Towards an Explorable Conceptual Map of Large Language Models

Lorenzo Bertetto[1][0009−0000−6807−9745], Francesca Bettinelli[2][0009−0000−9526−9923], Alessio Buda[2][0009−0007−5682−8391], Marco Da Mommio[2], Simone Di Bari[1][0009−0003−6868−797X], Claudio Savelli[1][0009−0005−1108−1170], Elena Baralis[1][0000−0001−9231−467X], Anna Bernasconi[2][0000−0001−8016−5750], Luca Cagliero[1][0000−0002−7185−5247], Stefano Ceri[2][0000−0003−0671−2415], and Francesco Pierri[2][0000−0002−9339−7566]

[1] Politecnico di Torino, Torino, Italy – [2] Politecnico di Milano, Milano, Italy

Abstract. Large Language Models (LLMs) have revolutionized the current landscape of Natural Language Processing, enabling unprecedented advances in text generation, translation, summarization, and more. Currently, limited efforts have been devoted to providing a high-level and systematic description of their properties. Today's primary source of information is the Hugging Face (HF) catalog, a rich digital repository for researchers and developers. Although it hosts several models, datasets, and applications, its underlying data model supports limited exploration of linked information.

In this work, we propose a conceptual map for describing the landscape of LLMs, organized by using the classical entity-relationship model. Our semantically rich data model allows end-users to answer insightful queries regarding, e.g., which metrics are most appropriate for assessing a specific LLM performance over a given downstream task. We first model the resources available in HF and then show how this map can be extended to support additional concepts and more insightful relationships. Our proposal is a first step towards developing a well-organized, high-level knowledge base supporting user-friendly interfaces for querying and discovering LLM properties.

Keywords: Conceptual Modeling · Knowledge Graph · Large Language Models · Knowledge Exploration · Knowledge Management

1 Introduction

The recent introduction of Large Language Models (LLMs) has sparked an explosion of interest in generative Artificial Intelligence tools, raising novel opportunities and challenges, but also a pressing need to systematize and comprehend their intricacies [2]. Currently, Hugging Face (HF) is the most popular and widely used NLP library [11]. Rooted in the area of Transformer-based applications [21], HF has been recently extended to support the exploration of LLMs; however, its interface fails to provide connected and structured information about the interdependencies between models, datasets, and applications, as well as their performance evaluation with suitable metrics. Typical users of the HF platform (e.g.,

machine learning engineers and developers) know these limitations and carefully consider specific requirements and constraints when utilizing the library. For example, the choice of appropriate metrics to assess model performance over a downstream task requires inspecting dedicated leaderboards.

A promising approach to mitigate these limitations is to leverage conceptual models to represent and connect concepts within the LLM domain, thereby aiding in selecting the most suitable language model and datasets tailored to specific tasks, domains, regulatory requirements, and considerations regarding potential bias or hallucination. Towards this goal, we provide our vision for a new conceptual map[1] that overcomes the limitations of HF; moreover, we present exploratory queries that can be supported by a knowledge base designed on our conceptual map.

In Sec. 2, we present the conceptual map underlying the HF library, which is significantly extended in our proposal (Sec. 3), enabling several queries listed in Sec. 4 on top of our envisioned knowledge base (Sec. 5).

2 HF conceptual map and its limitations

HF is an online platform that hosts, as of March 2024, over 530,000 open-source models and 110,000 datasets [9]. In Fig. 1 we describe the main concepts and relationships on top of which HF exposes its main services. Notice that the underlying data model is proprietary. Hence, our reverse engineering process is exclusively based on the exposed resources and APIs documentation. Rather than highlighting HF model design issues, our goal is to highlight complementary or additional information about LLM models and applications that is worth considering and build an enriched conceptual data model on top of it.

A LargeLanguageModel presents a *Name* (e.g., "gpt2") and the following attributes: *URI* (meaning Unique Resource Identifier) may contain a recognized (bibliographic) reference to the model; *Language* indicates the one or more languages the model was trained on (e.g., "English"); *Library–Framework* indicates which software is adopted by the model (e.g., "JAX", "PyTorch", "TensorFlow"); *ModelCreator* is the company, research institute, university, or individual that has developed/trained the model[2] (e.g. "openai-community"); *LicenseToUse* formally specifies when the model can be used (e.g. "MIT"); *Architecture* is the foundation model underlying the described model (usually a multi-purpose, pre-trained model developed by well-funded institutions, e.g., "T5" by Google [16]); *Fine-tuned* is a Boolean flag indicating foundation models (=`False`) or models trained for solving a specific DownstreamTask, possibly on a specific *Domain* (=`True`). The *NumberOfParameters* characterizes the model, with larger ones usually performing better but requiring more computational resources for deployment. Scaling laws for the performance-parameters

[1] Throughout the text we do not employ the term "model" to avoid overloading the reader when referring to LLMs.

