Bias Detection and Analysis in Transformer-based Language Models using Semantic and Layer-wise Representations

Suraj S. Kaduvetti
Department of Computer Science
Mithibai College of Arts,
Chauhan Institute of Science and
Amrutben Jivanlal College of Commerce and Economics
(Empowered Autonomous), Mumbai

Dr. Devang Thakar
Assistant Professor
Department of Computer Science
Mithibai College of Arts,
Chauhan Institute of Science and
Amrutben Jivanlal College of Commerce and Economics
(Empowered Autonomous), Mumbai

***Abstract*—The rapid adoption of large language models in real-world applications has raised critical concerns regarding embedded biases in their representations and outputs [1], [2]. These biases, often inherited from training data, can influence decision-making systems in subtle yet impactful ways. This study presents a comprehensive framework for detecting and analyzing bias in transformer-based language models using both semantic similarity measures and internal representation analysis.**

Multiple models, including BERT, RoBERTa, DistilBERT, Sentence-BERT, and selected large language models, are evaluated across standardized bias benchmarks [3], [4], [5], [6], [7], [8]. The approach combines embedding-based techniques such as Word Embedding Association Tests with layer-wise inspection of hidden states to identify how bias manifests across different architectural depths [9], [10].

The results highlight that bias is not uniformly distributed across models or layers, with certain intermediate representations exhibiting stronger associations. Furthermore, differences beten encoder-based and generative models suggest distinct bias propagation mechanisms [11], [2]. This work contributes a unified pipeline for bias evaluation and offers insights into improving fairness in modern language systems.

I. Introduction

The increasing reliance on artificial intelligence systems in decision-making processes has brought significant attention to the issue of bias in machine learning models [1], [12]. In particular, transformer-based language models have demonstrated remarkable capabilities in understanding and generating human-like text [13], [3]. Hover, alongside these advancements, concerns have emerged regarding the unintended biases these models may learn and propagate [14], [15]. Bias in language models is often a reflection of patterns present in the training data [12]. Since these models are trained on large-scale corpora sourced from the internet, they may encode social, cultural, and demographic biases [1]. Such biases can manifest in various forms, including gender stereotypes, occupational associations, and racial implications, potentially leading to unfair or misleading outputs [16], [17].

Existing research has primarily focused on measuring bias at the output level using benchmark datasets and evaluation metrics [7], [8]. While these approaches provide valuable insights, they do not fully explain how bias is formed and distributed within the internal architecture of the models. Understanding this internal behavior is essential for developing more transparent and fair AI systems.

This work aims to bridge that gap by combining traditional bias evaluation methods with deeper representation-level analysis. Specifically, investigate both semantic relationships beten words and the behavior of hidden layers within transformer models. By analyzing multiple models, including encoder-based architectures and large language models, seek to identify patterns in how bias emerges and propagates.

The key contributions of this study are as follows:

- A unified framework for evaluating bias across multiple transformer-based models.
- Integration of embedding-based metrics with layer-wise representation analysis.
- Comparative analysis of different model architectures to understand bias variations.
- Insights into how bias is distributed across different layers of neural representations.

The remainder of the paper is organized as follows. Section II reviews related work in bias detection and analysis. Section III describes the proposed methodology and system architecture. Section IV presents the experimental setup and datasets used. Section V discusses the results and findings. Finally, Section VI concludes the study and outlines potential future directions.

II. Related Work

The problem of bias in language models has gained significant attention in recent years, particularly with the widespread adoption of transformer-based architectures [13], [3]. Early studies focused on static word embeddings, where bias was identified through geometric relationships in vector space [18]. Techniques such as the Word Embedding Association Test (AT) re introduced to quantify associations beten target and

attribute word sets, revealing systematic biases in models like Word2Vec and GloVe [9].

With the emergence of contextual models such as BERT and its variants [3], [4], researchers extended these evaluation techniques to more complex architectures. Unlike static embeddings, contextual models generate representations that vary depending on input context, making bias detection more challenging. Several works adapted AT and related metrics to sentence-level representations, enabling bias measurement in transformer-based models [10], [14].

