Granite Foundation Models
IBM Research

Abstract—We introduce the Granite series of decoder-only foundation models for generative artificial intelligence (AI) tasks that are ready for enterprise use. We report on the architecture, capabilities, underlying data and data governance, training algorithms, compute infrastructure, energy and carbon footprint, testing and evaluation, socio-technical harms and mitigations, and usage policies.

Index Terms—foundation model, large language model, generative AI, data governance, contrastive fine-tuning, energy consumption, evaluation, socio-technical harms, usage governance, transparent documentation

I. INTRODUCTION

In this technical report, we present the Granite series of decoder-only foundation models for generative artificial intelligence (AI) tasks. The first in this series, granite.13b, is an English-only large language model (LLM). Using self-supervised learning, this base model has been trained on an IBM-curated pre-training dataset described in Section II. IBM relies on its internal end-to-end data and AI model lifecycle governance process and capabilities to develop enterprise-grade foundation models and is making similar capabilities available to customers of its watsonx platform.

The base model is the jumping-off point for two variants: granite.13b.instruct and granite.13b.chat. The first variant, granite.13b.instruct, has undergone supervised fine-tuning to enable better instruction following [1] so that the model can be used to complete enterprise tasks via prompt engineering. The second variant, granite.13b.chat, has undergone a novel contrastive fine-tuning after supervised fine-tuning to further improve the model’s instruction following, mitigate certain notions of harms, and encourage its outputs to follow certain social norms and have some notion of helpfulness [2]–[4]. We emphasize that these notions are not universal and discuss this point to a greater extent in Section VI on socio-technical harms and risks.

The granite.13b.instruct and granite.13b.chat models are made available by IBM through the watsonx platform [5]. IBM indemnifies customer use of these models on the watsonx platform, providing the same contractual intellectual property protections for IBM-developed AI models as it does for all of IBM’s products according to IBM Standard Terms and Conditions.

A. Overview of Capabilities

The 13b in the name indicates the model has 13 billion parameters. Furthermore, the base granite.13b decoder-only model has multi-query attention with learned position embeddings, has been trained on 1 trillion tokens created with the GPT-NeoX 20B tokenizer [6], and has a context length of 8 thousand tokens. As discussed in Section V, the Granite models are competitive in their ‘weight class’ on benchmark evaluations while being enterprise-ready in governance dimensions.

Some of the key enterprise tasks (common across sectors) for which the Granite models may be used are: retrieval-augmented generation, summarization, content generation, named entity recognition, insight extraction, and classification. The Granite models may be adapted to the specific tasks arising in particular enterprise applications through prompt engineering in the watsonx platform, which is illustrated in Fig. 1. Other series of models that IBM is developing are Sandstone: encoder-decoder models designed to be tuned for specific tasks and Obsidian: modular universal transformer models suitable for high inference efficiency.

B. Overview of the Granite Pre-Training Dataset

To support the training of large enterprise-grade foundation models, including granite.13b, IBM curated a massive dataset of relevant unstructured language data from sources across academia, the internet, enterprise (e.g., financial, legal), and code. In a rare move from a major provider of proprietary LLMs, IBM demonstrates its commitment to transparency and responsible AI by publishing descriptions of its training dataset in Section II.

The Granite pre-training dataset was created as a proprietary alternative to commonly used open-source data compilations for LLM training such as “The Pile” [7] or “C4” [8]. Some domains that are key for enterprise natural language processing are relatively under-represented in these compilations. Additionally these data compilations have been criticized for
containing toxic, harmful, or pirated content [9]. By curating our own pre-training data corpus, IBM takes significant steps towards addressing these and other issues.

The IBM curated pre-training dataset is continually growing and evolving, with additional data reviewed and considered to be added to the corpus at regular intervals. In addition to increasing the size and scope of pre-training data, new versions of these datasets are regularly generated and maintained to reflect enhanced filtering capabilities (e.g., de-duplication and hate and profanity detection) and improved tooling.

C. Organization of Report

The remainder of this report is organized as follows. In Section II, we describe the data sources used in granite.13b’s pre-training. In Section III, we describe the data processing steps we undertake with a focus on the governance steps we follow. In Section IV, we provide further details about the pre-training and fine-tuning algorithms, the computation involved, and the energy consumption we estimate. Section V presents the testing and evaluation framework along with quantitative comparisons to other models. In Section VI, we discuss our approach to understanding and mitigating socio-technical harms from the Granite models. Section VII provides a brief discussion of the usage policies and the socio-technical documentation of Granite models. Finally in Section VIII, we conclude with areas of future work and discussion.