[2] We recall that HF is an open-source community portal, on which everyone can load their model.

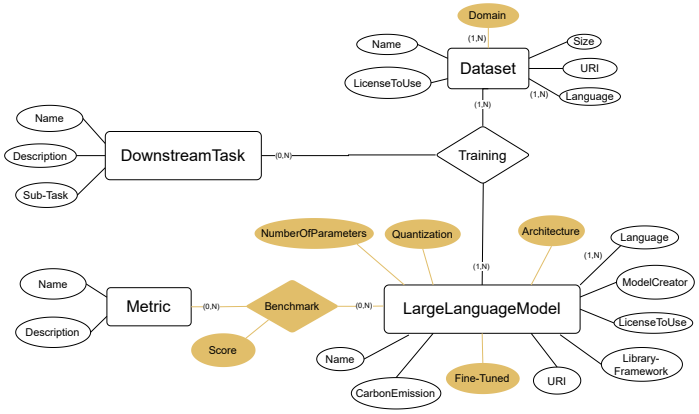

Fig. 1. Conceptual map of HF platform with four entities: LargeLanguageModel, Dataset, DownstreamTask, Metric. Attributes and relationships in white are derived from the website/API, whereas those in yellow can only be derived from the HF leaderboard [10] or by inspecting the descriptions of the entities on the HF portal.

tradeoff are currently an object of study [13]; recent experiments have shown that smaller models can outperform larger ones when trained on bigger text corpora [7]. *Quantization* indicates types of model compression to convert parameters to lower precision (e.g., "LLM.int8()" [6]) so that fewer resources are needed to deploy the model. Finally, *CarbonEmission* tracks the emissions [8] needed to train the model.

A Dataset has a *Name* (e.g., "Wikipedia") and a *Size* expressed in the number of tokens; datasets used for fine-tuning or testing a model are commonly smaller than those used in the pre-training phase. Datasets can be single or multi-language (cf. *Language* attribute); multi-lingual models have been trained on datasets that contain multiple languages. For this reason, models should be evaluated on datasets containing the languages seen by the model in the training phase. It is essential to check the dataset's *LicenseToUse*; publicly available datasets are not necessarily usable for any (commercial) purpose. Datasets can be of interest to one or more *Domain*s. When using models on specific DownstreamTasks, it could be helpful to fine-tune (or test) specific domains, such as the "legal" or "medical" ones. Recent research has focused on building domain-specific models [3]. Finally, a Dataset contains an external *URI* reference.

The DownstreamTask entity has a *Name* (e.g., "Summarization", "Question-Answering (Q-A)", "Translation"), a brief *Description*, and possibly a more specific *Sub-task* (e.g., "Multiple choice Q-A", "Open-domain Q-A", "Extractive Q-A").

Finally, the Metric entity, with *Name* (e.g., "Perplexity") and a *Description*, measures a model's performance in response to a given task.

Two relationships characterize the HF map. The ternary relationship TRAINING highlights that each LargeLanguageModel is trained (and pos-

sibly fine-tuned) on one or more DATASETs to solve one or more DOWNSTREAMTASKs. The binary relationship BENCHMARK represents the evaluation of LARGELANGUAGEMODELs through none, one, or more METRICs with a resulting *Score*.

HF's map presents several limitations. While some attributes are categorized on the main webpage for both the LARGELANGUAGEMODELs and the DATASETs and can be easily used as filters, some are included in a generic category denoted as 'Others'. That is the case of *Quantization* for LARGELANGUAGEMODELs and *Domain* for DATASETs. Other attributes, such as *Architecture* for LARGELANGUAGEMODELs – which is used as a tag for models fine-tuned on a common base (e.g., "T5") – can be retrieved via the API but do not appear on the main webpage. Conversely, other information is available on the website but not accessible via the API (for instance, the *NumberOfParameters* in MODELs, which is associated with an optional tag). The METRIC entity is particularly critical; all the related information is unstructured (available as a textual description). The issues in the map described so far limit the queries readily available to the user. Even though information on the TRAINING relationship can be retrieved from the web interface, several attributes of the involved entities are not immediately accessible. Instead, the BENCHMARK relationship can be reconstructed exclusively through the available leaderboards, which are user-defined and based on arbitrary METRICs.