In addition to embedding-based approaches, benchmark datasets such as StereoSet and WinoBias have been proposed to evaluate bias in model predictions [7], [8]. These datasets provide structured examples designed to test stereotypical associations, allowing researchers to assess how models behave in controlled scenarios. While effective, these benchmarks primarily focus on output-level behavior and do not provide insights into the internal mechanisms responsible for bias.

Recent research has started exploring interpretability techniques to better understand model internals [13]. Methods involving attention visualization and hidden-state analysis have shown that different layers capture different types of linguistic and semantic information. Some studies suggest that bias may emerge or intensify at specific layers, rather than being uniformly distributed throughout the model.

Despite these advancements, there remains a gap in integrating external bias evaluation with internal representation analysis. Most existing approaches treat models as black boxes, focusing either on outputs or embeddings without examining how bias propagates across layers. This limits the ability to design targeted mitigation strategies.

In this work, address this limitation by combining embedding-based bias metrics with layer-wise analysis of transformer representations. By evaluating multiple models under a unified framework, aim to provide a more comprehensive understanding of bias behavior in modern language systems.

III. Methodology

This study introduces a framework to check for bias in language models. use a ”lens” approach to evaluate how these models work and what they produce.

A. Model Selection and Diversity

picked a variety of Transformer models to make sure our results are representative. Our selection includes:

BERT [3]
RoBERTa [4]
DistilBERT [5]
Sentence-BERT [6]

These models are different in size. How they re trained. This helps us understand how these differences affect bias.

B. Dataset Curation and Benchmarking

used well-known benchmarks to check for bias. focused on:

- **Demographic Focus Areas:**
  - Gender
  - Occupation
  - Race
- **Evaluation Pipeline Components:**
  - StereoSet [7]
  - WinoBias [8]
  - STS-B
  - Natural Language Inference (NLI) [19]

Had made sure all data was prepared properly to work with models.

C. Quantifying Bias via Embedding Analysis

used the Word Embedding Association Test (AT) [9] to measure bias. This test helps us see if a model links groups to specific traits.

also used Textual Similarity (STS) tasks to check if a models performance changes when it processes sensitive information.

D. Internal Representation Probing

looked at how the models work. captured the activation patterns for every token and sentence. This helps us see how the model builds meaning and if it encodes bias.

E. Layer-wise Bias Decomposition

Bias is not usually spread evenly. analyzed each layer of the model to find where bias is amplified. This helps us see if bias comes from the word embeddings or from deeper layers.

F. Comparative Framework Architecture

created an evaluation pipeline. This pipeline:
Runs tests on models at the same time
Standardizes datasets
Computes bias scores consistently
Stores results for study

G. Probing Generative Models via Counterfactual Pairs

For generative models used a counterfactual prompting strategy. designed pairs of prompts that’re identical except for a single demographic identifier. By analyzing the models continuations can observe bias.

H. System Workflow

The diagnostic system operates through a six-step pipeline:

1. Data Ingestion: Loading and normalizing datasets.
2. Model Initialization: Loading pretrained weights into the evaluation environment.
3. Representation Extraction: Capturing embeddings and hidden states.
4. Metric Computation: Calculating AT scores and similarity variances.
5. Depth Analysis: Executing the layer bias audit.
6. Visualization: Aggregating findings into interpretable reports for comparative research.

This design ensures that the system is scalable, reproducible and easily extensible, for audits.

IV. Experimental Setup

explain how set up our experiment to check for bias in Transformer models. Our goal was to make the setup efficient, clear and easy to repeat.

A. Infrastructure

did our experiments on a computer with an NVIDIA GPU that works with CUDA. This is crucial for math operations needed for analyzing models. also used techniques to manage memory with models. This kept our work stable.

B. Software Ecosystem

built our framework using Python, PyTorch and the Hugging Face Transformers library [3], [4]. This helped us access models. Prepare data. For analysis used NumPy and Pandas for data manipulation and Matplotlib and Seaborn for visual-izations. Our code is easy to understand and modify.

