machine learning & knowledge extraction

MDPI

Article

Benchmarking with a Language Model Initial Selection for Text Classification Tasks

Agus Riyadi [1,2,*], Mate Kovacs [3], Uwe Serdült [3,4] and Victor Kryssanov [3]

[1] Graduate School of Information Science and Engineering, Ritsumeikan University, Ibaraki 5678570, Osaka, Japan
[2] Ministry of National Development Planning/BAPPENAS, Jakarta 10310, Indonesia
[3] College of Information Science and Engineering, Ritsumeikan University, Ibaraki 5678570, Osaka, Japan; kovacsm@fc.ritsumei.ac.jp (M.K.); serdult@fc.ritsumei.ac.jp (U.S.); kvvictor@is.ritsumei.ac.jp (V.K.)
[4] Center for Democracy Studies Aarau (ZDA), University of Zurich, 8006 Zurich, Switzerland
* Correspondence: agus.riyadi@bappenas.go.id

Abstract: The now-globally recognized concerns of AI's environmental implications resulted in a growing awareness of the need to reduce AI carbon footprints, as well as to carry out AI processes responsibly and in an environmentally friendly manner. Benchmarking, a critical step when evaluating AI solutions with machine learning models, particularly with language models, has recently become a focal point of research aimed at reducing AI carbon emissions. Contemporary approaches to AI model benchmarking, however, do not enforce (nor do they assume) a model initial selection process. Consequently, modern model benchmarking is no different from a "brute force" testing of all candidate models before the best-performing one could be deployed. Obviously, the latter approach is inefficient and environmentally harmful. To address the carbon footprint challenges associated with language model selection, this study presents an original benchmarking approach with a model initial selection on a proxy evaluative task. The proposed approach, referred to as Language Model-Dataset Fit (LMDFit) benchmarking, is devised to complement the standard model benchmarking process with a procedure that would eliminate underperforming models from computationally extensive and, therefore, environmentally unfriendly tests. The LMDFit approach draws parallels from the organizational personnel selection process, where job candidates are first evaluated by conducting a number of basic skill assessments before they would be hired, thus mitigating the consequences of hiring unfit candidates for the organization. LMDFit benchmarking compares candidate model performances on a target-task small dataset to disqualify less-relevant models from further testing. A semantic similarity assessment of random texts is used as the proxy task for the initial selection, and the approach is explicated in the context of various text classification assignments. Extensive experiments across eight text classification tasks (both single- and multi-class) from diverse domains are conducted with seven popular pre-trained language models (both general-purpose and domain-specific). The results obtained demonstrate the efficiency of the proposed LMDFit approach in terms of the overall benchmarking time as well as estimated emissions (a 37% reduction, on average) in comparison to the conventional benchmarking process.

Keywords: language model benchmarking; machine learning model selection; carbon emission reduction

Academic Editor: Irena Spasić

Received: 24 October 2024
Revised: 23 December 2024
Accepted: 3 January 2025
Published: 5 January 2025

Citation: Riyadi, A.; Kovacs, M.; Serdült, U.; Kryssanov, V. Benchmarking with a Language Model Initial Selection for Text Classification Tasks. *Mach. Learn. Knowl. Extr.* **2025**, *7*, 3. https://doi.org/10.3390/make7010003

1. Introduction

In the past few years, concerns surrounding the substantial energy consumption and ever-increasing computational resource demands of artificial intelligence (AI) have rapidly proliferated through mass media but also in scholarly publications [1,2]. Schwartz et al. [3] introduced the terms of "Green AI" and "Red AI" to delineate studies dealing with the environmental implications of AI, as opposed to those primarily centered on improving the AI "accuracy". A survey of Green AI publications by Verdecchia et al. [2] identified popular research topics and categorized the reported studies. The top four (by the number of studies) categories include reports on methodologies for monitoring carbon footprints, assessments of the impact of model hyperparameter tuning, approaches to benchmarking the carbon footprints of AI models, and analyses of the environmental effects of various model deployment strategies. Notably, it was observed that within the realm of benchmarking efforts, there had been few studies offering practical solutions for reducing AI carbon footprints as such.

Benchmarking is a pivotal component in the evaluation of AI model performance [4]. It leverages designated metrics to facilitate comparisons among models, thereby identifying the "best" or optimal performer for any given AI task [5]. The execution of model comparison presents a formidable challenge, typically requiring iterative runs with all candidate models for the specified task before the best model would be determined. This exhaustive nature of the benchmarking process, although usually successful in establishing the top performers, bears environmental implications of ever-larger carbon footprints, as more and more models need to be compared. Strubell et al. [1] estimated CO_2 emissions generated from one AI model training on a GPU for a natural language processing (NLP) task, including hyperparameter tuning and subsequent experiments. The authors revealed that one-model emissions are on par with those produced by a car in half of its lifetime and are seven times the average of a human's emissions over a year.

Metaphorically, AI model selection can draw on similarities with the process of selecting qualified candidates from a pool of applicants for a job in an organization. The process entails candidate screening, involving evaluative examinations [6], which may be in the form of proxy measures for the real job duties. Paralleling the AI benchmarking process, the personnel candidate selection shares a similar objective, namely identifying the candidates most adept for the given task. Unlike the case of personnel recruitment, however, conventional AI benchmarking typically skips the initial candidate selection part but ends up with a costly task execution for all candidates. One possible solution to reduce the environmental costs of AI model benchmarking would, then, be to complement the process with a simplified evaluative procedure helping to forecast the models' performances before they are actually tested with the task at hand. To the authors' best knowledge, there have been only few, yet premature, attempts to develop a framework, methods, or metrics that would allow one to vet AI models by predicting their performances at the stage preceding full-scale experiments (e.g., see [7–10]). Hence, there is a need for additional efforts in this research direction.

The goal of this study was, therefore, to develop an approach to AI model benchmarking that would incorporate candidate model initial selection aimed at the reduction of carbon emissions associated with the benchmarking process. To realize the metaphor of human candidate selection in the benchmarking domain, the authors focused on text classification, a fundamental task of NLP [11,12]. Also, this study exclusively dealt with BERT-based language models, as they are increasingly and successfully used for text classification in various contexts [13].

The projected original contribution of the presented study is twofold:

- A new approach to model benchmarking called Language Model-Dataset Fit (LMDFit) benchmarking is devised. The approach allows for substantially reducing carbon emissions associated with model testing by implementing a candidate model initial selection for a target dataset prior to model performance assessment. The efficiency of the LMDFit approach was verified through extensive experiments in comparison with the conventional benchmarking process. The application of the proposed approach allowed for emission reductions in the range of 10 to 75% (37% on average), depending on the classification task at hand.
- The mean and skewness vector of the semantic similarity score distribution is shown to be a reliable group-predictor of language model classification performance for a given dataset. Unlike the existing approaches to forecasting AI model performance, which aim to rank candidate models for further experiments, the two-vector of text similarity statistics can be used to categorize all candidate models as either "more fit" or "less fit". All "more-fit" models are then to be analyzed in benchmarking experiments that may result in not as dramatic emission cuts as in the case when only a few top-ranked models would be considered. On the other hand, this conservative approach is more robust and secure, as it minimizes the risk of inadvertently cutting off relevant models due to noise or bias in the data or statistics used for the initial categorization.

The rest of the paper is organized as follows. Section 2 surveys the related work. Section 3 describes the resources used, and Section 4 explains the study's methodology and introduces the LMDFit benchmarking approach. Section 5 presents experiments, while Section 6 discusses the experimental results obtained. Finally, Section 7 provides conclusions and outlines plans for future work.

2. Related Work

This section gives an overview of recent studies focused on the computational cost assessments and environmental impacts of AI, existing practices of language model benchmarking, as well as the model performance prediction in NLP.

2.1. AI Costs and Environmental Impacts

While AI offers substantial benefits for addressing many global challenges across diverse domains, its negative impact on the environment can no longer be ignored. Schwartz et al. [3] reported that the computational costs of AI research attained an astounding 300,000-times increase in the period from 2012 to 2019, doubling every month. The authors uncovered that this surge should mainly be attributed to the research community's focus on performance rather than on computational efficiency improvements. The continually growing computing requirements of machine learning models constitute a major predicament for tackling the AI carbon emission problem [14].

Acknowledging the environmental consequences of the broad introduction of machine learning models in the industry but also in our daily lives, there is a growing interest in the so-called "Green AI" research [2,3]. Strubell et al. [1] were among the early contributors, followed by many others, who investigated the environmental impact of AI training. Verdecchia et al. [2] classified the related publications by 14 topics (such as "monitoring", "estimation", "emissions", etc.) and three types of studies (such as "solution", "observation", and "position"). While "model benchmarking" was found among the top three research topics, only three of the 17 papers in this category offered practical solutions. Also, all surveyed research on this topic appeared to be in the early, if not preliminary, stages. Presently, therefore, there is a well-recognized need for additional efforts in the realm of model benchmarking that would lead to practical solutions allowing one to mitigate the environmental problems associated with AI (also, see [15]).

2.2. Language Model Benchmarking

Benchmarking, a term defined differently across domains, usually presumes the general concepts of measurement, comparison, identification of best practices, as well as implementation and improvement [16]. In NLP, to provide for a fair comparison of machine learning models, a number of benchmarking approaches have been proposed [17–19]. These efforts have resulted in open standardized resources intended to support various NLP downstream tasks, such as text classification, question–answering, summarization, and text similarity assessment. Various metrics, such as accuracy [20], energy consumption [1], justice [21], and fairness [22], have been used to compare model performance. Among the models investigated, transformer-based pre-trained language models (PTLMs) [23] have recently received a great deal of research attention. According to a survey conducted by Casola et al. [24], the number of PTLM studies has especially grown in the past five years. The authors identified the most frequently used PTLMs as BERT [20], RoBERTa [25], DistilBERT [26], XLNet [27], and ALBERT [28]. Figure 1a outlines the developmental steps all these transformer-based models undergo before they would be employed to solve a downstream NLP task. Figure 1b then illustrates the model benchmarking process, where various pre-trained models are compared to identify the best-performing one for the task at hand. It is understood that model fine-tuning with its "redundant" retraining of all candidate PTLMs is the major source of carbon emissions associated with benchmarking. One way to make PTLM benchmarking "greener" is to try to predict each candidate model's performance, based on its pre-trained state, before actually fine-tuning it. In the following subsection, recent studies in this direction are surveyed.