3 Extended conceptual map

We propose an extended conceptual map for better representing the landscape of LLMs, shown in Fig. 2. It comprises the same four entities described in Sec. 2; however, it provides additional information to support relevant queries involving attributes, entities, and relationships that were not available in the HF map. We highlight in green all the entities/attributes/relationships that are currently not present in the conceptual map of Fig. 1, as not exposed in HF.

The LARGELANGUAGEMODEL is uniquely identified by the pair ⟨*Name*, *Version*⟩; this distinction is not formalized and sometimes missing/not exposed in HF (e.g., "Llama-2" in HF is split into "Llama", "2" in our map). To enrich the HF information, we add the *NumberOfParameters* of the model, the *ModelCreator*, i.e., the original author of a model (e.g., "Meta") and the *Developer*, i.e., whoever has fine-tuned the model to serve a specific need – starting from a foundation model. All the remaining HF attributes are preserved. In addition, we introduce the *ContextLength*, which characterizes the number of tokens that the model can handle (e.g., "4k"), and a *Tokenizer* that determines how the input prompt (and the output answer) are divided into tokens (e.g., "SentencePiece Byte-Pair Encoding"). The Boolean attribute *OpenSource* identifies whether or not a model is utterly transparent in terms of its architecture, training data, and methodologies; all models hosted on the HF portal are open-source (then, it is possible to have access to their weights), but this is not valid in general.

The DATASET entity is identified by its *Name*. We further define a total, exclusive, specialization of this entity in TRANINGDATASET and EVALUATION-

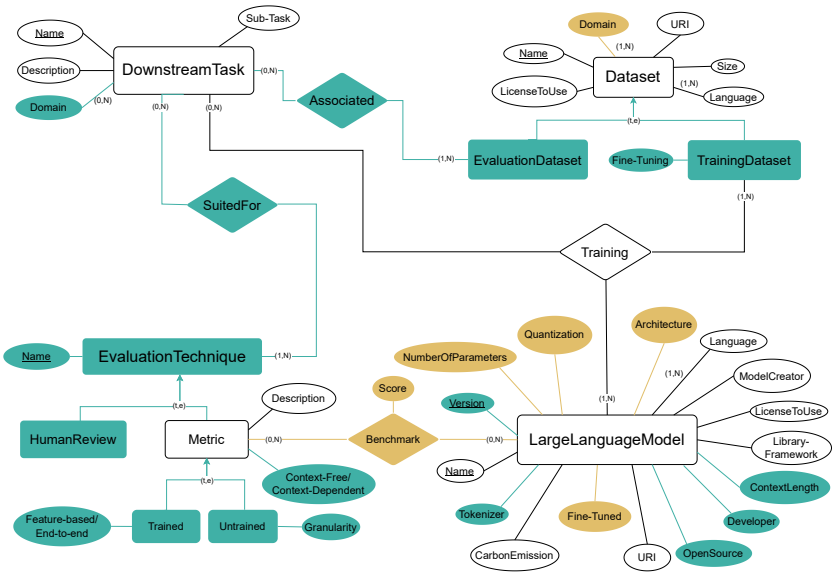

Fig. 2. Extended conceptual map. Green entities/attributes/relationships are new, as they are not present in the conceptual map of Fig. 1.

DATASET, assigned depending on the role assigned by the creator of the dataset. A TRAININGDATASET can be used to train or fine-tune the model on a specific downstream task (in the latter case, the *Fine-tuned* attribute is `True`). EVALUATIONDATASETs often provide a target associated with a sequence of tokens to allow comparison.

The DOWNSTREAMTASK entity presents the same attributes as in the HF map, except for introducing the possibility of assigning one or more *Domain*s to which the task refers.

The EVALUATIONTECHNIQUE entity represents different forms of evaluation of the performance of a model and is identified by its *Name*. It relies on a computed METRIC or a HUMANREVIEW of the model. A metric is characterized by a *Description* and a *Context-Free/Context-Dependent* flag. Context-free metrics only need the model's output and a ground truth text to use as a reference. Context-dependent metrics, instead, need the model's input, such as a table, a document, etc. For this reason, most context-free metrics are task-agnostic and can be adapted to many scenarios, while most context-dependent metrics are task-specific.