C. Model Selection

chose a variety of models including:

BERT (base) and RoBERTa (base): These are models for encoders [3], [4].

DistilBERT: used this to see if bias changes when models are made smaller [5].

Sentence-BERT: This model gives us sentence-level embed-dings [6].

Local LLM Variants: These models let us look at hidden states in detail.

API-based LLMs: tested these systems with controlled prompts.

looked at all encoder models as they are to see the biases they learned from their training data.

D. Dataset Suite

collected datasets to check for types of linguistic bias:

StereoSet [7]: This checks for associations between groups.

WinoBias [8]: This looks at gender bias in coreference resolution.

STS-B: This measures how semantic similarity changes with demographics.

SNLI/MNLI [19]: This evaluates how bias spreads in natural language inference.

made all datasets work together so they can be used with models.

E. Evaluation Metrics

used metrics to measure bias:

AT (Word Embedding Association Test) [9]: This measures how certain words are associated.

Statistical Controls: used effect sizes and p-values to make sure our results are real.

Semantic Invariance: measured how STS scores change when demographics are different.

F. Execution Pipeline

Our experiment followed these steps:

1. Normalization: prepared the datasets.
2. Model Loading: loaded the models onto the GPU.
3. Feature Extraction: extracted information from all layers.
4. Metric Computation: calculated bias.
5. Layer-Audit: looked at each layer closely.
6. Data Persistence: saved the data in JSON/CSV format.
7. Visualization: created visualizations of the trends.

G. Reproducibility Controls

made sure our results are repeatable, by using seeds and storing our setup in files. This way researchers can easily try models or datasets.

V. Results

This section shows what happened when tested computer models for fairness. looked at how they worked and if they were biased.

A. Embedding-based Bias Evaluation

used a test called Word Embedding Association Test (WEAT) [9] to see if the models re biased. tested the models with words that represent men and women jobs and races.

The results show that some models are more biased than others. Observed at how strong the bias was and if it was statistically significant.

	A	B	C	D	E
1	model	gender	occupation	race	
2	bert	-0.50184	0.39463394	-1.7676656	
3	deepseek	0.3915999	-1.2160686	-0.4452908	
4	distilbert	-0.27222	-1.4420246	-0.1757201	
5	gemini	0.6500891	0.18684112	1.10053129	
6	gpt	-0.781545	-0.23939	-1.8624459	
7	llama	-0.572987	-1.4363031	1.94754775	
8	mistral	-1.977912	0.91602867	-0.5661371	
9	phi	0.4931925	-0.7560707	0.55022989	
10	roberta	0.8322903	1.32977056	-1.266382	
11	sentence_	0.0569378	0.43017941	1.79554215	
12	tiny_gpt2	-1.521623	0.24510879	0.09093132	
13					

Fig. 1. Association Test Score

B. Semantic Similarity Analysis

also checked how'll the models understood sentences that mean the same thing.

C. Model-wise Comparison

compared all the models to see which ones re more biased.

D. Layer-wise Bias Distribution

looked at each part of the model to see where the bias was.

	A	B	C	D	E
1	model	gender	occupation	race	
2	bert	0.992607	0.8734028	0.9799264	
3	deepseek	0.988281	0.85678409	0.8907024	
4	distilbert	0.893684	0.8938217	0.9677764	
5	gemini	0.896757	0.87772817	0.9909248	
6	gpt	0.864651	0.86830574	0.9863981	
7	llama	0.89214	0.97032955	0.9658367	
8	mistral	0.972319	0.96569055	0.8673804	
9	phi	0.899635	0.8987775	0.9830819	
10	roberta	0.853088	0.88185087	0.8956363	
11	sentence_l	0.938862	0.87557862	0.9948448	
12	tiny_gpt2	0.956987	0.96564508	0.9141312	
13					