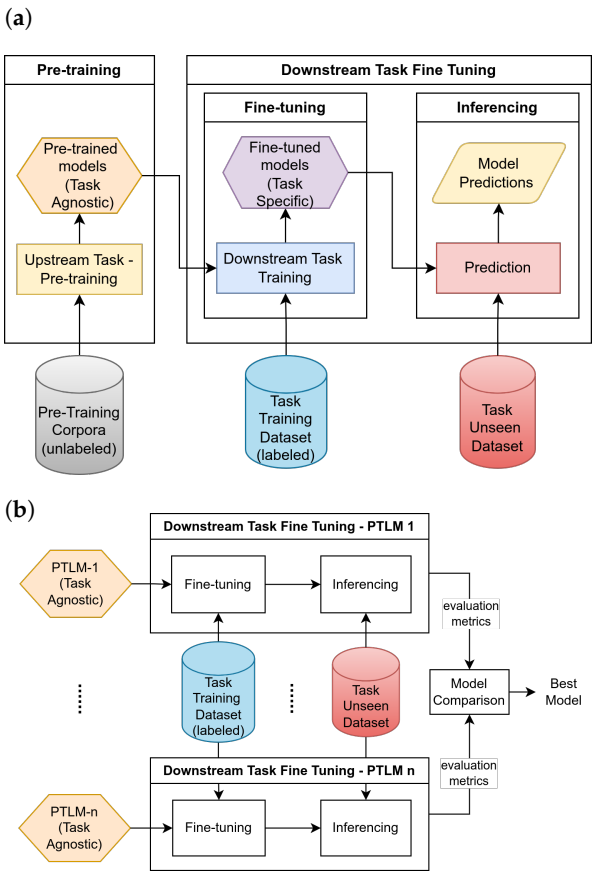

Figure 1. Pre-trained transformer-based model utilization (**a**) and the model benchmarking process (**b**).

2.3. Language Model Performance Prediction

There have been several attempts to develop methods for forecasting the performance of an NLP model on a downstream task that would not require fine-tuning the model. Xia et al. [9] suggested forecasting the model's "potential" performance based on its past experimental records. The authors constructed several regression models to predict the model evaluation score, using experimental parameter settings as the input. Ahuja et al. [7] proposed a similar approach to estimate the performances of multilingual models for different tasks and languages. Ye et al. [29], on the other hand, investigated the predictive potential of fine-grained performance measures that would deal with not only holistic aspects but also with task-specific performance. It should be noted, however, that all these approaches require data from past experimental records to make predictions. In other words, the proposed forecasting methods have a "hidden" component of extensive model training to produce the records. The latter could hardly lead to significant emission cuts and is not applicable in the case of new models or those with a short utilization history.

To address this predicament, Kadiķis et al. [10] suggested an alternative approach. The authors considered an abductive natural language inference task (that is, a binary classification task) that entails selecting the most probable hypothesis for a given set of observations. The prediction problem was formulated as a ranking problem for an indicator (metric) for model selection. The study deployed the cosine similarity score of the input and hypothesis text vectors computed with pre-trained models, and labeled data were used as "ground truth" to assess the obtained ranking of the pre-trained models. It was found that there is a strong correlation between the text cosine similarity and the model classification accuracy, and that the text similarity assessment would serve as a proxy task for predicting the classification performance of pre-trained models. The methodology proposed in [10], while appealing with its ground-breaking potential, requires labels (hypotheses) to be compared with the input texts, which makes it impractical in the case of unsupervised classification. Also, the authors did not go beyond the binary inference task, did not investigate fine-tuned model performances, and focused on predictions in terms of accuracy only. The latter metric can be inadequate when the classes are imbalanced [30].

Striving to exclude exhaustive model performance testing from classification experiments, there have been several attempts to propose measures for forecasting a model's generalization capability. Jiang et al. [31] investigated the predictive potential of over 40 measures on more than 10,000 models developed for image classification. Although the study produced no results that could be considered consistent in general, several complexity measures were identified as promising for future research. Building on [31], Dziugaite et al. [32] further examined various complexity measures through extensive experiments and concluded that none is a reliable indicator of a model's generalization capability. Martin et al. [33] suggested to reckon a model's expected performance based on metrics assessing its internal structure. It was found that certain power-law-based metrics can be used to detect poorly trained models, while some other metrics allow for a comparison of models in terms of their generalization potential. In a recent study, Yang et al. [34] developed an empirical spectral density metric and demonstrated its effectiveness for attesting the model's ability to generalize. It should be noted, however, that this and other similar research works focused on establishing a model's overall "qualifications" rather than on what model would perform best in a given context. Numerous experiments [35,36] have shown that the same model could perform very differently on different downstream tasks. Therefore, it is hard to expect that any measure computed outside of a task-specific performance assessment experiment would provide for a reliable predictor of a model's behavior with real-world data. Metrics should, thus, be proposed to complement and improve (e.g., in terms of emissions), rather than to replace, benchmarking experiments.

Motivated by [10], the presented work aimed to understand if text similarity assessment would serve as a universal proxy assignment to forecast the performances of pre-trained models for a specific text classification task. The next section describes data and other resources collected to form an experimental basis for the proposed methodology of model initial selection.

3. Resources Used

This section elaborates on the resources, including PTLMs and open datasets, and describes evaluation metrics used in the experiments.

3.1. Pre-Trained Language Models

Recently, there has been a dramatic surge in the applications of various PTLMs across different domains [37]. A major reason for the still ongoing increase in the number of pre-trained language models is the now well-established fact that when dealing with downstream tasks, models pre-trained with general corpora perform worse than those pre-trained with domain-specific data [35,38]. For instance, SciBERT, a model trained on academic papers far outperformed the more general BERT model in solving science-related NLP assignments [35]. The performance of a particular PTLM may, however, vary depending on not only the task at hand but also the specific dataset it would need to process. For this study's experiments, seven popular BERT-based models, both general-purpose and domain-specific, sourced from diverse domains were chosen (see Table 1). All these models were originally pre-trained with English texts in uncased settings with a masked language model (MLM) objective and 110 M parameters. Also, they all have 12 layers, a hidden layer size of 768, 12 attention heads, and a vocabulary size of approximately 30,000 tokens.

Table 1. Pre-trained language models tested.

No.	Name	Huggingface * Address	Pre-Training Approach	Training Data
1	BERT [20]	bert-base-uncased	from scratch	Wikipedia and Books Corpus
2	SciBERT [35]	allenai/scibert_scivocab_uncased	from scratch	Scientific papers
3	LegalBERT [36]	nlpaueb/legal-bert-base-uncased	from scratch	Legal text (court cases, contracts, legislations, etc.)
4	FinancialBERT [39]	ahmedrachid/FinancialBERT	from scratch	TRC2-financial corpus, Bloomberg news articles, corporate reports, and earning call transcripts
5	PharmBERT [40]	Lianglab/PharmBERT-uncased	fine-tuned from BERT (1)	DailyMed drug labels
6	Agriculture BERT	recobo/agriculture-bert-uncased	fine-tuned from SciBERT (2)	National Agricultural Library (NAL) documents and agricultural literature
7	Chemical BERT	recobo/chemical-bert-uncased	fine-tuned from SciBERT (2)	Chemical industry domain documents and Wikipedia chemistry documents

* https://huggingface.co/, accessed on 5 September 2024.

3.2. Datasets

Eight text classification datasets were selected for the experiments. The datasets are publicly accessible, have previously been used in numerous studies, and are, therefore, well documented. A synopsis of the datasets is given below (all data discussed are in a textual format):

(a) Environmental claims (https://huggingface.co/datasets/climatebert/environmental_claims, accessed on 5 March 2024). The set contains 2647 real-world environmental claims mostly in the financial domain by listed companies [41]. The data were annotated by 16 domain experts and have been used in studies, such as [42,43]. It is to support the task of environmental claim detection, which is a sentence-level binary classification task. The set includes both true claims (approximately 25%) and false claims (approximately 75%) with an average word token count of 27.61 per sentence.

(b) AGNews (https://huggingface.co/datasets/ag_news, accessed on 5 March 2024) is a large collection of news articles. It serves to support the general task of text classification [44]. The news articles are categorized into four classes: "World", "Sport", "Business", and "Science/Technology". The set comprises 120,000 training samples and 7600 testing samples with equal representation for each class. The average word token count across all articles is 43.93. This dataset has been widely used for benchmarking purposes in NLP (e.g., see [45,46]).

(c) Financial phrase-bank (https://huggingface.co/datasets/financial_phrasebank, accessed on 5 March 2024). The dataset was created to support sentiment analysis in the financial domain [47]. It contains phrases selected from financial news articles and company press releases. The phrases were labeled by 16 human annotators as "positive", "negative", or "neutral". The data have been used by several research groups (e.g., [48,49]). For the purposes of the presented study, a sample consisting of phrases with an over 50% inter-annotator agreement was compiled from the data. The sample includes 4,840 financial statements classified as "negative" (59.41% of the total), "positive" (28.13%), or "neutral" (the rest). On average, one phrase in the sample has 23.15 tokens.

(d) Rheology (https://huggingface.co/datasets/bluesky333/chemical_language_understanding_benchmark, accessed on 5 March 2024) is a subset of the Chemical Language Understanding Benchmark collection [50]. The dataset is meant to support a sentence-level classification task. It consists of 2017 single-labeled sentences from research papers in the chemistry domain with a sentence average length of 39.67 tokens. The sentences are organized into five classes exemplifying different polymer structures and properties.

(e) Plant-chemical relationship corpus (http://gcancer.org/plant_chemical_corpus/, accessed on 5 March 2024). The data comprise 939 documents describing plant–chemical relationships [51]. The relationships were annotated by experts with labels of either a "positive" or "negative" containment of chemicals in the plants. The set has been used in NLP research to support the named entity recognition task [52]. For the purposes of this study, abstract sentences and plant and chemical element names of the set were concatenated to form the input for the language models using the following template: "Relation of {plant} and {chemical} on {sentence}". The average length of the concatenated text is 41.92 tokens.