Metrics are TRAINED or UNTRAINED. TRAINED metrics have trainable parameters and need human annotations to be trained on. An example of trained metrics can be the recognition of hallucinations of LLMs as occurs in [1], where some prompts and responses of LLMs are manually annotated by humans as hallucinated or not to train NLP models for their automatic recognition. They can use other metrics/heuristics as input features or can be taught in an end-to-end

manner, requiring the input given to the model, the output of the model, and the ground truth; we encode this information in the *Feature-based/End-to-end* attribute. UNTRAINED metrics can operate at the Word-Character level or use Embeddings to evaluate the output of the models. Embeddings-based metrics are usually able to capture the semantics of the phrase. We represent this information in the *Granularity* attribute. Note that both trained and untrained metrics can be context-free or context-dependent, but context-dependent metrics are usually trained. The described taxonomy represents an important addition to the HF map, allowing queries to be performed on previously unstructured information.

The extended map includes the TRAINING and the BENCHMARK relationships shown in Fig. 1 and adds two new relationships. The SUITED_FOR relationship can connect none, one or more DOWNSTREAMTASKs with one or more EVALUATIONTECHNIQUEs. An EVALUATIONTECHNIQUE must be suited for at least one DOWNSTREAMTASK (e.g., ROUGE for the summarization task [14]). Similarly, the ASSOCIATED relationship connects EVALUATIONDATASET to DOWNSTREAMTASK (e.g., the SQuAD dataset for Question-Answering [17]).

4 Exploratory queries

In the following, we propose a few queries that can be performed on our map while are not supported on HF.

Query 1. *"Find all the LLaMA-based large language models fine-tuned for the chat task."* This query exploits the TRAINING relationship between LARGELANGUAGEMODEL and DOWNSTREAMTASK and filters the "LLaMA" *Architecture* and a "`True`" *Fine-tuned* attribute of LARGELANGUAGEMODEL, as well as the "Chat" *Name* of the DOWNSTREAMTASK. **Example output.** Vicuna [4], an open-source ChatBot that achieves promising performances on the chat task [23]. **User advantage.** Currently, filtering models on specific tasks and architectures is not supported in HF.

Query 2. *"Find the models with less than 8 billion parameters, fine-tuned on question answering (Q-A) in the medical domain."* The query considers the TRAINING relationship between the LARGELANGUAGEMODEL and DOWNSTREAMTASK entities. We filter on the attributes *NumberOfParameters* ($\leq 8B$) and *Fine-tuned* (`True`) for LARGELANGUAGEMODEL, and the *Name* and *Domain* of the DOWNSTREAMTASK ("Question Answering" and "medical"). **Example output.** MEDITRON-7B [3]. **User advantage.** Hardware limitations must be considered when looking for models that solve a specific task. Here, we can determine which model is both suitable for the Q-A task in the medical domain and not too big so that it can be easily deployed. Currently, this possibility is absent on HF, which does not include the *NumberOfParameters* attribute; moreover, it is not possible to look for *Fine-tuned* models on a specific *Domain*.

Query 3. *"Find all the models fine-tuned on the legal domain for text summarization, with a context length greater than* $32k$ *tokens."* The query considers the TRAINING relationship between the LARGELANGUAGEMODEL and DOWNSTREAMTASK entities, filtering on the *ContextLength* attribute ($\geq 32k$) and the

Name/*Domain* of the DownstreamTask ("text summarization" and "legal"). **Example output.** SaulLM7B [5], based on Mistral 7B [12]. **User advantage.** Summarizing long documents is an essential task, which requires models to have a reasonable context length (in our case, $32k$). This filter is not supported in HF, which lacks the *ContextLength* attribute in LargeLanguageModel and *Domain* in DownstreamTask.

Query 4. *"Find a suitable untrained metric with character-based granularity suitable for machine translation."* This query considers the SUITED_FOR relationship between the EvaluationTechnique and DownstreamTask entities. Considered attributes are the *Granularity* of Untrained metric ("character"), and the *Name* of DownstreamTask ("Machine Translation"). **Example output.** The chrF metric [15]. **User advantage.** Enabling filtering on evaluation techniques based on specific downstream tasks, a link currently lacking in HF.

Query 5. *"Find open-source Large Language Models that are specialized in Code Generation and were trained for the Python language on at least 50 billion tokens."* The query takes into account the TRAINING relationship between the LargeLanguageModel, Dataset, and DownstreamTask entities. The attributes considered are *OpenSource* ("`True`") for LargeLanguageModel; *Size* and *Language* of the Dataset (requiring that the sum of datasets' sizes for the "Python" language is $\geq 50B$); and the *Name* and *Sub-task* of the DownstreamTask ("Text Generation"/"Code Generation"). **Example output.** Code Llama Python [18], a model trained on publicly available code, discussions about code and code snippets. **User advantage.** The search for LLMs specialized in a particular programming language, forcing the minimum training data, is important. This complex query is not possible in HF.