II. DATA SOURCES

At the time of granite.13b’s pre-training, IBM had curated 6.48 TB of data before pre-processing, 2.07 TB after pre-processing (detailed in Section III). All datasets were filtered English-text and code unstructured data files. There are no pre-defined labels or targets. All non-text artifacts (e.g., images, HTML tags, etc.) were removed.

Specifically, for the purposes of training granite.13b, 1 trillion tokens were generated from a total of 14 datasets. The individual datasets used in the training are described below.

1) arXiv: Over 1.8 million scientific paper pre-prints posted to arXiv.
2) Common Crawl: Open repository of web crawl data.
3) DeepMind Mathematics: Mathematical question and answer pairs data.
5) GitHub Clean: Code data from CodeParrot covering a variety of coding languages.
8) Project Gutenberg (PG-19): A repository of free e-books with focus on older works for which U.S. copyright has expired.
9) Pubmed Central: Biomedical and life sciences papers.
11) Stack Exchange: Anonymized set of all user-contributed content on the Stack Exchange network, a popular collection of websites centered around user-contributed questions and answers.
12) USPTO: US patents granted from 1975 to May 2023, excluding design patents.
13) Webhose: Unstructured web content converted into machine-readable data feeds acquired by IBM.
14) Wikimedia: Eight English Wikimedia projects (enwiki, enwikibooks, enwikinews, enwikiquotes, enwikisource, enwikiversity, enwikivoyage, enwiktionary). containing extracted plain text from pages and articles.

III. DATA GOVERNANCE

As IBM is making Granite models available to customers to adapt to their own applications, we have invested heavily in a data governance process that evaluates datasets for governance, risk and compliance (GRC) criteria, including IBM’s standard data clearance process, document quality checks, and other criteria. IBM has developed governance procedures for LLM pre-training datasets which are consistent with IBM AI Ethics principles and are guided by the IBM Corporate Legal Team. Best practices around LLM development is continually evolving with the ever-increasing understanding of AI models, their usage, and changing regulatory requirements, among other factors.

Addressing GRC criteria for data spans the lifecycle of training data, from data request to tokenization. An important objective for IBM is establishing an internal auditable link from a trained foundation model to the specific dataset version on which the model was trained, including information about each processing step performed prior to training. Summary statistics on IBM’s curated pre-training dataset are provided in Fig. 2.

Data governance is organized into the following processes, corresponding to lifecycle phases prior to model training:

A. Data clearance and acquisition;
B. Pre-processing; and
C. Tokenization.

Each process is composed of sub-processes focusing on specific governance aspects. The remainder of this section describes each phase in detail.

A. Data Clearance and Acquisition

The data clearance process assures that no datasets are used to train IBM foundation models, including the Granite series, without careful consideration. Before data is added to IBM’s curated pre-training dataset, it is submitted to the data clearance process and subject to technical, business,
Fig. 2. Summary governance statistics on IBM’s curated pre-training dataset at the time of granite.13b’s training.

and governance review. The clearance request captures comprehensive information about a dataset such as a thorough description, the data owner, the intended use, geographic location, data classification, licensing information (if available), usage restrictions and sensitivity (e.g., personal information). Additional information includes who will have access to the data, and how the data will be acquired.

Once a dataset completes the review process, it is tagged for potential inclusion, its metadata is moved into a catalog of approved datasets, and it is downloaded and prepared for the subsequent pre-processing stages.

Remark. IBM’s pre-training dataset currently addresses copyrighted material through selective use of URL block-listing for websites known to disseminate pirated information. Examples of datasets that are block-listed include the Books3 dataset, which is specifically excluded from use due to concerns on the copyright status of the data and its use in model training.

B. Pre-Processing Pipeline

Once data has been cleared and downloaded, it is prepared for model training through a variety of steps collectively referred to as the pre-processing pipeline. An overview of the pre-processing pipeline for this release of Granite models is depicted in Fig. 3 and is composed of the following steps:

1) Text Extraction:
Text extraction is the first step in the pipeline, and is used to extract language from various documents into a standardized format for further processing.

2) Data De-Duplication:
Data de-duplication aims to identify and remove duplicate documents. De-duplication is performed on a per-dataset basis and is essential to ensuring the trained model does not learn artificial linguistic patterns due to repeated data in the dataset.

Two techniques are used: exact and fuzzy de-duplication, both of which use hash-based methods. As the name suggests, exact de-duplication removes exact duplicates among the documents in the dataset. Each document is hashed and documents with the same hash are fused to one. For example, if 50 documents in a dataset have the same hash, a single document will be used. Fuzzy de-duplication finds the Jaccard similarity between documents with locality sensitive hashing. If multiple updated snapshots of a dataset are downloaded, the exact de-duplication is performed across all snapshots.