(f) arXiv (https://www.kaggle.com/datasets/Cornell-University/arxiv, accessed on 5 March 2024). The arXiv collection is maintained by Cornell University. It includes over 1.7 million scholarly articles publicly available on arXiv.org. The dataset has been used in multiple text classification studies (e.g., see [53,54]). By design, it supports multi-label classification, as each archived article can belong to more than one field of study. The stored articles are supplemented with extensive metadata, such as versions, titles, authors, categories, and abstracts. For the purposes of this study, a random sample of 51,774 article abstracts (https://www.kaggle.com/code/chhatrabikramshah123/researchpaperrecommendation/input?select=arxiv_data_210930-054931.csv, accessed on 5 March 2024) was utilized. The nine largest category names plus "Other" were used as labels for the texts. The average token count is 194.50 per abstract in the sample.

(g) The European Court of Human Rights (ECtHR) cases (https://huggingface.co/datasets/ecthr_cases, accessed on 5 March 2024). This is a commonly used benchmarking dataset for NLP in the legal domain (see [55] for a related study). The data include facts from 11,000 ECtHR cases. The facts are multiple-labeled by designations

of the European Convention of Human Rights (ECHR) articles they violate. There are 33 labels, and the average case size is 1918.76 word tokens.

(h) Ohsumed (http://disi.unitn.it/moschitti/corpora.htm, accessed on 5 March 2024) dataset consists of abstracts of MEDLINE records on cardiovascular diseases registered in 1991. There are 34,389 abstracts in total, and the average abstract size is 115.75 tokens. The abstracts are multiple-labeled by 23 specific cardiovascular conditions. The data are meant to support the multi-label multi-class text classification task, and the dataset has been widely used in NLP research (e.g., see [56,57]).

3.3. Efficiency Measures

To assess the "efficiency" of the benchmarking process, two metrics typically used in studies focused on AI carbon footprint (e.g., see [58,59]) were evaluated: computation time and carbon emissions. Computation time quantifies the computing time differences, while carbon emissions estimate the environmental impact of computing. The time metric assumes the standard Unix time format, and the carbon emissions are calculated using the CodeCarbon (https://pypi.org/project/codecarbon/, accessed on 5 March 2024) Python package. The latter is an open-source software tool that allows for assessing CO_2 emissions based on the hardware energy consumption data and the regional carbon intensity values of the computing location. In the experiments of this study, emissions are calculated, assuming Japan's carbon intensity of 482.0 gCO_2/kWh (in the energy mix of 2020 with 74.5% fossil fuels, 0.3% geothermal, 9.7% hydro, 4.6% nuclear, 9.7% solar, and 1.1% wind energy).

4. LMDFit and Model Initial Selection

This section provides an overview of the proposed approach, explains its underlying assumptions, and details the model assessment methodology.

4.1. Overview

The approach proposed in this study draws inspiration from well-established theories in the realm of personnel candidate selection, which is a perpetual challenge in human decision-making. These theories include Human Capital Theory [60], Person–Organization Fit [61], and Person–Job Fit [62]. Human Capital Theory views individuals' learning capacities as akin to profits of organizations [63]. Choosing the right candidate is, therefore, important to maximize returns to the organization. The Person–Organization Fit theory, on the other hand, scrutinizes the compatibility of individuals with the organizations they would work for [61]. The Person–Job Fit theory then provides for an analysis of what individual skills would fit the requirements and characteristics of a particular job [62]. All theories consider the personnel hiring process as a (more or less) linear succession of organizational activities and decisions made, as summarized in Figure 2.

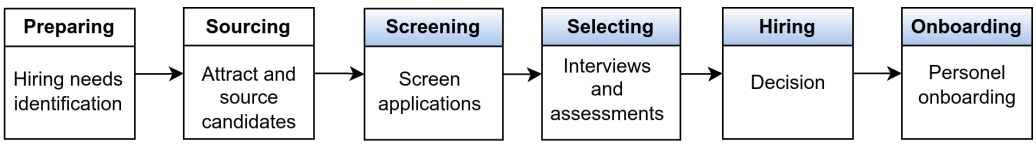

Figure 2. Personnel hiring process.

The proposed LMDFit approach aims to optimize the AI model benchmarking process by minimizing both time spent and emissions produced. Metaphorically, PTLM M_i, $i = 1, \ldots, N$, where N is the number of candidate models for benchmarking, is thought to correspond to an individual applying for a job. The dataset D is to "represent" the organization, and (NLP) task T_i is to stand for a job within the organization. Model selec-

tion is then organized in four steps, as illustrated in Figure 3a. More specifically, LMDFit benchmarking assumes the following processes:

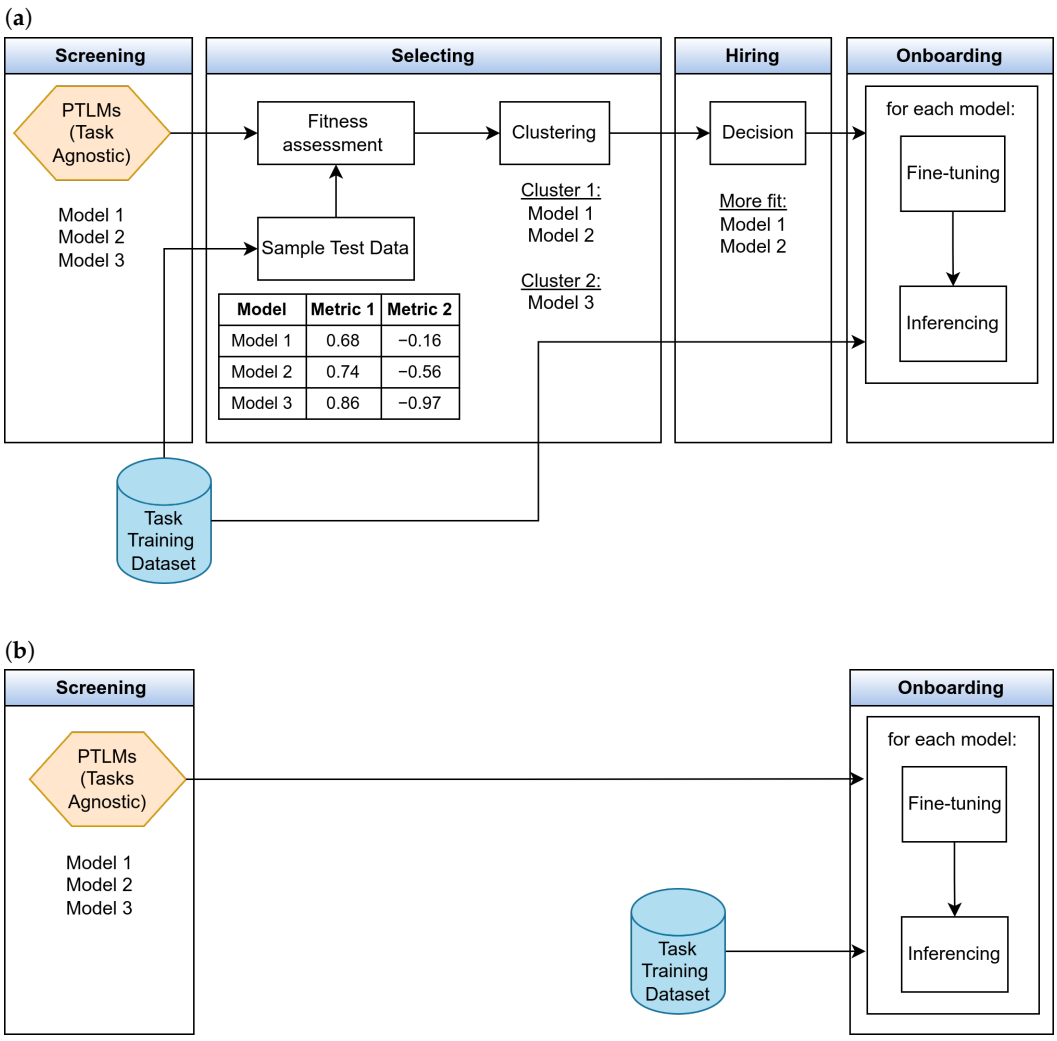

Figure 3. The LMDFit approach (**a**) vs. conventional model benchmarking (**b**).

1. **Screening**. This step entails recruiting pre-trained model candidates. The candidates M are manually selected based on model specifications (including the language and training corpora), utilization precedents, and other relevant information. The screening process is meant to gather potential candidates for benchmarking.
2. **Selecting**. The second step is to assess the fitness of each candidate-model M_i for the target task T_i. All mobilized models are classified as either less-fit or more-fit based on how they perform on a proxy evaluative task. The latter task is to assess "basic skills"—the abilities to differentiate and to generalize texts—of the candidate-models.
3. **Hiring**. A (subjective) decision-making process for deciding which models should be benchmarked. The fitness assessment results are used to reduce the number of models to be tested.
4. **Onboarding**. In this study, it refers to the standard benchmarking process. Onboarding, thus, also includes all "must-have" procedures before the models could be deployed in comparison experiments, such as fine-tuning and inferencing.

Contrasting the conventional approach, LMDFit benchmarking focuses on candidate models that are more likely, performance-wise, to succeed with the task at hand. The overhead of the two additional steps—Selecting and Hiring (see Figure 3a)—is relatively low,

as they do not require model fine-tuning or any processing of the whole dataset associated with the target task. Model initial selection, i.e., "Selecting" of LMDFit is accomplished while testing pre-trained candidate models on a limited (in terms of both complexity and data involved) evaluative task. Based on their performance in the tests, all models are categorized as either less-fit or more-fit. Only models of the latter category are then used in computationally extensive benchmarking experiments. Details of the PTLM fitness assessment procedure, including the proxy evaluative task, data and metrics used, and the clustering algorithm, are presented in the next subsection.

4.2. Candidate Model Fitness Assessment

4.2.1. Assumptions

Text classification, a fundamental problem of NLP [11,12], is considered as the target "job" of Onboarding (Figure 3a) in this research. The fitness of PTLMs for a specific task T_i is proposed to estimate, computing the distribution of (semantic) similarity of texts intended for classification. Distributional semantics assumes that linguistic meaning is reflected in the distributional patterns of words in large corpora. According to the distributional hypothesis, "a word is characterized by the company it keeps"; hence, linguistic items with similar distributions have similar meanings [64,65]. The latter is the theoretical foundation of vector-space models for the semantic processing of a text [66]. If two texts are semantically similar, their embedding vectors should lie in a close proximity in the vector space. Conversely, if they differ, their vectors should be more distant. This implies that a model producing embeddings that better reflect the intrinsic semantic distinctions of a given dataset is more "discriminative".