5 Conclusions and Vision

We identified specific high-quality information sources and will exploit them to instantiate a knowledge base of LLMs enriching the information exposed by the Hugging Face library. Primarily, we will employ HF APIs to retrieve immediately-available information (e.g., Tasks/sub-tasks, Models, Datasets, and Licenses). This will be compared and integrated with other up-to-date sources, such as scientific publications (tasks and related training datasets from Sanh et al. [20], Large Language Models from Zhao et al. [22], evaluation techniques from Sai et al. [19]).

Next, to scale up the instantiation of the knowledge base, we also aim to experiment with NLP techniques, supporting automatic scraping of information about LLMs from online resources. In the future, we aim to have the content fed using crowdsourcing approaches.

Overall, this vision paper proposes a conceptual map of LLM-related information. We will provide API services to query the content of each entity (i.e., retrieving a list of objects with their attributes) and run queries connecting instances. Once fully functioning, this knowledge map can be extended to an interactive web app that supports all researchers in exploring and explaining this

new arising domain, guiding the design and engineering of LLM usage, comparisons, and evaluations.

Acknowledgements. The authors are thankful to other members of the ChatIMPACT project of Alta Scuola Politecnica: Andrea Clerici, Flavio Giobergia, Pietro Pinoli, and Piercesare Secchi.

References

1. Borra, F., et al.: MALTO at SemEval-2024 Task 6: Leveraging Synthetic Data for LLM Hallucination Detection. arXiv:2403.00964 (2024)
2. Chang, Y., et al.: A Survey on Evaluation of Large Language Models. ACM Transactions on Intelligent Systems and Technology (2024)
3. Chen, Z., et al.: MEDITRON-70B: Scaling Medical Pretraining for Large Language Models. arXiv:2311.16079 (2023)
4. Chiang, W.L., et al.: Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality. https://lmsys.org/blog/2023-03-30-vicuna/
5. Colombo, P., et al.: SaulLM-7B: A pioneering Large Language Model for Law. arXiv:2403.03883 (2024)
6. Dettmers, T., et al.: LLM.int8(): 8-bit Matrix Multiplication for Transformers at Scale. arXiv:2208.07339 (2022)
7. Hoffmann, J., et al.: An empirical analysis of compute-optimal large language model training. In: Advances in Neural Information Processing Systems (2022)
8. Hugging Face: Displaying carbon emissions for your model. https://huggingface.co/docs/hub/model-cards-co2
9. Hugging Face: Hub documentation. https://huggingface.co/docs/hub/index
10. Hugging Face: Open LLM Leaderboard. https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard
11. Hugging Face: The Hugging Face portal. https://huggingface.co/
12. Jiang, A.Q., et al.: Mistral 7B. arXiv:2310.06825 (2023)
13. Kaplan, J., et al.: Scaling laws for neural language models. arXiv:2001.08361 (2020)
14. Lin, C.Y.: ROUGE: A Package for Automatic Evaluation of Summaries. In: Text Summarization Branches Out. pp. 74–81. ACL, Barcelona, Spain (2004)
15. Popović, M.: chrF: character n-gram F-score for automatic MT evaluation. In: Proc. of the 10th workshop on statistical machine translation. pp. 392–395 (2015)
16. Raffel, C., et al.: Exploring the limits of transfer learning with a unified text-to-text transformer. The Journal of Machine Learning Research **21**(1), 5485–5551 (2020)
17. Rajpurkar, P., et al.: SQuAD: 100,000+ Questions for Machine Comprehension of Text. In: Su, J., et al. (eds.) Proc. of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 2383–2392. ACL (2016)
18. Roziere, B., et al.: Code Llama: Open Foundation Models for Code. arXiv:2308.12950 (2023)
19. Sai, A.B., et al.: A survey of evaluation metrics used for NLG systems. ACM Computing Surveys (CSUR) **55**(2), 1–39 (2022)
20. Sanh, V., et al.: Multitask Prompted Training Enables Zero-Shot Task Generalization. In: International Conference on Learning Representations (2022)
21. Vaswani, A., et al.: Attention is All you Need. In: Guyon, I., et al. (eds.) Advances in Neural Information Processing Systems. vol. 30 (2017)
22. Zhao, W.X., et al.: A survey of large language models. arXiv:2303.18223 (2023)
23. Zheng, L., et al.: Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In: 37th Conference on Neural Information Processing Systems Datasets and Benchmarks Track (2023)