Interactive Tool for Tracking Open-source Artificial Intelligence Progress on Hugging Face

Bogdan Šinik
bogdan.sinik@famnit.upr.si
UP FAMNIT
Koper, Slovenia

Domen Vake
domen.vake@famnit.upr.si
UP FAMNIT
Koper, Slovenia

Jernej Vičič
jernej.vicic@upr.si
UP FAMNIT, UP IAM
Koper, Slovenia

Aleksandar Tošić
aleksandar.tosic@upr.si
UP FAMNIT, InnoRenew CoE
Koper, Slovenia

Abstract

Given its increasing importance in our daily lives, Artificial Intelligence has become a prominent subject that needs extensive investigation and understanding. This study presents an analysis of the open-source community in the field of Artificial Intelligence (AI). Various questions arise anytime AI is introduced. open-source AI introduces additional concerns. Should artificial intelligence (AI) be universally accessible, or should it be restricted to private use? Is it worthwhile to offer basic models to the broad user population? We chose the most important data from the primary website in the field, Hugging Face. We have developed a tool that allows for straightforward monitoring of the progress of various open-source AI models using data obtained from their leader board. The platform offers accessible and valuable information about various AI models, including their architectures and the activities of authors. Through performing a quick review with our tool, it becomes evident that the open-source community is becoming large and has an undeniable impact on the AI community.

Keywords

LLM, open-source, AI, Hugging Face

1 Introduction

Artificial intelligence, particularly large language models (LLMs), is an important topic in the computer industry today. Despite the numerous fears and dogmas around it, it is certain that AI has become an integral aspect of our lives. This research has specifically concentrated on the development of a tool for monitoring the impact of the open-source community in the area of artificial intelligence. As implied, these models are accessible to all individuals. There is considerable debate on whether this type of technology should be universally accessible. We wanted to investigate if the open-source community is actively contributing to the development of the field, regardless of one's philosophical convictions. Due to the substantial computational requirements, it was previously impossible to execute Large Language Models on personal computers. As increasingly compact versions with impressive capabilities are being produced, this scenario undergoes a significant transformation. Currently, it is feasible to execute your own model, as long as it is of a modest enough size, on a home computer's graphics processing unit (GPU), even if the GPU is a few years old [9]. The rise in accessibility also enables a larger community to test and develop new solutions and build on top of existing models. We believe that there is a big lack of tools for monitoring the impact of this movement.

Hugging Face [1] has grown into one of the primary platforms for the open-source community. Users are able to download and interact with all significant open-source models. Subsequently, users have the option to publish their models on the platform and compare their performance by adding them to the leaderboard, where all the models are benchmarked and ranked. The open-source community relies heavily on the distribution of models by large corporations, as creating a model from scratch is a hard undertaking [9]. This tool facilitates collaboration among open-source contributors, enabling them to collectively generate social media content, exchange ideas, and even publish concise articles. In addition to the models, they have the ability to generate and upload useful datasets. It represents the most advanced and innovative developments in the field of open-source AI and Machine learning.

An issue that has been observed is the absence of effective visualization tools on Hugging Face, which would enable users to easily see patterns and gain a comprehensive understanding of the open-source AI area. In order to address this issue, we have developed a sophisticated tool that offers users various viewpoints on the data.

2 Literature review

Large Language Models (LLMs) have proven essential in enhancing software engineering (SE) tasks, demonstrating their effectiveness in code comprehension Similar to conventional software engineering tools, open-source cooperation is essential for achieving superior products in this area. [8]

The article authored by Patel et al. [9] emphasizes the significance of the open-source AI community and elucidates its rapid growth in the wake of major industry leaders like Google, Microsoft, and OpenAI. An important milestone in this subject is often emphasized as the day when the LLama model was initially made available to the open-source community. The community promptly recognized the possibilities and potential involved in this release.

Due to its continuous growth, Hugging Face has emerged as the primary platform for exchanging machine learning (ML) models, resulting in an increasing level of complexity. A relational

Information Society 2024, 7–11 October 2024, Ljubljana, Slovenia

https://doi.org/https://doi.org/10.70314/is.2024.sikdd.1

[1] https://huggingface.co/

database called HFCommunity was established to facilitate the analysis and resolution of this issue [1].

As previously said, open-source AI models offer an extensive range of possibilities. At the recent conference, the authors [12] demonstrated their effective use of Hugging Face. Due to the significant difficulty in developing a model with broad intelligence, researchers have merged ChatGPT capabilities with models from HuggingFace using agentic architecture to get impressive results in multiple domains. ChatGPT was tasked with creating a plan of action and assigning specific duties to each open-source model based on their own areas of expertise. This is an excellent demonstration of the influence and capabilities of the open-source community, given the familiarity with open models and their capabilities.