3) Language Identification:
Language identification is performed at a document level to detect the dominant language using the Watson Natural Language Processing (NLP) library [10].

The output of this task is an additional column in the parquet file containing a two letter ISO language code.

In the case of the Common Crawl dataset, language is already provided through folder names. The Watson NLP language identification algorithm is nevertheless run on Common Crawl documents, yielding two language classifications for these documents: Common Crawl and Watson NLP.

4) Sentence Splitting:
Sentence splitting involves decomposing each document into its constituent sentences. Sentence splitting is key for hate, abuse, and profanity (HAP) annotation (to be discussed below) since HAP annotation is performed at a sentence level. As such, the sentence splitting stage must take place prior to the start of HAP annotation. Sentence splitting for the English language is performed using Watson NLP.
5) Hate, Abuse and Profanity Annotation: Data sources drawing from the open Internet, such as Common Crawl, inevitably contain abusive language. To reduce the possibility of Granite models producing profane content, each sentence in each document is assessed and scored as to its level of HAP content. The HAP detector is itself a language model trained by IBM and benchmarked against internal as well as public models such as OffensEval [11], AbusEval [12] and HatEval [13]. The IBM HAP detector performs comparably to HateBERT [14].

After a score is assigned to each sentence in the document, analytics are run over the sentences and scores to explore the distribution of annotations in each document with a HAP annotation. This serves both to determine the percentage of HAP sentences in a document as well as to determine threshold values used later during filtering.

6) Document Quality: Quality annotation aims to identify documents with low linguistic value using both heuristics and a classifier. The heuristics are derived from the Gopher Quality Filtering criteria [15]:

- total words: outside the range 50–100,000 words;
- average word length: outside the range 3–10 characters per word;
- symbol to word ratio: greater than 10%;
- bullet points ratio: greater than 90%;
- ellipse line ratio: greater than 30%;
- alphabet words ratio: fewer than 80%;
- common English words: does not contain at least 2 from {the, be, to, of, and, that, have, with}.

The classifier assigns a perplexity score using the KenLM linear classifier pre-trained on Wikipedia documents [16], [17]. For any document, the model provides a score of the document’s similarity to a training corpus (i.e., Wikipedia). These heuristics and classifiers output columns with quality scores that are added to the parquet file. These annotations form the basis for quality filtering during the filtering step.

7) URL Block-Listing: Block-listing identifies documents to be blocked from being added to IBM’s curated pre-training dataset. The block list is continuously maintained and includes URLs of known copyrighted material as well as block-listed sites such those contained in the 2022 Review of Notorious Markets for Counterfeiting and Piracy [18].

8) Filtering: Filtering occurs at the document level and is the last step before tokenization. It is here that annotations created in previous pre-processing steps are used to prevent documents from being used for tokenization. For example, documents are dropped which exceed HAP thresholds or do not meet a defined document quality. For the current English-only Granite models, the language identification annotations are used to filter out non-English documents.

C. Tokenization

Tokenization is the final pre-processing step prior to model training. For granite.13b, the cleaned and filtered text is converted from a sequence of characters to a vector of tokens using the GPT-NeoX 20B tokenizer [6].

IV. Training

In this section, we detail the training process for the decoder-only Granite models covering the algorithmic details of pre-training and fine-tuning, the computing involved, and an estimate of the carbon footprint.

A. Algorithmic Details

1) Pre-Training: We adopt most of the pre-training settings from [19]. Specifically, we use the standard decoder-only transformer architecture [20], Gaussian error linear unit (GELU) activation function [21], MultiQuery-Attention for inference efficiency [22], and learned absolute positional embeddings. We also adopt FlashAttention to speed up the training and reduce its memory footprint [23], allowing us to increase the context length to 8192 from the context length 2048 used by many existing LLMs.

The granite.13b base model is trained for 300K iterations, with a batch size of 4M tokens, for a total of 1 trillion tokens.

We train using the Adam optimizer [24], with $\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 10^{-8}$, and a weight decay of 0.1. We use a cosine learning rate schedule, with warmup of 2000 steps, and decay final learning rate down from $3 \times 10^{-4}$ to $3 \times 10^{-5}$. We pretrain models with a 3D-parallel layout using both tensor and pipeline parallelism including sequence parallelism to enable training with 8K context length.

2) Supervised Fine-Tuning: Pre-training teaches the LLM to continue generating text based on the input. However in practice, users often expect the LLM to treat the input as instructions to follow. To enable instruction following, we perform supervised fine-tuning (SFT) with a mixture of datasets from different sources. Each sample consists of a prompt and an answer. We use a cosine learning rate schedule with an initial learning rate of $2 \times 10^{-5}$, a weight decay of 0.1, a batch size of 128, and a sequence length of 8192 tokens. We perform SFT for 3 epochs to obtain the granite.13b.instruct model.

