiv:2008.11825 (2020).

[24] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, et al., "Learning Transferable Visual Models From Natural Language Supervision," arXiv preprint arXiv:2103.00020 (2021)

[25] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, et al., "LAION-5B: An open large-scale dataset for training next generation image-text models," arXiv preprint arXiv:2210.08402 (2022)

[26] A. Goldfarb, F. Teodoridis, "Why is AI adoption in health care lagging?", brookings Report, Article Link, (2023)

[27] G. Cinà, T. Röber, R. Goedhart, I. Birbil, "Why we do need Explainable AI for Healthcare," arXiv preprint arXiv:2206.15363 (2022)

[28] D. McDuff, T. Curran, A. Kadambi, "Synthetic Data in Healthcare," arXiv preprint arXiv:2304.03243 (2023)