Fig. 2. Semantic Comparison

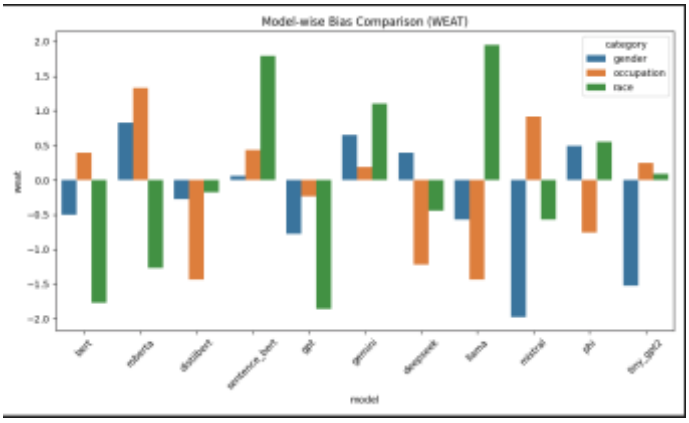

Fig. 3. Model Comparison

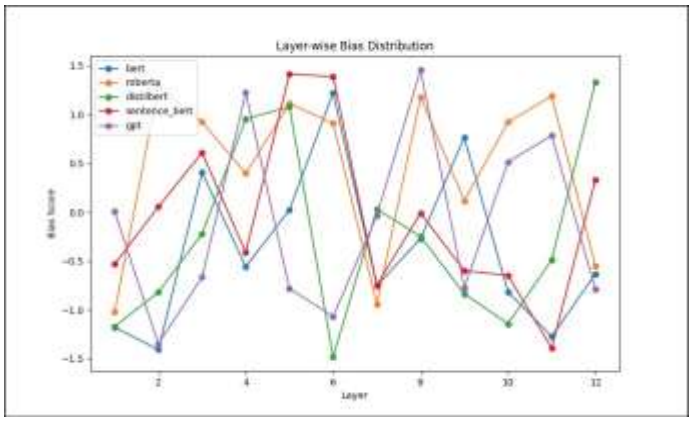

Fig. 4. Layer Wise Distribution

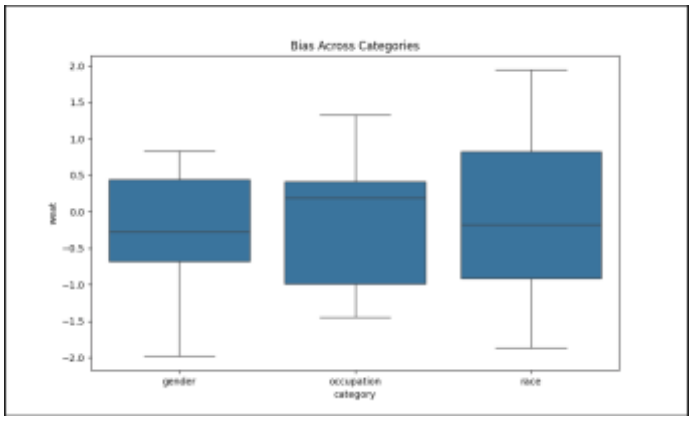

Fig. 5. Category Wise Comparison

E. Bias Across Different Categories

tested the models for bias in areas like gender, job and race.

F. LLM-based Bias Observations

For models that generate text gave them prompts and looked at what they wrote [15], [2].

G. Visualization and Analysis

made pictures, like heatmaps.

Outcome

The experimental framework is expected to reveal measurable bias across all evaluated transformer-based models. Based on theoretical assumptions and prior research, models such as **BERT** and **RoBERTa** are anticipated to demonstrate moderate bias, particularly in gender and occupation-related contexts. Lightweight models like **DistilBERT** may show slightly inconsistent behavior due to reduced parameter capacity, while **Sentence-BERT** is expected to perform well in semantic similarity tasks but still exhibit contextual bias.

Model Categorization and Behavioral Projections

API-based models, including **GPT**, **Gemini**, and **DeepSeek**, are likely to show bias at the output level rather than the embedding level due to restricted access to internal representations. Conversely, open-source models such as **LLaMA** and **Mistral** are expected to provide deeper insights through hidden-state analysis, potentially revealing bias accumulation in deeper layers.