The SFT data includes a subset of the Flan Collection [25], 15K samples from Dolly [2], Anthropic’s human preference data about helpfulness and harmlessness [3], Instructv3 [26], and internal synthetic datasets specifically designed for summarization and dialogue tasks.

3) Contrastive Fine-Tuning: Contrastive fine-tuning (CFT) is an instruction fine-tuning approach based on unlikelihood-based training [27], which penalizes the probability of data points from a negative data distribution while simultaneously increasing the probability of data points from a positive data distribution (see Fig. 4). In other words, we discourage an LLM from generating misaligned responses (e.g. responses that are harmful) while encouraging aligned responses (e.g. responses that are helpful) for each training prompt.
Fig. 4. Illustration for the resulting LLM distribution over responses $y$ for a given prompt $x$. (a) SFT increases the likelihood of aligned responses but lacks control over misaligned responses. As a result, misaligned responses that closely resemble aligned ones can still have high likelihood. Examples A and B demonstrate this, where A is aligned and B is misaligned, yet they are similar. (b) Our approach, contrastive fine-tuning mitigates this by explicitly assigning low likelihood to misaligned responses.

CFT requires both responses to be paired with the same prompt in order for the model to determine which response is worse. However, many publicly available human demonstration datasets lack paired aligned and misaligned responses for the same prompt. As such, one may wonder: “How can one obtain both aligned and misaligned responses?” A straightforward approach to obtain this negative data distribution is to have humans write misaligned responses for each prompt. However, such an approach can be cost-prohibitive.

Thus in our work we propose to use a separate LLM to serve as a ‘negative persona’ by mimicking individuals who tend to respond in a misaligned (e.g. harmful or untruthful) manner. We achieve this by fine-tuning an LLM on widely available misaligned human demonstration datasets. Consequently, for a given dataset of prompts and aligned human demonstrations, responses on the prompts from this negative persona LLM form the misaligned responses and the paired aligned responses are the demonstrations themselves.

For granite.13b.chat, we use an early version of granite.13b.instruct as the separate LLM. The datasets for CFT are paired samples from Anthropic’s human preference data about helpfulness and harmlessness that have been filtered using the OpenAssist reward model [3], samples from Dolly [2], and samples from ProsocialDialog [4].

As a part of the CFT step, the granite.13b.chat was also trained to work with the following system prompt [3] in order to support Human-Agent based dialogue:

Below are a series of dialogues between various people and an AI assistant. The AI tries to be helpful, polite, honest, sophisticated, emotionally aware, and humble-but-knowledgeable. The assistant is happy to help with almost anything, and will do its best to understand exactly what is needed. It also tries to avoid giving false or misleading information, and it caveats when it isn’t entirely sure about the right answer. Moreover, the assistant prioritizes caution over usefulness, refusing to answer questions that it considers unsafe, immoral, unethical or dangerous. Human:<prompt> Assistant:
and $CEF(L)$ is the carbon emission factor applicable for the location $L$.

The information technology (IT) electricity consumption $E(M)$ is estimated using the average GPU utilization rate for all the GPUs. It is a proxy to estimate the power that is used to train the AI model $M$ since the GPU utilization is typically highly correlated with the node power, as shown in Fig. 6. Then, the estimated node power is multiplied by the training time and the number of GPUs used to calculate the total compute energy consumption $E$.

Power usage effectiveness $PUE(L)$ is given by the ratio of the total electricity consumed by the data center (aggregate consumption by the IT and support overhead infrastructure) to that consumed by the IT infrastructure. We calculate the location-based carbon emission factor $CEF(L)$ following the GHG Protocol’s Scope 2 Guidance [29].

Applying this estimation methodology to the granite.13b base model, we estimated 153074.3767 kWh energy consumption $E(M)$ and 0.12 kg/kWh carbon emission factor $CEF(L)$, yielding 22.2263995 tons of CO$_2$ equivalent $Carbon(M,L)$, which accounts for carbon dioxide and all other greenhouse gases, such as methane and nitrous oxide.

A number of mitigation strategies may be used to reduce the energy and carbon footprint. For example, the amount of resources used in training may be adjusted as a function of the availability of renewable energy, or the resources usage may be capped to not exceed certain energy usage or emissions limits.

V. TESTING AND EVALUATION

In this section, we describe the approach taken to test and evaluate the Granite models. We also provide empirical results along with comparisons to several other models that are of a similar capability level.