The article [6] examines the vulnerabilities associated with open-source AI. A much higher number of repositories with high vulnerabilities has been discovered compared to those with low vulnerabilities, particularly in root repositories. This emphasizes the significance of ensuring the security of technology in order to facilitate its utilization.

In a recent paper [10], authors have analyzed the transparency of Hugging Face pre-trained models regarding database usage and licenses. The analysis revealed that there is often a lack of transparency regarding the training datasets, inherent biases, and licensing details in pre-trained models. Additionally, this research identified numerous potential licensing conflicts involving client projects. 159,132 models were examined. It was found that merely 14% of these models explicitly identify their datasets with specific tags. Furthermore, a detailed examination of a statistically significant sample comprising 389 of the most frequently downloaded models showed that 61% documented their training data in some form.

3 Methodology

We obtained the data by extracting the Open LLM Leaderboard from Hugging Face [2] by saving the data server sent to the client. This data contains information about repositories of models that are currently on the leaderboard and the models that are waiting to be evaluated for the leaderboard. A Python pipeline was developed to clean and enrich this data available on [2]. The leaderboard data includes model architecture and precision as well as the model type and performance on the following benchmarks: ARC[3], HellaSwag[14], MMLU[5], TruthfulQA[7], Winogrande[11] and GSM8K[4]. In addition to the data provided on the leaderboard, additional information on the given models was obtained by using the HF API client. This included data about repository contributors, tags, base models, used datasets, and repo activity. It is important to note that the data is self-reported by the developers and is not enforced by HuggingFace. Additionally, the leaderboard includes duplicates due to developers being able to replace models in the repository with different models under the same name. This means the duplicates have the same repository data but distinct performances. Due to the inability to programmatically determine the current model in the repository, we chose the best-performing model under the repository name as the model representing the repository when removing duplicates. Thus, all datasets were generated for further utilization. The following analysis was conducted using the R programming language. The data was mostly studied via the perspective of time, as our focus was on identifying any obvious trends. The data was categorized using several criteria, such as model type, model architecture, and amount of parameters. The data was initially selected and aggregated to ensure that all crucial components were easily accessible. All models that were categorized as flagged have been excluded from the dataset. In addition, we have collected data on the authors' activities and conducted a study on that particular aspect. Once the data had been cleaned and prepared for visualization, we utilized the R ggplot library to create visual representations of the data. A comprehensive R Shiny app was developed by aggregating all the visuals. We chose to utilize Shiny because it is a great option for constructing interactive data analysis solutions due to several factors. Firstly, it enables the development of web applications that are capable of responding and adapting to real-time changes and user interactions. This simplifies the process of exploring and analyzing data. Shiny easily incorporates with R, utilizing its robust statistical and graphical functionalities to generate complex, interactive visualizations without the need for experience in web technologies such as HTML, CSS, or JavaScript. [13] Finally, our application was deployed to a server, making it accessible online.

4 Results

The outcome of this study is the tool we have developed. The link may be accessed via the following URL. [3]. It has six distinct viewpoints, all conveniently accessible inside its tab. The initial figure, labeled as 1, displays both the count of new models and the distribution of various model types. Hugging Face has identified five distinct categories of models: basic mergers and moerges, fine-tuned on domain-specific datasets, chat models, continuously pretrained models, and pretrained models. If the model did not belong to any of these classes, its type was classified as unknown. The user has the ability to effortlessly choose their preferred categories, along with the desired time frame and unit of aggregate (daily, weekly, or monthly). This allows the viewer to clearly observe the evolution of model types and their popularity over time. It is evident that fine-tuned models are predominantly utilized. This is logical, as users are adapting base models by training them on unique datasets to achieve specialization. Also, we can see that merged models are a relatively recent phenomenon.

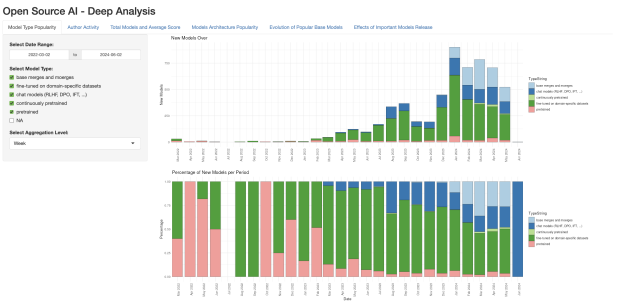

Figure 1: Popularity by model type over time

The second view, referenced as 2, has two interconnected visualizations. The upper section displays the activity of the top 10 authors within a specific range of dates. The display showcases every model they have developed, along with its corresponding type. The lower section presents the average benchmark score for

[2]https://github.com/VakeDomen/HF_analysis

[3]https://oai.dltlt.famnit.upr.si/

each model, organized by author. This visualization enables users to effortlessly monitor the most prominent authors and observe their patterns and accomplishments in model development over time. Users have the ability to effortlessly choose a certain range of dates and also narrow down the list to the top 10 authors according to their preferences. It is evident that leading authors typically do not adhere to trends and consistently provide models of similar type.