Text similarity assessment or the ability to semantically differentiate between one text and another can be realized with practically any PTLM. To that end, cosine similarity, which is a popular text similarity measure, was used for the model initial selection. The efficacy of cosine similarity has been demonstrated in numerous studies, confirming its strong correlation with human judgment of semantic relatedness [67,68]. Furthermore, unlike other relevant metrics, cosine similarity does not suffer from the so-called "curse of dimensionality" phenomenon in high-dimensional embeddings spaces, where the discriminating power of many proximity measures, such as Euclidean and Manhattan distances, diminishes [69]. Cosine similarity computes the cosine of the angle of two embedding vectors that are "in-model" representations of words comprising the compared texts [70]. Two identical representations correspond to a cosine similarity of 1, and the smaller the measure value (bounded by 0), the further apart the representations. Similar texts are naturally expected to have close in-model representations and are, therefore, more difficult to differentiate (e.g., assign different labels, etc.) with the model (also, see [10]).

For a random sample of texts, their cosine similarity is asymptotically Gaussian distributed [71]. Figure 4a exemplifies cosine similarity distributions obtained with three different PTLMs on the same document collection. Similarity scores were computed through a pairwise comparison of 200 documents randomly selected from the financial phrase-bank collection (see Section 3.2). All three distributions are left-skewed, and the asymmetry is more pronounced as the mean moves to the right. This phenomenon can be attributed to two major factors: (1) the distributions are truncated owing to the restricted range of the cosine similarity measure, and (2) some of the higher similarity scores may be due to the insufficient number of dimensions of the model's embedding space to quantify fine-grain differences in the texts (see, for instance, ref. [72] for how dimensionality reduction affects the ability of cosine similarity to differentiate). It appears natural to expect that models producing distributions with means closer to 1 would perform worse, on average, when classifying the corresponding texts. At the same time, however, a distribution with

pronounced left skewness would signal that the corresponding model would "overfit" by differentiating similar texts, owing to a lack of generalization ability.

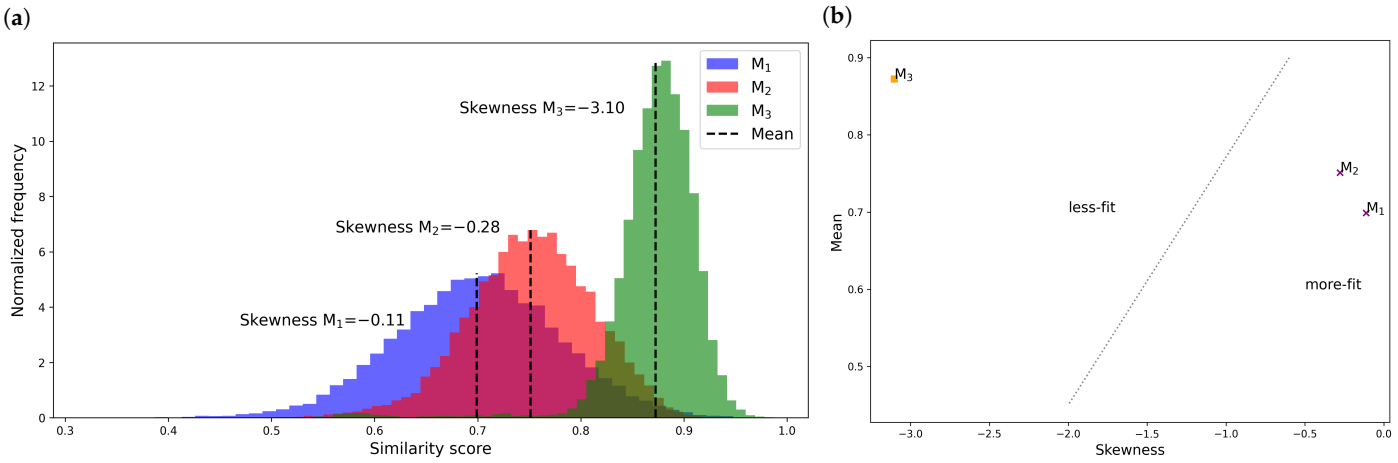

Figure 4. Cosine similarity distributions obtained with three different models M_1, M_2, and M_3 on the same data (**a**). Categorization of the models into "less-fit" and "more-fit", based on the two-vector of cosine similarity mean and skewness (note that the dotted line is not to set any threshold or the like) (**b**).

We, therefore, assume the cosine similarity calculation as the proxy evaluative task performed on D to assess the fitness of candidate models M for the given classification task. With each candidate model M_i, two statistics—the mean similarity score and the similarity distribution skewness—are obtained by computing the cosine similarity for pairs of documents in D. The two-vectors of the statistics are used to categorize the corresponding models into either "less-fit" or "more-fit" classes in an unsupervised manner. Models producing cosine similarity distributions further skewed to the left and with higher mean-values are considered less-fit for the target classification task (for illustration, see Figure 4b). As a result, the number of candidate models for computationally extensive benchmarking (i.e., Onboarding) can be reduced, with only "more-fit" models being selected for the final evaluation.

4.2.2. Sampling and Implementation Details

The dataset intended for benchmarking experiments, which include model fine-tuning, can be very large. For the model initial selection, D_{sample} a (much) smaller subset of D is created by sampling from the benchmarking data. As the cosine similarity statistic is known to correlate with the text size, especially in the case of short documents [73], stratified quota sampling [74] is performed to reduce the text size effects on model selection. The texts are first grouped in 100 strata based on their size (in tokens), and D_{sample} is generated by randomly selecting texts from the strata in equal proportions. To determine the optimal sample size, experiments were conducted, monitoring fluctuations of the cosine similarity mean and skewness statistics. The skewness was estimated, calculating the adjusted Fisher–Pearson coefficient of skewness [75]. This allowed us to reduce the influence of data outliers on the estimates obtained in the case of short texts. The arXiv, Ohsumed, and AGNews datasets (see Section 3.2) and the BERT (general-purpose) and SciBERT (domain-specific) PTLMs (see Table 1) were used in the experiments. Figure 5 presents the experimental results obtained.

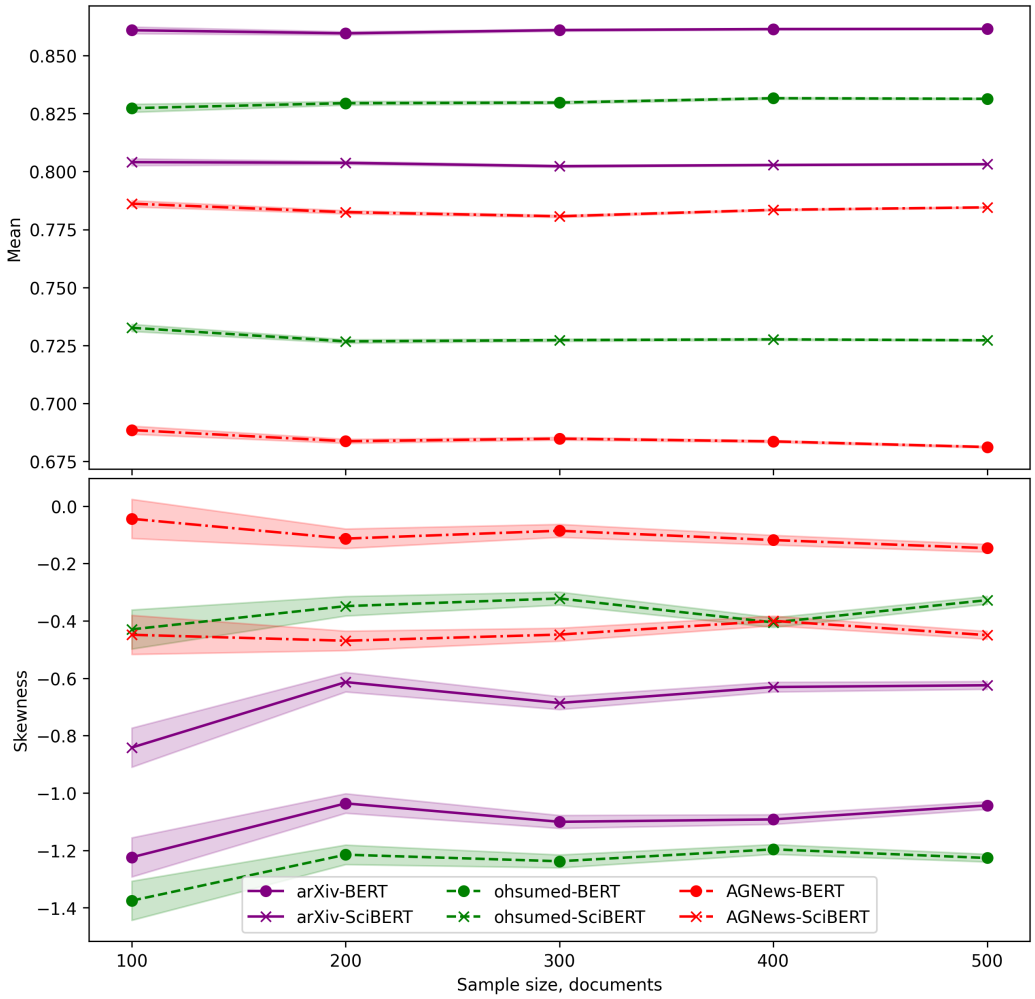

Figure 5. The impact of sample size on the cosine similarity mean and skewness. Color bands show 95% confidence intervals around lines connecting the values obtained by averaging the corresponding statistic over 10 samples of the same size.

As one can see from Figure 5, there are no major changes in the calculated values, beginning from the sample size of 200 almost in all cases. Therefore, the size of D_{sample} was set to 200 (which entails 19,900 text similarity comparisons per distribution computed) for the model initial selection in all experiments conducted in this study.

Once created, the same D_{sample} is used to obtain similarity score distributions and the mean and skewness statistics for all models recruited. *k*-means clustering [76] is performed to group the obtained two-vectors into two classes. Per-class average skewness is then calculated. The PTLMs of the class with an average skewness closer to 0 are labeled as "more-fit" (while the other class models are labeled as "less-fit" and are typically excluded from further consideration).

Algorithm 1 provides implementation details of the model initial selection procedure. Key parameter settings and other model- and task-specific characteristics are given in the next section.