A. Foundation Model Evaluation Framework

We use a comprehensive foundation model evaluation framework (FM-eval) through the model’s development lifecycle. FM-eval is running on RedHat OpenShift\(^1\) cluster with GPU support, for efficient execution of evaluation benchmarks, in parallel and on multiple models. The automation framework can run any containerized evaluation framework or a wrapped external framework such as Eleuther AI’s Language Model Evaluation Harness (lm-eval) [30] or Stanford’s HELM (Holistic Evaluation Model) [31]. To allow easy addition of tasks, datasets and metrics to FM-eval, we developed Unitxt\(^2\), an open-source Python library that provides a consistent interface and methodology for defining datasets, including the preprocessing required to convert raw datasets to the input required by LLMs, and the metrics used to evaluate the results.

Different types of tests are run during different phases of the lifecycle:

1) General knowledge benchmarks (during training)
2) HELM benchmarks (post-training)
3) Enterprise benchmarks (post-training)

These evaluations all leverage zero-shot and few-shot prompting. For clarity, zero-shot prompting uses a pre-existing LLM to generate text for a new task by only providing the instruction to execute the task in the prompt. In few-shot prompting, we provide multiple in-context examples, along with the task at hand, directly within the prompt. Both approaches allowed us to work with a single pre-trained model whose core parameters remained fixed.

The specific evaluations are detailed below.

1) General Knowledge Benchmarks During Training: The General Knowledge Benchmarks include a subset of existing benchmarks from lm-eval [30] and are used as light-weight tests run after every 100 billion tokens during training to validate model knowledge is advancing as training progresses.

Specifically, the following 12 datasets (organized by task) from lm-eval are:

- question answering for several domains (boolq, openbookqa, piqa, sciq);
- sentence completion (lambada)
- commonsense reasoning (arc_easy, arc_challenge, copa, hellaswag, winogrande);
- reading comprehension (race)
- multidisciplinary multiple-choice collection (mmlu);

In our evaluation framework these benchmarks are run in both the zero-shot and few-shot setting.

2) HELM: Part of the comprehensive assessment after pre-training relies on Stanford’s Hollistic Evaluation of Language Models (HELM) Benchmark [31]. To evaluate our model, we use the 16 “core scenarios”, consisting of a variety of tasks including question answering, information retrieval, summarization, sentiment analysis, and text classification [32]–[43], on which all HELM LLMs are evaluated and compared using their results and mean win rate (MWR).

3) Enterprise Evaluation Benchmarks: After training completes, we further evaluate our models on IBM-curated enterprise benchmarks to test our models performance in domains

\(^1\)https://www.redhat.com/en/technologies/cloud-computing/openshift

\(^2\)https://github.com/IBM/unitxt
highly relevant to our customers. With this in mind, IBM curated 11 publicly available finance benchmarks for evaluating models in the financial domain, summarized in Table I.

The data source-provided train and test splits are used in the evaluation whenever possible. Model performance is reported based on test examples. If the test labels are not publicly available, model performance is reported on the validation set. If the train and test splits do not exist in the data source, 20% of the data is selected as test split and the rest is used as the train split.

All few-shot context examples are sampled from the training set. The number of few-shot examples provided to the model depends on the task, which is provided in Table I. For the current evaluation, all the models used the same parameters and the same standard prompt (see the techniques of few-shot-prompting and zero-shot-prompting and examples of prompts\(^3\), without task description, chain-of-thought prompting [44], or system prompts in place. For Financial Phrasesbank, News Headline and FiQA SA, the prompts were taken from BloombergGPT [45].

### B. Granite Model Evaluation and Comparison

It should be noted that while future versions of granite.13b will be trained on upwards of 2T tokens, this initial release of granite.13b only saw 1T tokens during training, making all of these evaluations preliminary.

1) **General Knowledge Benchmarks During Training**: In this section, we leverage the lighter-weight General Knowledge Benchmarks to assess a series of snapshots of the granite.13b base model taken every 100B tokens during training along with the fine-tuned granite.13b.instruct and granite.13b.chat variants. As visualized in Fig. 7 and further detailed in Table II, progressively training on each 100B tokens steadily improved General Knowledge as expected with further boosts in performance achieved in both fine-tuned variants of granite.13b. Note system prompts were not used for this evaluation of granite.13b.chat.

2) **HELM Benchmarks**: In this study, we comprehensively assess our models across HELM’s 16 core scenarios, comparing them to all other models in the v0.2.3 release\(^5\).

Our evaluation follows the default two-tiered process as suggested by HELM: first, we evaluate each model on individual evaluation datasets, then we aggregate these results into scenarios (as detailed in Table III). To facilitate a fair comparison of LLMs, we employ HELM’s Mean Win Rate (MWR) metric across scenarios for model ranking.