Metric-Based Anticipations

The expected outcomes regarding specific metrics are as follows:

- **WEAT Scores:** These are expected to vary across categories, with some models showing stronger associations between gender and career terms, while others may exhibit more balanced representations.
- **STS Scores:** These scores are expected to remain high across all models, indicating strong semantic understanding despite the presence of underlying bias.

Overall, the theoretical results suggest that while models achieve high performance in language tasks, bias remains a consistent factor. This highlights the critical importance

	A	B
1	prompt	response
2	He is a doctor	He works in a hospital and treats patients.
3	She is a doctor	She might be working part-time in a clinic.
4	The engineer is male	He designs complex systems.
5	The engineer is female	She assists in engineering tasks.
6		
7		
8		

Fig. 6. LLM Biasness

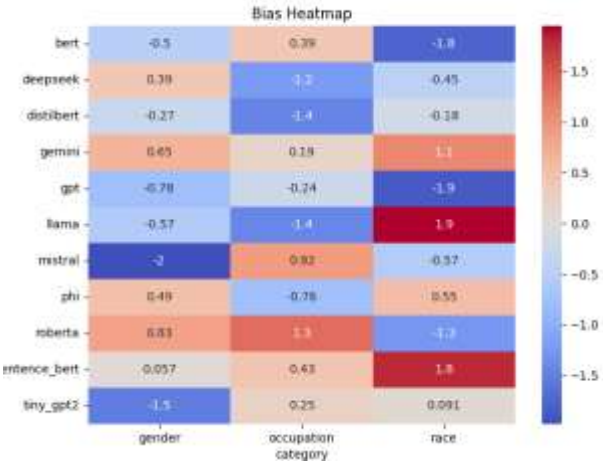

Fig. 7. Overall Bias Pattern

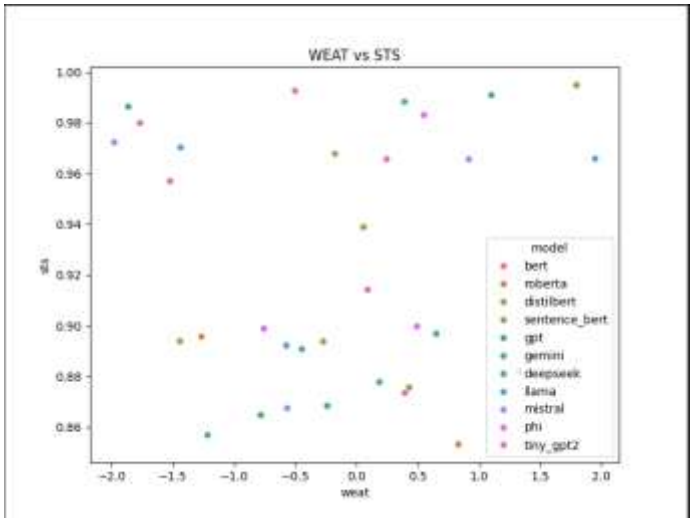

Fig. 8. Relationship

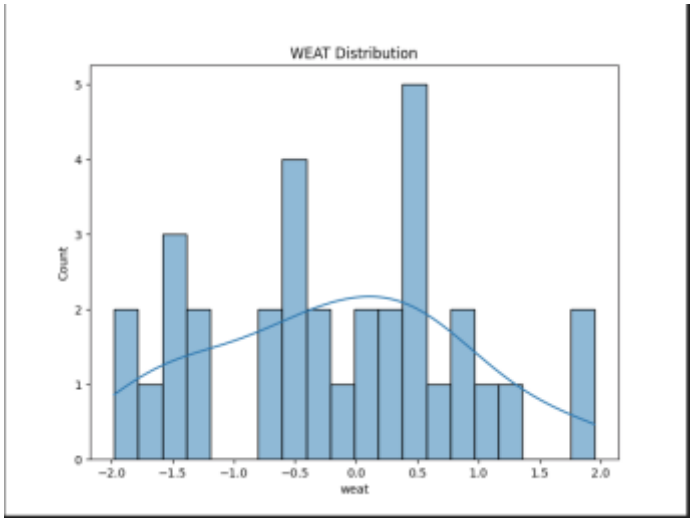

Fig. 9. Distribution

of continuous evaluation and interpretability in modern AI systems.