Algorithm 1: model initial selection

```
Data: model_candidates, dataset, sample_size
Result: more_fit_models

function prepare_data(d: dataset, size: sample_size = 200):
    dataset_strata ← group_dataset(d, 100)  ▷ Group dataset into 100 strata based
     on token sizes
    data ← get_sample(dataset_strata, size) ▷ Random sampling to group_dataset
    data ← clean_punctuation(data)
    return data

function compute_embeddings(text: text, model:model, tokenizer:tokenizer):
    inputs ← tokenizer(text)
    return model(inputs).last_layer.mean

function get_embeddings(data:data, model:model, tokenizer:tokenizer)):
    embeddings ← empty list
    foreach text ∈ data do
        embeddings.append(compute_embeddings(text, model, tokenizer))
    end
    return embeddings

function calculate_similarity(pairs: pairs):
    result ← empty list
    foreach pair in pairs do
        similarity ← cosine_similarity(pair)
        append similarity to result
    end
    return result

# Step 1: Data preparation
d_sample ← prepare_data(dataset, sample_size)

# Step 2: Fitness computation
results ← empty list
foreach model ∈ model_candidates do
    tokenizer ← tokenizer_from_pretrained(model)
    embeddings_data ← get_embeddings(text_data, model, tokenizer)
    pairs ← pair_every_permutation(embeddings_data)
    results[model] ← calculate_similarity(pairs)
end

# Step 3: Clustering into two groups
model_clusters ← get_clusters(Clustering(n_clusters = 2), results)
avg_cluster_1 ← average_skewness(model_clusters, 0) ▷ Get skewness average of
 models in Cluster 1
avg_cluster_2 ← average_skewness(model_clusters, 1) ▷ Get skewness average of
 models in Cluster 2
```

Algorithm 1: *Cont.*

```
if avg_cluster_1 > avg_cluster_2 then
    more_fit_models ← get_models(model_clusters, 0)   ▷ Cluster 1 as more-fit
      models;
end
else
    more_fit_models ← get_models(model_clusters, 1)   ▷ Cluster 2 as more-fit
      models;
end
return more_fit_models
```

5. Experiments

This section presents comparison results of the LMDFit and conventional benchmarking approaches obtained with seven popular PTLMs on eight open data collections. Figure 3a portrays the design of the benchmarking experiments, while Section 3 provides specifications of the PTLMs and data used.

A single NVIDIA GeForce RTX 4090 GPU was employed for all computations. The fine-tuning process assumed retraining all layers of the tested models for three epochs (which is a standard practice in benchmarking, e.g., see [20,77]). The initial model weights were always the default weights of the given PTLM. The models were compared through a five-fold cross-validation. The number of batches for training and evaluation was set to 64 by default but was adjusted as necessary to accommodate memory constraints. In single-label classification experiments, the learning rate was set to 2×10^{-5} to ensure steady updates of the pre-trained weights. In multiple-label experiments, a learning rate of 5×10^{-5} was used to accelerate the learning process. In both cases, the weight decay parameter was set to 0.01 to prevent model overfitting. To construct embedding vectors, the inputs were formed, assuming a maximum length of 512 tokens with truncation or padding, as necessary. All models tested were deployed with their original tokenizers.

In the experiments, the models' performance was assessed, using F1 micro and macro scores (conventional benchmarking), as well as cosine similarity distribution skewness and mean (LMDFit model initial selection).

5.1. Environmental Claims Collection

The onboarding (benchmarking) task of single-label binary classification was performed with 2118 documents allocated for training and 529 documents for validation. Table 2 lists metric values obtained in the experiments. Table 3 then compares the benchmarking approaches in terms of computational time and emissions.

Table 2. Values of the LMDFit and conventional benchmarking metrics averaged in 5-fold cross-validation on the environmental claims dataset (sorted by the cosine similarity mean value).

	Model	LMDFit Metrics		Conventional Metrics *	
		Mean	Skewness	F1 Micro	F1 Macro
more-fit	BERT	0.712	−0.335	0.892	0.858
	SciBERT	0.740	−0.445	0.898	0.871
	PharmBERT	0.748	−0.512	0.887	0.855
	FinancialBERT	0.774	−0.323	0.890	0.857
	Chemical BERT	0.832	−0.659	0.853	0.810
	Agriculture BERT	0.841	−0.422	**0.911**	**0.887**
less-fit	LegalBERT	0.880	−3.082	0.855	0.819

* Best results are shown in bold.

Table 3. The computing time and the environmental impact associated with the LMDFit and conventional benchmarking approaches for the environmental claims data.

Task	LMDFit		Conventional	
	Time (s)	Emissions (g)	Time(s)	Emissions (g)
Fitness assessment	51.2	1.1	-	-
Clustering and Hiring	2.1	<0.1	-	-
Benchmarking	1419.7	47.8	1646.3	55.5
Total	1473.0	48.8	1646.3	55.5

5.2. AGNews

The benchmarking task is a single-label four-class classification. In each fold of model fine-tuning, approximately 102,000 documents were used for training and the rest for validation. Table 4 gives the model initial selection and benchmarking metric values obtained, while Table 5 compares the two approaches in terms of the computational time and carbon emissions.

Table 4. Values of the LMDFit and conventional benchmarking metrics averaged in 5-fold cross-validation on the AGNews dataset (sorted by the cosine similarity mean value).

No	Model	LMDFit Metrics		Conventional Metrics *	
		Mean	Skewness	F1 Micro	F1 Macro
more-fit	BERT	0.681	−0.141	**0.948**	**0.948**
	PharmBERT	0.779	−0.397	0.945	0.945
less-fit	SciBERT	0.777	−0.589	0.943	0.943
	FinancialBERT	0.800	−0.635	0.928	0.928
	Agriculture BERT	0.853	−0.622	0.943	0.943
	Chemical BERT	0.887	−1.017	0.932	0.932
	LegalBERT	0.887	−0.879	0.942	0.942

* Best results are shown in bold.

Table 5. The computing time and the environmental impact associated with the LMDFit and conventional benchmarking approaches for the AGNews data.

Task	LMDFit		Conventional	
	Time (s)	Emissions (g)	Time (s)	Emissions (g)
Fitness assessment	55.3	1.2	-	-
Clustering and Hiring	2.0	<0.1	-	-
Benchmarking	27,599.1	1851.8	97,620.6	6569.1
Total	27,656.4	1853.0	97,620.6	6569.1

5.3. Financial Phrase-Bank

Model fine-tuning for this single-label three-class classification task was carried out with 3876 sentences reserved for training and 970 sentences for validation in each fold. Tables 6 and 7 show results of the model initial selection and benchmarking experiments.

Table 6. Values of the LMDFit and conventional benchmarking metrics averaged in 5-fold cross-validation on the financial phrase-bank dataset (sorted by the cosine similarity mean value.

	Model	LMDFit Metrics		Conventional Metrics *	
		Mean	Skewness	F1 Micro	F1 Macro
more-fit	BERT	0.713	−0.164	0.840	0.825
	FinancialBERT	0.749	−0.129	0.841	0.824
	PharmBERT	0.752	−0.240	0.835	0.810
	Agriculture BERT	0.817	−0.162	**0.844**	**0.830**
less-fit	SciBERT	0.762	−0.825	0.840	0.823
	Chemical BERT	0.815	−0.490	0.729	0.592
	LegalBERT	0.881	−0.809	0.731	0.576

* Best results are shown in bold.

Table 7. The computing time and the environmental impact associated with the LMDFit and conventional benchmarking approaches for the financial phrase-bank data.

Task	LMDFit		Conventional	
	Time (s)	Emissions (g)	Time (s)	Emissions (g)
Fitness assessment	52.3	1.1	-	-
Clustering and Hiring	2.1	<0.1	-	-
Benchmarking	1431.2	63.4	2487.5	109.9
Total	1485.6	64.5	2487.5	109.9

5.4. Rheology Dataset

The single-label five-class classification task of the Rheology corpus was performed with 1612 documents used for training and 403 for validation in each fold. Tables 8 and 9 provide summaries of the model initial selection and benchmarking experiments with these data.

Table 8. Values of the LMDFit and conventional benchmarking metrics averaged in 5-fold cross-validation on the Rheology dataset (sorted by the cosine similarity mean value).

	Model	LMDFit Metrics		Conventional Metrics *	
		Mean	Skewness	F1 Micro	F1 Macro
more-fit	PharmBERT	0.798	−0.608	0.577	0.387
	Chemical BERT	0.831	−0.515	0.563	0.397
	Agriculture BERT	0.838	−0.520	**0.639**	**0.573**
less-fit	BERT	0.779	−0.668	0.539	0.358
	SciBERT	0.783	−0.768	0.589	0.491
	FinancialBERT	0.818	−0.766	0.543	0.357
	LegalBERT	0.902	−0.692	0.472	0.291

* Best results are shown in bold.

Table 9. The computing time and the environmental impact associated with the LMDFit and conventional benchmarking approaches for the Rheology data.

Task	LMDFit		Conventional	
	Time (s)	Emissions (g)	Time (s)	Emissions (g)
Fitness assessment	55.0	1.2	-	-
Clustering and Hiring	1.9	<0.1	-	-
Benchmarking	753.6	28.8	1784.6	68.0
Total	810.5	30.0	1784.6	68.0

5.5. Plant–Chemical Relationship Corpus

The benchmarking task for the plant–chemical relationship data is a single-label binary classification. The fine-tuning process was accomplished with 752 documents for training and 187 for validation in each fold. Tables 10 and 11 provide results of the model initial selection and benchmarking experiments.

Table 10. Values of the LMDFit and conventional benchmarking metrics averaged in 5-fold cross-validation on the plant–chemical relationship corpus (sorted by the cosine similarity mean value.

	Model	LMDFit Metrics		Conventional Metrics *	
		Mean	Skewness	F1 Micro	F1 Macro
more-fit	SciBERT	0.737	−0.425	0.816	0.812
	BERT	0.786	−0.508	0.784	0.781
	FinancialBERT	0.824	−0.342	0.756	0.754
	Agriculture BERT	0.827	−0.555	**0.840**	**0.837**
	Chemical BERT	0.854	−0.230	0.769	0.766
	LegalBERT	0.893	−0.544	0.707	0.697
less-fit	PharmBERT	0.784	−1.711	0.803	0.801

* Best results are shown in bold.

Table 11. The computing time and the environmental impact associated with the LMDFit and conventional benchmarking approaches for the plant–chemical relationship data.