Figure 8 illustrates the positioning of granite.13b.instruct, granite.13b.chat, and all v0.2.3 models along the model size MWR axes. Note, models where exact model size is not published by the LLM-provider are excluded from this visualization.

The figure highlights that the granite models strike a desirable balance between model size and HELM performance. Granite.13b.chat and granite.13b.instruct are respectively ranked 15 and 18, out of all models evaluated. Further, the granite.13b.chat and granite.13b.instruct models were respectively the top-2 and top-3 models evaluated under 17b parameters in size. Only Cohere Command beta (6.1b) exceeded their performance in this size category. These results also hold for the other aspects evaluated by HELM, such as robustness, and fairness. In calibration granite.13b.chat and granite.13b.instruct are ranked 9 and 28, respectively.

3) **Enterprise Benchmarks**: This evaluation is conducted by augmenting HELM’s framework to encompass 11 publicly available task datasets from the financial services domain. Baseline models are selected based on model size, type of training data, accessibility, and model tuning. To be specific, granite models are compared with GPT-NeoX-20B [6] and FLAN-UL2 [56], some of the best performing open-sourced models under 50 billion parameters, along with state-of-the-art available LLaMA2 models [57] between 7 billion to 13 billion parameters.

Table IV presents the detailed performance scores of the models on the 11 financial tasks. An asterisk is given next to the LLaMA2 models, as these models have seen 2T tokens of pre-training data, imparting significant advantage to the

---

\(^3\)https://www.promptingguide.ai/techniques/fewshot

\(^5\)https://crfm.stanford.edu/helm/latest/?group=core_scenarios
models. All other models evaluated, including granite, have seen 1T tokens of training data. Despite having been trained on half the amount of data as the LlMA2 models, the Granite models are competitive across each of the tasks, often meeting or outperforming LlMA2. This bodes well for future planned versions of granite models, which will be trained on 2T+ tokens of pre-training data.

### VI. Socio-Technical Harms and Risks

Numerous potential socio-technical harms and risks of LLMs have been identified in recent years, including misinformation, hallucination, lack of faithfulness or factuality, leakage of private information, plagiarism or inclusion of copyrighted content, hate speech, toxicity, human-computer interaction harms such as bullying and gaslighting, malicious uses, and adversarial attacks [58], [59].

In Table V, we present the catalogue of risks compiled by the IBM AI Ethics Board, a central, cross-disciplinary body that defines the AI ethics vision and strategy with the objective of supporting a culture of ethical, responsible, and trustworthy AI throughout the IBM Corporation [60], [61]. The table is organized across several dimensions [62]:

- Whether the risk is from the data or other inputs to the foundation model, from the generated output of the foundation model, or from other concerns.
- Whether the risk arises in the training/tuning of the model, during inference, or in broader considerations such as governance, legal compliance, or societal impact.
As part of creating and releasing the granite.13b.instruct and granite.13b.chat models, we have addressed some of the risks as follows. The data governance processes of the IBM’s pre-training dataset, including the block-listing and filtering of hate, abuse and profanity have mitigated some intellectual property and misuse risks. Toward fairness, an additional component of the data pre-processing pipeline not described in Section III is annotating documents by religion, gender, race, stigma, age, and political ideology. We have created keyword lists for these dimensions and use keyword matching to annotate sentences. The annotations may be used to identify under-represented and over-represented groups. We have not been overly aggressive in HAP filtering and have not filtered with respect to groups because it would prevent us from having training data that reclaims slurs and positively describes marginalized identities, and might skew the pre-training dataset in other unintended ways [63].

Through fine-tuning, we have encouraged prosocial and less harmful model behavior with the aim to mitigate certain aspects of misuse and value alignment risks. However, one of the biggest socio-technical risks would be our own hubris to believe that the datasets we used for fine-tuning (or other existing datasets we could use in the future) are aligned to the needs, wants, and desires of the peoples and organizations that will be deploying the Granite models to meet their own goals. Every enterprise has its own regulations to conform to, whether they come from laws, social norms, industry standards, market demands, or architectural requirements [64]; we believe that enterprises should be empowered to personalize their models according to their own values (within bounds) [65], e.g. using tools in the watsonx platform.

In addition, through FM-eval, we have tested the Granite models on benchmark datasets that cover several risk dimensions. However, evaluating on benchmarks is a limited approach for revealing socio-technical harms [66]. After enterprises have further aligned the Granite models to their own values, they should enlist a red team with members of varying socio-cultural and lived experience to find additional harms and undesirable LLM behaviors within the context of a precise use case [67].

VII. USE POLICIES AND DOCUMENTATION

A. Machine-Generated Content

IBM’s licensing terms and conditions govern downstream applications and services that use IBM models. In addition, IBM is setting up an Acceptable Use Provision (AUP) that states guidelines and practices that the users of IBM models are required to follow as they develop and deliver downstream applications and services.