Figure 2: Top authors activity over time

The following perspective 3 illustrates two aspects. The first aspect is the alteration in the average benchmark score for each model type as time progresses. The display showcases the top-performing model for each category and time interval (daily, weekly, or monthly). In addition to the dots representing each model, we have incorporated a smooth line to aid the user in seeing the temporal changes for a particular model type. Following the first visualization, we have included a second visualization that displays the total number of models for each model type within the chosen period range. Through these visualizations, users can easily identify the model type that experienced the most improvement and the model types that were mainly produced. We can see the trend, which indicates that open-source AI models are improving, as evidenced by the improvement in average benchmark scores across most of them. The overall number of models is rapidly increasing, indicating a rise in the popularity of open-source AI models.

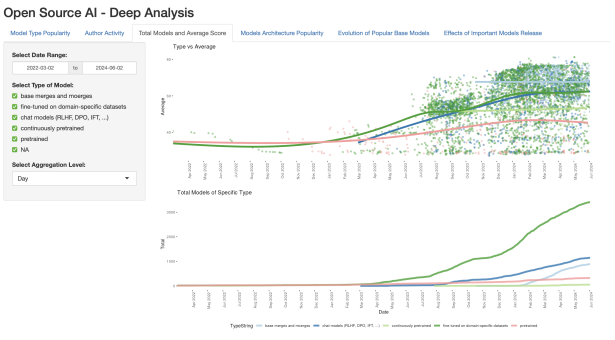

Figure 3: Change of benchmark score and total models per type over time

The fourth perspective, as seen in Figure 4, examines the changing popularity of various model architectures throughout time. The following architectures have been chosen for this specific objective: LLama, Mixtral, Mistral, Qwen2, Gemma, Phi, Opt, GPT2, and GPT2-NeoX. All architectures that did not fit into any one category were classed as "Other". This perspective has two graphics that depict popularity. The first comparison assesses the popularity of a model relative to itself, depending on the number of new models introduced before. The second one compares it to the average number of new models created, taking into account their architecture. Both are depicted by coloring the area, as it is the most convenient way to track. Users may analyze the fluctuation in popularity of well-known model architectures over time and examine how the rising popularity of a particular architecture might impact the popularity of a certain architecture of interest. The lower plot indicates that LLama and Mistral are the predominant models; nonetheless, they have experienced fluctuations throughout time, as visible on the upper plot.

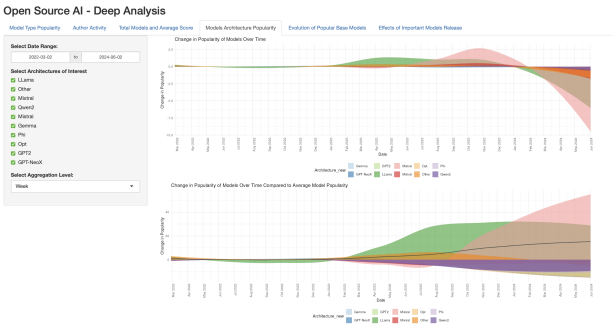

Figure 4: Change of popularity of main architectures over time

The graphic labeled as 5 illustrates the progressive improvement of the key base models developed by famous companies. This was accomplished by isolating each incremental improvement in score over time, using the base model as a reference. In order to fulfill this objective, we have chosen five distinct variations of LLama, Mistral, and Mixtral, as well as three iterations of Phi. The user may easily observe the overall improvement in benchmark scores for each base model. In addition, users have the ability to view the overall duration required for the model to achieve its maximum performance. We have included a feature that enables users to toggle the visibility of model labels, hence enhancing visibility and facilitating more in-depth examination according to their preferences. This allows the user to observe the speed at which specific models reached their peak performance and the extent of their improvement relative to the base models.

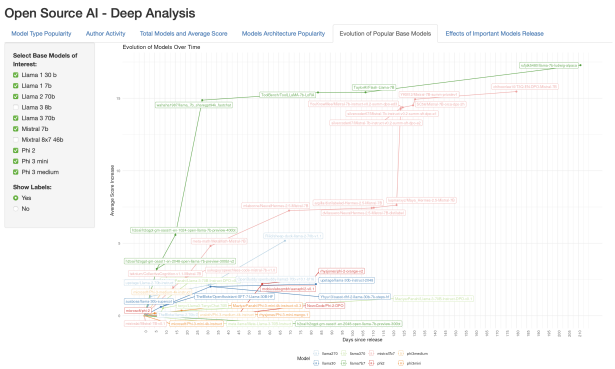

Figure 5: Evolution of famous base models

The final view, as depicted in Figure 6, illustrates the impact of significant releases on the popularity of various model designs. As we have employed identical model designs to those in view four, we have extracted and categorized all significant release dates of these models. The user has the option to choose the time unit for aggregate, which can be either day, week, or month. Users may quickly analyze the impact of significant releases and observe how they influence the popularity and mass creation of specific models. We can observe the evident impact of the recent releases of LLama and Mistral for their popularity.