Task	LMDFit		Conventional	
	Time (s)	Emissions (g)	Time (s)	Emissions (g)
Fitness assessment	54.7	1.2	-	-
Clustering and Hiring	2.0	<0.1	-	-
Benchmarking	1376.7	46.2	1614.9	54.3
Total	1433.4	47.4	1614.9	54.3

5.6. arXiv Documents

The benchmarking task for the arXiv data is multiple-label 10-class text classification. Model fine-tuning was conducted with 41,419 abstracts reserved for training and 10,355 for validation in each fold. Table 12 lists the averaged metric values obtained in the experiments, while Table 13 compares the two benchmarking approaches in terms of computational time and environmental impact.

Table 12. Values of the LMDFit and conventional benchmarking metrics averaged in 5-fold cross-validation on the arXiv collection (sorted by the cosine similarity mean value).

	Model	LMDFit Metrics		Conventional Metrics *	
		Mean	Skewness	F1 Micro	F1 Macro
more-fit	SciBERT	0.804	−0.494	0.807	**0.605**
	PharmBERT	0.847	−0.717	0.794	0.562
	FinancialBERT	0.866	−0.586	0.790	0.553
	Agriculture BERT	0.891	−0.465	**0.808**	0.600
	Chemical BERT	0.911	−0.652	0.779	0.522
	LegalBERT	0.932	−0.561	0.786	0.536
less-fit	BERT	0.835	−0.946	0.793	0.553

* Best results are shown in bold.

Table 13. The computing time and the environmental impact associated with the LMDFit and conventional benchmarking approaches for the arXiv data.

Task	LMDFit		Conventional	
	Time (s)	Emissions (g)	Time (s)	Emissions (g)
Fitness assessment	59.9	1.3	-	-
Clustering and Hiring	1.9	<0.1	-	-
Benchmarking	55,153.1	3612.0	64,373.6	4214.6
Total	55,214.9	3613.3	64,373.6	4214.6

5.7. ECtHR Cases

The benchmarking task for this data collection is multiple-label 33-class text classification. The fine-tuning process was performed with 8800 documents allocated for training and 2200 for validation in each fold. Tables 14 and 15 detail results obtained in the model initial selection and benchmarking experiments on these data.

Table 14. Values of the LMDFit and conventional benchmarking metrics averaged in 5-fold cross-validation on the ECtHR documents (sorted by the cosine similarity mean value).

	Model	LMDFit Metrics		Conventional Metrics *	
		Mean	Skewness	F1 Micro	F1 Macro
more-fit	BERT	0.816	−0.496	0.702	0.190
	SciBERT	0.854	−1.088	0.701	0.212
	PharmBERT	0.861	−0.787	0.702	0.200
	FinancialBERT	0.872	−1.126	0.691	0.202
	LegalBERT	0.885	−0.728	**0.717**	0.205
	Agriculture BERT	0.908	−1.274	0.700	**0.219**
less-fit	Chemical BERT	0.929	−4.083	0.662	0.163

* Best results are shown in bold.

Table 15. The computing time and the environmental impact associated with the LMDFit and conventional benchmarking approaches for the ECtHR data.

Task	LMDFit		Conventional	
	Time (s)	Emissions (g)	Time (s)	Emissions (g)
Fitness assessment	147.1	3.5	-	-
Clustering and Hiring	2.2	<0.1	-	-
Benchmarking	18,906.3	924.5	22,104.4	1080.0
Total	19,055.6	928.0	22,104.4	1080.0

5.8. Ohsumed Collection

The fine-tuning process for the multiple-label 23-class classification task of the Ohsumed data was conducted using 27,511 documents for training and the remainder for validation in each fold. Model selection statistics computed in the experiments with the data are listed in Tables 16 and 17.

Table 16. Values of the LMDFit and conventional benchmarking metrics averaged in 5-fold cross-validation on the Ohsumed document collection (sorted by the cosine similarity mean value).

	Model	LMDFit Metrics		Conventional Metrics *	
		Mean	Skewness	F1 Micro	F1 Macro
more-fit	SciBERT	0.699	−0.109	**0.783**	**0.746**
	Agriculture BERT	0.802	−0.124	0.776	0.739
less-fit	PharmBERT	0.764	−0.609	0.764	0.725
	BERT	0.792	−0.822	0.753	0.700
	FinancialBERT	0.824	−0.708	0.733	0.677
	LegalBERT	0.881	−0.834	0.732	0.663
	Chemical BERT	0.882	−0.654	0.715	0.652

* Best results are shown in bold.

Table 17. The computing time and the environmental impact associated with the LMDFit and conventional benchmarking approaches for the Ohsumed data.

Task	LMDFit		Conventional	
	Time (s)	Emissions (g)	Time (s)	Emissions (g)
Fitness assessment	61.1	1.4	-	-
Clustering and Hiring	2.1	<0.1	-	-
Benchmarking	9136.3	593.3	37,039.9	2431.6
Total	9199.5	594.7	37,039.9	2431.6

6. Discussion

It should be noted first that the proposed LMDFit approach demonstrated superior efficiency, compared to conventional benchmarking, in terms of computational time (a 36% decrease on average, see Table 18) and associated carbon footprint (a 37% reduction on average). Also, the best-performing models passed the model initial selection of the LMDFit benchmarking in all considered cases, which confirms the consistency of the proposed approach. As one can see from Table 18, the emission cuts achieved are strongly correlated with the reductions in computational time and depend on the number of models that did not pass the LMDFit test of model initial selection. While one might argue that in some cases, "less-fit" models would be "common-sense" recognized and manually disqualified at the stage of model requirement (e.g., the case of Chemical BERT for the ECtHR data), there are many other cases where this choice is not obvious or even impossible to make without testing the candidate models (e.g., the case of Agriculture BERT, which is the best performer on the ECtHR data).

Table 18. Benchmarking efficiency improvements with LMDFit.

Dataset	Number of Models Not Selected for Onboarding	Computational Time Decrease (%)	Emission Reduction (%)
Environmental claims	1	10.5	11.8
AGNews	5	71.7	71.8
Financial phrase-bank	3	40.3	41.2
Rheology	4	54.6	55.9
Plant–chemical relationship	1	11.2	12.6
arXiv	1	14.2	14.3
ECtHR	1	13.8	14.1
Ohsumed	5	75.2	75.5
Average	2.6	36.4	37.1

It should also be noted that while the LMDFit "more-fit" cluster of models always contained the best-performing PTLM, the statistics used for the clustering do not provide for a reliable ranking of the models performance-wise (see Table 19). This fact invalidates straightforward attempts to further refine the structure of this cluster by repeating the clustering procedure (or by increasing the number of clusters). Furthermore, when examining cases where all or some of the tested models show rather close results (e.g. the case of AGNews, see Table 4 for details), one should not "blindly" rely on the *k*-means analysis. A more flexible, manual decision-making procedure based on a visual analysis would be used to expand (or narrow down) the "more-fit" cluster. For instance, consider Figure 6, which depicts the results of the LMDFit model initial selection for the AGNews corpus. As SciBERT appears to be not too far, in terms of both cosine similarity mean and skewness, from PharmBERT, one might want to also examine the former model in the benchmarking experiments. Such decisions, however, should be made, taking into account the domain of the data and how it would correspond to the domain of the specific PTLM.

Table 19. Correlation of the LMDFit metrics with F1 micro and macro scores obtained in the experiments.

Dataset	F1 Micro				F1 Macro			
	Spearman		Pearson		Spearman		Pearson	
	Cosine Similarity		Cosine Similarity		Cosine Similarity		Cosine Similarity	
	Mean	Skewness	Mean	Skewness	Mean	Skewness	Mean	Skewness
Environmental claims	0.32	0.64	0.50	0.64	0.32	0.64	0.46	0.58
AGNews	0.61	0.86	0.47	0.63	0.67	0.88	0.47	0.63
Financial phrase-bank	0.32	0.71	0.75	0.55	0.43	0.68	0.77	0.56
Rheology	0.07	0.21	0.48	0.43	0.11	0.36	0.30	0.35
Plant–chemical relationship	0.57	0.39	0.69	0.23	0.57	0.39	0.69	0.23
arXiv	0.57	0.43	0.58	0.39	0.71	0.39	0.60	0.45
ECtHR	0.89	0.86	0.53	0.92	0.04	0.07	0.24	0.75
Ohsumed	0.89	0.75	0.88	0.76	0.89	0.75	0.89	0.79
Average	0.53	0.61	0.61	0.57	0.47	0.52	0.55	0.54

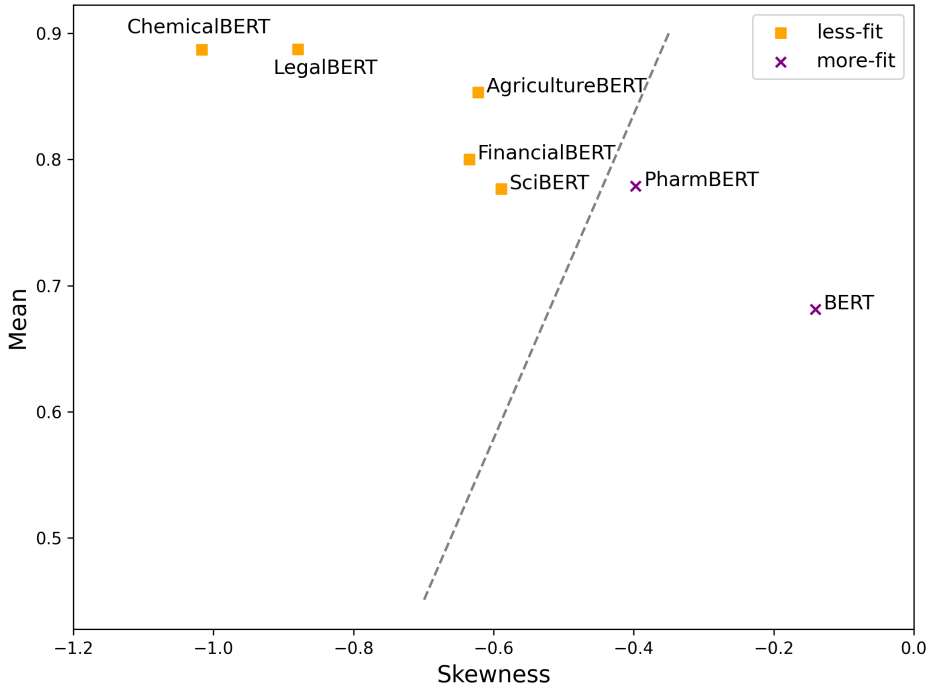

Figure 6. Model initial selection results for the AGNews classification task.