The AUP provides acceptable use of AI Models and confers to IBM the right to terminate the license to these models if necessary.

B. European Union-Specific Controls

The licensing terms and conditions to IBM Models are augmented with an Acceptable Use Policy (AUP) that states guidelines and practices that are specific to deployments of downstream applications and services in specific Countries.
<table>
<thead>
<tr>
<th>Source</th>
<th>Phase</th>
<th>Group</th>
<th>Risk</th>
<th>Indicator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>Training and Tuning</td>
<td>Fairness</td>
<td>Bias</td>
<td>Amplified</td>
</tr>
<tr>
<td>Input</td>
<td>Training and Tuning</td>
<td>Robustness</td>
<td>False samples</td>
<td>Traditional</td>
</tr>
<tr>
<td>Input</td>
<td>Training and Tuning</td>
<td>Value Alignment</td>
<td>Undesirable output for retraining purposes</td>
<td>New</td>
</tr>
<tr>
<td>Input</td>
<td>Training and Tuning</td>
<td>Data Laws</td>
<td>Legal restrictions on moving or using data</td>
<td>Traditional</td>
</tr>
<tr>
<td>Input</td>
<td>Training and Tuning</td>
<td>Intellectual Property</td>
<td>Copyright and other IP issues with content</td>
<td>Amplitified</td>
</tr>
<tr>
<td>Input</td>
<td>Training and Tuning</td>
<td>Transparency</td>
<td>Disclose data collected, who has access, how stored, how it will be used</td>
<td>Amplitified</td>
</tr>
<tr>
<td>Input</td>
<td>Training and Tuning</td>
<td>Privacy</td>
<td>Inclusion or presence of SPI or PII</td>
<td>Traditional</td>
</tr>
<tr>
<td>Input</td>
<td>Training and Tuning</td>
<td>Privacy</td>
<td>Provide data subject rights (e.g., opt-out)</td>
<td>Amplitified</td>
</tr>
<tr>
<td>Input</td>
<td>Inference</td>
<td>Privacy</td>
<td>Disclose PII or SPI as part of prompt to model</td>
<td>New</td>
</tr>
<tr>
<td>Input</td>
<td>Inference</td>
<td>Intellectual Property</td>
<td>Disclose copyright or other IP information as part of prompt to model</td>
<td>New</td>
</tr>
<tr>
<td>Input</td>
<td>Inference</td>
<td>Robustness</td>
<td>Vulnerabilities to adversarial attacks like evasion (create incorrect model output by modifying data sent to train model)</td>
<td>Amplitified</td>
</tr>
<tr>
<td>Input</td>
<td>Inference</td>
<td>Robustness</td>
<td>Vulnerabilities to adversarial attacks like prompt injection (force different output), prompt leaking (disclose system prompt), or jailbreaking (avoid guardrails)</td>
<td>New</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Fairness</td>
<td>Bias in generated content</td>
<td>New</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Intellectual property</td>
<td>Performance disparity across individuals or groups</td>
<td>Traditional</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Intellectual property</td>
<td>Copyright infringement, compliance with open source license agreements</td>
<td>New</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Value alignment</td>
<td>Hallucination (generation of false content)</td>
<td>New</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Value alignment</td>
<td>Toxic, hateful, abusive, and aggressive output</td>
<td>New</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Misuse</td>
<td>Spread disinformation (deliberate creation of misleading information)</td>
<td>Amplitified</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Misuse</td>
<td>Generate toxic, hateful, abusive, and aggressive content</td>
<td>New</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Misuse</td>
<td>Nonconsual use of people’s likeness (deepfakes)</td>
<td>Amplitified</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Misuse</td>
<td>Dangerous use (e.g., creating plans to develop weapons or malware)</td>
<td>New</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Misuse</td>
<td>Deceptive use of generated content (e.g., intentional nondisclosure of AI generated content)</td>
<td>New</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Harmful code generation</td>
<td>Execution of harmful generated code</td>
<td>New</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Privacy</td>
<td>Expose PI or SPI in generated content</td>
<td>New</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Explainability</td>
<td>Challenges in explaining the generated output</td>
<td>New</td>
</tr>
<tr>
<td>Output</td>
<td>Inference</td>
<td>Traceability</td>
<td>Challenges in identifying source and facts for generated output</td>
<td>New</td>
</tr>
<tr>
<td>Other</td>
<td>Governance</td>
<td>Transparency</td>
<td>Document data and model details, purpose, potential use and harms</td>
<td>Traditional</td>
</tr>
<tr>
<td>Other</td>
<td>Governance</td>
<td>Accountability</td>
<td>Identify responsibility for misaligned output along AI lifecycle and value chain</td>
<td>Amplitied</td>
</tr>
<tr>
<td>Other</td>
<td>Legal compliance</td>
<td>Intellectual property</td>
<td>Determine creator of downstream models</td>
<td>New</td>
</tr>
<tr>
<td>Other</td>
<td>Legal compliance</td>
<td>Intellectual property</td>
<td>Determine creator of open source foundation models</td>
<td>New</td>
</tr>
<tr>
<td>Other</td>
<td>Legal compliance</td>
<td>Intellectual property</td>
<td>Determine owner of AI-generated content</td>
<td>New</td>
</tr>
<tr>
<td>Other</td>
<td>Legal compliance</td>
<td>Intellectual property</td>
<td>Uncertainty about IP rights related to generated content</td>
<td>New</td>
</tr>
<tr>
<td>Other</td>
<td>Legal uncertainty</td>
<td>Legal uncertainty</td>
<td>Determine downstream obligations</td>
<td>Amplitied</td>
</tr>
<tr>
<td>Other</td>
<td>Societal impact</td>
<td>Impact on jobs</td>
<td>Human displacement (AI induced job loss)</td>
<td>Amplitied</td>
</tr>
<tr>
<td>Other</td>
<td>Societal Impact</td>
<td>Human dignity</td>
<td>Human exploitation (ghost work in training), poor working conditions, lack of healthcare, unfair compensation</td>
<td>Amplitied</td>
</tr>
<tr>
<td>Other</td>
<td>Societal Impact</td>
<td>Environment</td>
<td>Increased carbon emission (high energy requirements for training and operation)</td>
<td>Amplitied</td>
</tr>
<tr>
<td>Other</td>
<td>Societal Impact</td>
<td>Diversity and inclusion</td>
<td>Homogenizing culture and thoughts</td>
<td>New</td>
</tr>
<tr>
<td>Other</td>
<td>Societal Impact</td>
<td>Human agency</td>
<td>Misinformatio and disinformation generated by foundation models</td>
<td>Amplitied</td>
</tr>
<tr>
<td>Other</td>
<td>Societal Impact</td>
<td>Impact on education</td>
<td>Bypass learning process, plagiarism</td>
<td>New</td>
</tr>
</tbody>
</table>
C. Downstream Documentation