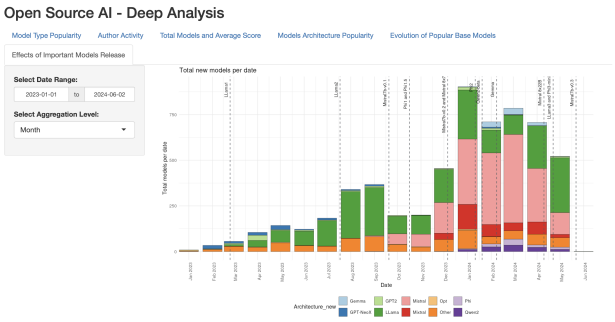

Figure 6: Effect of big releases on architecture of produced models

5 Conclusion and future work

Given the growing importance of Artificial Intelligence in modern culture, it is beneficial to explore the free solutions that are accessible rather than just depending on commercial alternatives. This paper offers valuable insights into a tool designed to simplify the examination of trends in open-source AI in a user-friendly manner. It offers various viewpoints and enables users to acquire knowledge and reach certain conclusions about the subject. Hugging Face has the capability to function as an excellent tool for finding a certain model. As time progresses, open-source AI is expected to provide a growing contribution to the AI community and provide more specific applications for models that could be ignored by big organizations.

We aim to enhance the functionality of our Shiny application by incorporating more perspectives and expanding the range of data interaction options. Our objective is to ensure that the system is as updated as possible. Besides that, we want to conduct a comprehensive analysis of the data to identify patterns and correlations inside this group. We aim to assess the potential of these models and examine their capabilities and potential uses in addressing real-world issues. We would like to analyze the sustained popularity and efficacy of these models over a longer time frame.

References

[1] Adem Ait, Javier Luis Cánovas Izquierdo, and Jordi Cabot. 2023. Hfcommunity: a tool to analyze the hugging face hub community. In *2023 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*, 728–732. DOI: 10.1109/SANER56733.2023.00080.

[2] Edward Beeching, Clémentine Fourrier, Nathan Habib, Sheon Han, Nathan Lambert, Nazneen Rajani, Omar Sanseviero, Lewis Tunstall, and Thomas Wolf. 2023. Open llm leaderboard. https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard. (2023).

[3] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. (2018). arXiv: 1803.05457 [cs.AI].

[4] Karl Cobbe et al. 2021. Training verifiers to solve math word problems. (2021). arXiv: 2110.14168 [cs.CL].

[5] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. (2021). arXiv: 2009.03300 [cs.CY].

[6] Adhishree Kathikar, Aishwarya Nair, Ben Lazarine, Agrim Sachdeva, and Sagar Samtani. 2023. Assessing the vulnerabilities of the open-source artificial intelligence (ai) landscape: a large-scale analysis of the hugging face platform. In *2023 IEEE International Conference on Intelligence and Security Informatics (ISI)*, 1–6. DOI: 10.1109/ISI58743.2023.10297271.

[7] Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: measuring how models mimic human falsehoods. (2022). arXiv: 2109.07958 [cs.CL].

[8] Zhihao Lin et al. 2024. Open-source ai-based se tools: opportunities and challenges of collaborative software learning. *arXiv preprint arXiv:2404.06201*.

[9] Dylan Patel and Afzal Ahmad. 2023. Google "we have no moat, and neither does openai.". *SemiAnalysis. May*, 4, 2023.

[10] Federica Pepe, Vittoria Nardone, Antonio Mastropaolo, Gerardo Canfora, Gabriele Bavota, and Massimiliano Di Penta. 2024. How do hugging face models document datasets, bias, and licenses? an empirical study.

[11] Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. WINOGRANDE: an adversarial winograd schema challenge at scale. (2019). arXiv: 1907.10641 [cs.CL].

[12] Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. Hugginggpt: solving ai tasks with chatgpt and its friends in hugging face. In *Advances in Neural Information Processing Systems*. A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors. Vol. 36. Curran Associates, Inc., 38154–38180. https://proceedings.neurips.cc/paper_files/paper/2023/file/77c33e6a367922d003ff102ffb92b658-Paper-Conference.pdf.

[13] Carson Sievert. 2020. *Interactive web-based data visualization with R, plotly, and shiny*. Chapman and Hall/CRC.

[14] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: can a machine really finish your sentence? (2019). arXiv: 1905.07830 [cs.CL].