Unlike the existing approaches to predicting language model performance, LMDFit benchmarking does not endeavor to rank candidate-models (e.g., [10], which also evaluates PTLMs using cosine similarity) or make any specific extrapolations (e.g., [7,29]). Experiments conducted in this study demonstrate that model performance predictions made based solely on the cosine similarity statistics are noisy but still effective when analyzed in terms of the clusters they beget. On the other hand, to the authors' best knowledge, all existing approaches to forecasting language model performance require additional information, such as data labels [10] or a model's past application records [7,9]. Furthermore, the existing approaches do not provide specific guidelines on which (of the tested) models should be selected for benchmarking. Contrastingly, the LMDFit approach allows for automatically disqualifying less-fit models from the benchmarking process based on an analysis of nothing else but a relatively small sample of data. All this makes the proposed approach not only unique but also a better choice in practical settings, where researchers seldom have access to PTLM historical records or any additional information about the data to be processed.

Reviewing the model initial selection procedure, an important observation is that its results may be unreliable when the number of tested models is small. The authors would, therefore, recommend exercising an explorative rather than conservative strategy in deciding models for a specific classification task. The latter would hardly lead to a significant increase in emissions as the model initial selection overhead is close to negligible (typically, much less than 1% of the benchmarking emissions; see the corresponding data of Section 5).

A limitation of the presented study is that LMDFit benchmarking was examined exclusively for BERT-architecture PTLMs in a text classification context and with a single set of hyperparameters. While BERT language models are among the most used at present [78], in specialized domains and on downstream text classification tasks [79], additional research is needed to scrutinize the proposed approach in more diverse NLP scenarios. However, this is outside the scope of this particular paper.

It is understood that the CO_2 emission estimates reported in this paper are specific to the regional carbon intensity values of Japan, the country where the experiments were conducted. The estimates are based on the average values for the regional energy grid, as determined by the CodeCarbon tool. These estimates may not generalize to regions with different energy mixes and carbon intensities. Future studies could conduct sensitivity analyses across multiple regions to better understand the variability of emission estimates under different energy scenarios.

7. Conclusions

The goal of the presented study was to devise an efficient benchmarking approach for BERT language models recruited to perform downstream classification tasks. By analogy with personnel selection in an organization, it is proposed to incorporate model initial selection into the benchmarking process. LMDFit, the developed approach, first examines all candidate-models on a simple text similarity assessment task. Based on the results of this examination, only those models that are expected to perform better on the given data are selected for the full-scale performance evaluation. LMDFit benchmarking was tested on eight different datasets and with seven popular BERT language models and was found to work effectively and efficiently in all experiments. The proposed approach consistently selected the best-performing model for the given data, yet it was 36% faster and 37% greener (in terms of carbon footprints associated with the computations) than the conventional benchmarking, on average.

Thus, the presented study contributes to the development of Green AI by providing a set of tools (in the form of open-source code) and the benchmarking methodology for text classification with BERT language models. Also, the study demonstrates that the distribution of cosine similarity computed with word embedding vectors of texts sampled from a corpus can be used as a fitness metric for BERT language models deemed suitable for classification tasks on this corpus. Cosine similarity mean and skewness, the two statistics of the distribution, are shown to reflect the classification capability of the model intended for the given task.

The authors would like to conclude the paper by acknowledging the limitations of this study, particularly in the range of models considered and the theoretical verification of the model initial selection test. Future research is warranted to deal with these two issues, as well as to further reduce carbon footprints associated with the AI model benchmarking process.

Author Contributions: Conceptualization, methodology, software, validation, formal analysis, investigation, writing—original draft preparation, visualization, A.R.; writing—review and editing, M.K., U.S. and V.K.; supervision, V.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partly supported by the SDGs Global Leadership Program from the Japan International Cooperation Agency (JICA).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All code and data of this study are available online at https://github.com/Just108/LMD-Fit (accessed on 24 October 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Strubell, E.; Ganesh, A.; McCallum, A. Energy and policy considerations for deep learning in NLP. *arXiv* **2019**, arXiv:1906.02243.
2. Verdecchia, R.; Sallou, J.; Cruz, L. A systematic review of Green AI. *WIREs Data Min. Knowl. Discov.* **2023**, *13*, e1507. [CrossRef]
3. Schwartz, R.; Dodge, J.; Smith, N.A.; Etzioni, O. Green AI. *Commun. ACM* **2020**, *63*, 54–63. [CrossRef]
4. Li, B.; Qi, P.; Liu, B.; Di, S.; Liu, J.; Pei, J.; Yi, J.; Zhou, B. Trustworthy AI: From Principles to Practices. *ACM Comput. Surv.* **2023**, *55*, 1–46. [CrossRef]
5. Blagec, K.; Dorffner, G.; Moradi, M.; Samwald, M. A critical analysis of metrics used for measuring progress in artificial intelligence. *arXiv* **2021**, arXiv:2008.02577.
6. Klotz, A.C.; da Motta Veiga, S.P.; Buckley, M.R.; Gavin, M.B. The role of trustworthiness in recruitment and selection: A review and guide for future research. *J. Organ. Behav.* **2013**, *34*, S104–S119. [CrossRef]
7. Ahuja, K.; Dandapat, S.; Sitaram, S.; Choudhury, M. Beyond Static Models and Test Sets: Benchmarking the Potential of Pre-trained Models Across Tasks and Languages. *arXiv* **2022**, arXiv:2205.06356.
8. Ahuja, K.; Kumar, S.; Dandapat, S.; Choudhury, M. Multi task learning for zero shot performance prediction of multilingual models. *arXiv* **2022**, arXiv:2205.06130.
9. Xia, M.; Anastasopoulos, A.; Xu, R.; Yang, Y.; Neubig, G. Predicting performance for natural language processing tasks. *arXiv* **2020**, arXiv:2005.00870.
10. Kadiķis, E.; Vaibhav, S.; Klinger, R. Embarrassingly simple performance prediction for abductive natural language inference. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Langua, Online, 10–15 July 2022. [CrossRef]
11. Li, Q.; Peng, H.; Li, J.; Xia, C.; Yang, R.; Sun, L.; Yu, P.S.; He, L. A Survey on Text Classification: From Traditional to Deep Learning. *ACM Trans. Intell. Syst. Technol.* **2022**, *13*, 1–41. [CrossRef]
12. Altınel, B.; Ganiz, M.C. Semantic text classification: A survey of past and recent advances. *Inf. Process. Manag.* **2018**, *54*, 1129–1153. [CrossRef]
13. Garrido-Merchan, E.C.; Gozalo-Brizuela, R.; Gonzalez-Carvajal, S. Comparing BERT against traditional machine learning models in text classification. *J. Comput. Cogn. Eng.* **2023**, *2*, 352–356. [CrossRef]
14. Ferro, M.; Silva, G.D.; de Paula, F.B.; Vieira, V.; Schulze, B. Towards a sustainable artificial intelligence: A case study of energy efficiency in decision tree algorithms. *Concurr. Comput. Pract. Exp.* **2023**, *35*, e6815. [CrossRef]
15. Gutiérrez, M.; Moraga, M.Á.; García, F. Analysing the energy impact of different optimisations for machine learning models. In Proceedings of the 2022 International Conference on ICT for Sustainability (ICT4S), Plovdiv, Bulgaria, 13–17 June 2022; IEEE: New York, NY, USA, 2022; pp. 46–52. [CrossRef]
16. Gurumurthy, A.; Kodali, R. Benchmarking the Benchmarking Models. *Benchmarking Int. J.* **2008**, *15*, 257–291. [CrossRef]
17. Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S.R. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *arXiv* **2018**, arXiv:1804.07461.
18. Wang, A.; Pruksachatkun, Y.; Nangia, N.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S. Superglue: A stickier benchmark for general-purpose language understanding systems. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 3266–3280.
19. Liang, Y.; Duan, N.; Gong, Y.; Wu, N.; Guo, F.; Qi, W.; Gong, M.; Shou, L.; Jiang, D.; Cao, G.; et al. XGLUE: A new benchmark dataset for cross-lingual pre-training, understanding and generation. *arXiv* **2020**, arXiv:2004.01401.
20. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
21. Lundgard, A. Measuring justice in machine learning. *arXiv* **2020**, arXiv:2009.10050.