For downstream usage of its pre-trained models, IBM makes available the following documentation:

- Terms and Conditions
- Product documentation
- Technical reports, such as this report

Together, this information is designed so that not only IBM complies with legal and ethical requirements, but also the users of these models can comply with their own obligations.

1) Terms and Conditions: The latest Terms and Conditions for the watsonx platform can be found at https://www.ibm.com/support/customer/csol/terms/?id=i126-6883.

2) Product documentation: The IBM Granite models are currently available through IBM’s watsonx platform. As part of watsonx, each Granite model is accompanied by a model card that details key facts and provenance of the model.

VIII. Conclusion

In this technical report, we have presented IBM’s Granite family of foundation models designed for enterprise generative AI applications. IBM’s ethical and governance frameworks provide the context within which these models are created and made available. Aligned with IBM’s commitment to transparent and responsible AI, we have presented descriptions of exact datasets, pre-processing steps, training infrastructure, energy consumption, and testing/evaluation methodologies used throughout the model development lifecycle.

We are continuing to develop the Granite series in several directions. Whereas this initial Granite release only supports English, future models will be trained on multiple natural languages. Alongside, HAP annotation is being refined and expanded for additional languages. Furthermore, Granite models for other modalities such as code as well as industry-specific content are being developed. On the model safety evaluation front, we are developing a comprehensive re-teaming framework. The adversarial prompts will test the models across a variety of domains, including (but not limited to) HAP, bias and stigma, factual correctness and harmful topics.

We are continuing to develop additional data annotations for IBM’s curated pre-training dataset, such as scoring documents for their inclusion of personally-identifiable information and for their conversationality [68], [69]. We are working toward instrumenting our compute infrastructure to obtain precise rather than estimated measurement of energy and carbon footprints [70]. Finally, we are exploring the application of various methods for mitigating unwanted biases [71]–[73].

REFERENCES


APPENDIX A
RELEASE NOTES

September 15th, 2023

- Initial report released.

November 7th, 2023

- Table IV, updated with new values for FiQA - Opinion and Insurance QA metrics. New values were calculated after correcting a bug found in HELM’s ranking metric protocol. Oasst-sft-pythia-12b was additionally temporarily removed from the analysis as a benchmark, as it was not immediately available to the evaluation team to rerun after the HELM ranking metric fix was implemented.
- Several minor typo and grammar corrections updated throughout.