22. Caton, S.; Haas, C. Fairness in machine learning: A survey. *ACM Comput. Surv.* **2020**, *56*, 1–38. [CrossRef]
23. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 6000–6010.
24. Casola, S.; Lauriola, I.; Lavelli, A. Pre-trained transformers: An empirical comparison. *Mach. Learn. Appl.* **2022**, *9*, 100334. [CrossRef]
25. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
26. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv* **2019**, arXiv:1910.01108.
27. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.R.; Le, Q.V. Xlnet: Generalized autoregressive pretraining for language understanding. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 5753–5763.
28. Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; Soricut, R. Albert: A lite bert for self-supervised learning of language representations. *arXiv* **2019**, arXiv:1909.11942.
29. Ye, Z.; Liu, P.; Fu, J.; Neubig, G. Towards more fine-grained and reliable NLP performance prediction. *arXiv* **2021**, arXiv:2102.05486.
30. Boughorbel, S.; Jarray, F.; El-Anbari, M. Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric. *PLoS ONE* **2017**, *12*, e0177678. [CrossRef] [PubMed]
31. Jiang, Y.; Neyshabur, B.; Mobahi, H.; Krishnan, D.; Bengio, S. Fantastic Generalization Measures and Where to Find Them. *arXiv* **2019**, arXiv:1912.02178.
32. Dziugaite, G.K.; Drouin, A.; Neal, B.; Rajkumar, N.; Caballero, E.; Wang, L.; Mitliagkas, I.; Roy, D.M. In search of robust measures of generalization. In Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20, Red Hook, NY, USA, 6–12 December 2020.
33. Martin, C.H.; Peng, T.; Mahoney, M.W. Predicting trends in the quality of state-of-the-art neural networks without access to training or testing data. *Nat. Commun.* **2021**, *12*, 4122. [CrossRef]
34. Yang, Y.; Theisen, R.; Hodgkinson, L.; Gonzalez, J.E.; Ramchandran, K.; Martin, C.H.; Mahoney, M.W. Test Accuracy vs. Generalization Gap: Model Selection in NLP without Accessing Training or Testing Data. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Long Beach, CA, USA, 6–10 August 2023; ACM: New York, NY, USA, 2023; pp. 3011–3021. [CrossRef]
35. Beltagy, I.; Lo, K.; Cohan, A. SciBERT: A pretrained language model for scientific text. *arXiv* **2019**, arXiv:1903.10676.
36. Chalkidis, I.; Fergadiotis, M.; Malakasiotis, P.; Aletras, N.; Androutsopoulos, I. LEGAL-BERT: The muppets straight out of law school. *arXiv* **2020**, arXiv:2010.02559.
37. Han, X.; Zhang, Z.; Ding, N.; Gu, Y.; Liu, X.; Huo, Y.; Qiu, J.; Yao, Y.; Zhang, A.; Zhang, L.; et al. Pre-trained models: Past, present and future. *AI Open* **2021**, *2*, 225–250. [CrossRef]
38. Lee, J.; Yoon, W.; Kim, S.; Kim, D.; Kim, S.; So, C.H.; Kang, J. BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* **2020**, *36*, 1234–1240. [CrossRef] [PubMed]
39. Hazourli, A. FinancialBERT—A Pretrained Language Model for Financial Text Mining. 2022. Available online: https://huggingface.co/ahmedrachid/FinancialBERT (accessed on 23 October 2024). [CrossRef]
40. ValizadehAslani, T.; Shi, Y.; Ren, P.; Wang, J.; Zhang, Y.; Hu, M.; Zhao, L.; Liang, H. PharmBERT: A domain-specific BERT model for drug labels. *Briefings Bioinform.* **2023**, *24*, bbad226. [CrossRef] [PubMed]
41. Stammbach, D.; Webersinke, N.; Bingler, J.A.; Kraus, M.; Leippold, M. A Dataset for Detecting Real-World Environmental Claims. *arXiv* **2022**, arXiv:2209.00507. [CrossRef]
42. Webersinke, N.; Kraus, M.; Bingler, J.A.; Leippold, M. Climatebert: A pretrained language model for climate-related text. *arXiv* **2021**, arXiv:2110.12010. [CrossRef]
43. Schimanski, T.; Bingler, J.; Hyslop, C.; Kraus, M.; Leippold, M. Climatebert-netzero: Detecting and assessing net zero and reduction targets. *arXiv* **2023**, arXiv:2310.08096.
44. Zhang, X.; Zhao, J.; LeCun, Y. Character-level convolutional networks for text classification. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 649–657.
45. Li, Z.; Xu, J.; Zeng, J.; Li, L.; Zheng, X.; Zhang, Q.; Chang, K.W.; Hsieh, C.J. Searching for an effective defender: Benchmarking defense against adversarial word substitution. *arXiv* **2021**, arXiv:2108.12777.
46. Xiong, Y.; Feng, Y.; Wu, H.; Kamigaito, H.; Okumura, M. Fusing label embedding into bert: An efficient improvement for text classification. In Proceedings of the Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Online, 1–6 August 2021; pp. 1743–1750.
47. Malo, P.; Sinha, A.; Korhonen, P.; Wallenius, J.; Takala, P. Good debt or bad debt: Detecting semantic orientations in economic texts. *J. Assoc. Inf. Sci. Technol.* **2014**, *65*, 782–796. [CrossRef]

48. Soong, G.H.; Tan, C.C. Sentiment Analysis on 10-K Financial Reports using Machine Learning Approaches. In Proceedings of the 2021 IEEE 11th International Conference on System Engineering and Technology (ICSET), Shah Alam, Malaysia, 6 November 2021; IEEE: New York, NY, USA, 2021; pp. 124–129.
49. Leippold, M. Sentiment spin: Attacking financial sentiment with GPT-3. *Financ. Res. Lett.* **2023**, *55*, 103957. [CrossRef]
50. Kim, Y.; Ko, H.; Lee, J.; Heo, H.Y.; Yang, J.; Lee, S.; Lee, K.h. Chemical Language Understanding Benchmark. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*; Sitaram, S., Beigman Klebanov, B., Williams, J.D., Eds.; Association for Computational Linguistic: Toronto, ON, Canada, 2023; pp. 404–411. [CrossRef]
51. Cho, H.; Kim, B.; Choi, W.; Lee, D.; Lee, H. Plant phenotype relationship corpus for biomedical relationships between plants and phenotypes. *Sci. Data* **2022**, *9*, 235. [CrossRef]
52. Choi, W.; Kim, B.; Cho, H.; Lee, D.; Lee, H. A corpus for plant-chemical relationships in the biomedical domain. *BMC Bioinform.* **2016**, *17*, 1–15. [CrossRef]
53. Scharpf, P.; Schubotz, M.; Youssef, A.; Hamborg, F.; Meuschke, N.; Gipp, B. Classification and clustering of arxiv documents, sections, and abstracts, comparing encodings of natural and mathematical language. In Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020, Online, 1–5 August 2020; pp. 137–146.
54. Patadia, D.; Kejriwal, S.; Mehta, P.; Joshi, A.R. Zero-shot approach for news and scholarly article classification. In Proceedings of the 2021 International Conference on Advances in Computing, Communication, and Control (ICAC3), Mumbai, India, 3–4 December 2021; IEEE: New York, NY, USA, 2021; pp. 1–5.
55. Chalkidis, I.; Jana, A.; Hartung, D.; Bommarito, M.; Androutsopoulos, I.; Katz, D.M.; Aletras, N. LexGLUE: A benchmark dataset for legal language understanding in English. *arXiv* **2021**, arXiv:2110.00976. [CrossRef]
56. Peng, Y.; Wu, W.; Ren, J.; Yu, X. Novel GCN Model Using Dense Connection and Attention Mechanism for Text Classification. *Neural Process. Lett.* **2024**, *56*, 1–17. [CrossRef]
57. Burkhardt, S.; Kramer, S. Online multi-label dependency topic models for text classification. *Mach. Learn.* **2018**, *107*, 859–886. [CrossRef]
58. Schilter, O.; Schwaller, P.; Laino, T. Balancing computational chemistry's potential with its environmental impact. *Green Chem.* **2024**, *26*, 8669–8679. [CrossRef]
59. Martínez, F.S.; Parada, R.; Casas-Roma, J. CO2 impact on convolutional network model training for autonomous driving through behavioral cloning. *Adv. Eng. Inform.* **2023**, *56*, 101968. [CrossRef]
60. Becker, G.S. Human capital and the economy. *Proc. Am. Philos. Soc.* **1992**, *136*, 85–92.
61. Morley, M.J. Person-organization fit. *J. Manag. Psychol.* **2007**, *22*, 109–117. [CrossRef]
62. Edwards, J.R. *Person-Job Fit: A Conceptual Integration, Literature Review, and Methodological Critique*; John Wiley & Sons: Hoboken, NJ, USA, 1991.
63. Nafukho, F.M.; Hairston, N.; Brooks, K. Human capital theory: Implications for human resource development. *Hum. Resour. Dev. Int.* **2004**, *7*, 545–551. [CrossRef]
64. Harris, Z. *Distributional Structure*; Taylor & Francis Group: Abingdon, UK, 1954.
65. Firth, J.R. A synopsis of linguistic theory, 1930 $\pm$ 1955' Studies in Linguistic Analysis. In *Special Volume of the Philological Society*; Blackwell: Oxford, UK, 1957.
66. Turney, P.D.; Pantel, P. From frequency to meaning: Vector space models of semantics. *J. Artif. Intell. Res.* **2010**, *37*, 141–188. [CrossRef]
67. Hill, F.; Reichart, R.; Korhonen, A. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Comput. Linguist.* **2015**, *41*, 665–695. [CrossRef]
68. Gao, T.; Yao, X.; Chen, D. Simcse: Simple contrastive learning of sentence embeddings. *arXiv* **2021**, arXiv:2104.08821.
69. Aggarwal, C.C.; Hinneburg, A.; Keim, D.A. On the surprising behavior of distance metrics in high dimensional space. In Proceedings of the Database Theory—ICDT 2001: 8th International Conference, London, UK, 4–6 January 2001; proceedings 8; Springer: Berlin/Heidelberg, Germany, 2001; pp. 420–434.
70. Huang, A. Similarity measures for text document clustering. In Proceedings of the Sixth New Zealand Computer Science Research Student Conference (NZCSRSC2008), Christchurch, New Zealand, 14–18 April 2008; Volume 4, pp. 9–56.
71. Spruill, M. Asymptotic distribution of coordinates on high dimensional spheres. *Electron. Commun. Probab.* **2007**, *12*, 234–247. [CrossRef]
72. Paukkeri, M.S.; Kivimäki, I.; Tirunagari, S.; Oja, E.; Honkela, T. Effect of dimensionality reduction on different distance measures in document clustering. In Proceedings of the Neural Information Processing: 18th International Conference, ICONIP 2011, Shanghai, China, 13–17 November 2011; Proceedings, Part III 18; Springer: Berlin/Heidelberg, Germany, 2011; pp. 167–176.
73. Wijewickrema, M.; Petras, V.; Dias, N. Selecting a text similarity measure for a content-based recommender system: A comparison in two corpora. *Electron. Libr.* **2019**, *37*, 506–527. [CrossRef]
74. Parsons, V.L. Stratified Sampling. In *Wiley StatsRef: Statistics Reference Online*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2017; pp. 1–11. [CrossRef]

75. Doane, D.P.; Seward, L.E. Measuring skewness: A forgotten statistic? *J. Stat. Educ.* **2011**, *19*, 2. [CrossRef]
76. Lloyd, S. Least squares quantization in PCM. *IEEE Trans. Inf. Theory* **1982**, *28*, 129–137. [CrossRef]
77. Yu, H.; Gao, C.; Li, X.; Zhang, L. Ancient Chinese Poetry Collation Based on BERT. *Procedia Comput. Sci.* **2024**, *242*, 1171–1178. [CrossRef]
78. Raiaan, M.A.K.; Mukta, M.S.H.; Fatema, K.; Fahad, N.M.; Sakib, S.; Mim, M.M.J.; Ahmad, J.; Ali, M.E.; Azam, S. A review on large Language Models: Architectures, applications, taxonomies, open issues and challenges. *IEEE Access* **2024**, *12*, 26839–26874. [CrossRef]
79. Gasparetto, A.; Marcuzzo, M.; Zangari, A.; Albarelli, A. A survey on text classification algorithms: From text to predictions. *Information* **2022**, *13*, 83. [CrossRef]