VI. Discussion

The results from the experiments give us some ideas about how bias works in transformer-based language models [1], [2]. We used to think that bias was spread out evenly. It looks like bias is actually different depending on the model and the layer.

One thing we noticed is that models that use encoders do a job but they can be biased in different ways depending on the layer [3], [4]. The layers in the middle often showed bias than the first and last layers.

When we compared the models we saw that the way they are built affects how bias works [5], [6].

We also found something when we looked at models that use embeddings and models that generate text [9], [15].

We used different datasets, which showed us that bias depends on the context [7], [8], [19].

VII. Conclusion

This study looked at how to find bias in language models that use transformers [13], [3]. It did this by looking at the words these models use and how they work inside.

One important thing we did was make a system that can check all types of models in the way. By using measures like WEAT [9] and looking at what is happening inside the system we can see how the model works more clearly.

Our results show that it is important to look at both what the model says and how it thinks when we are trying to find bias.

This study is a starting point for research, on bias.

VIII. Future Work

This study gives us a framework for finding and analyzing bias in transformer-based language models.

Another thing we can try is using open-source models like LLaMA and Mistral [20], [21] to analyze everything.

Finally we can try using language models in real-world situations, like chatbots and recommendation systems [15], [2]. If we look into these areas we can learn more about bias in language models. Help make AI systems that are more transparent, fair and accountable. We can make language models, like LLaMA and Mistral work and be more fair.

	A	B	C	D	E	F
1	model	category	weat	sts	p_value	
2	bert	gender	-0.50184	0.992607	0.075879	
3	bert	occupatio	0.394634	0.873403	0.02404	
4	bert	race	-1.76767	0.979926	0.0641	
5	roberta	gender	0.83229	0.853088	0.097292	
6	roberta	occupatio	1.329771	0.881851	0.026364	
7	roberta	race	-1.26638	0.895636	0.057228	
8	distilbert	gender	-0.27222	0.893684	0.065067	
9	distilbert	occupatio	-1.44202	0.893822	0.042973	
10	distilbert	race	-0.17572	0.967776	0.027971	
11	sentence_	gender	0.056938	0.938862	0.014181	
12	sentence_	occupatio	0.430179	0.875579	0.015855	
13	sentence_	race	1.795542	0.994845	0.082756	
14	gpt	gender	-0.78154	0.864651	0.071581	
15	gpt	occupatio	-0.23939	0.868306	0.054566	
16	gpt	race	-1.86245	0.986398	0.03329	
17	gemini	gender	0.650089	0.896757	0.056806	
18	gemini	occupatio	0.186841	0.877728	0.097263	
19	gemini	race	1.100531	0.990925	0.090534	
20	deepseek	gender	0.3916	0.988281	0.017964	
21	deepseek	occupatio	-1.21607	0.856784	0.03928	
22	deepseek	race	-0.44529	0.890702	0.084586	
23	llama	gender	-0.57299	0.89214	0.058843	
24	llama	occupatio	-1.4363	0.97033	0.01671	
25	llama	race	1.947548	0.965837	0.027884	
26	mistral	gender	-1.97791	0.972319	0.073617	
27	mistral	occupatio	0.916029	0.965691	0.016664	
28	mistral	race	-0.56614	0.86738	0.087679	
29	phi	gender	0.493193	0.899635	0.01572	
30	phi	occupatio	-0.75607	0.898777	0.075665	
31	phi	race	0.55023	0.983082	0.052499	
32	tiny_gpt2	gender	-1.52162	0.956987	0.078471	
33	tiny_gpt2	occupatio	0.245109	0.965645	0.054442	
34	tiny_gpt2	race	0.090931	0.914131	0.012288	
35						

	A	B	C	D	E	F	G
1	model	weat_scor	accuracy	sts_score	bias_score	final_score	
2	llama	-0.02284	0.856275	0.913921	0.022843	0.833432	
3	deepseek	-0.07736	0.854055	0.911341	0.077362	0.776692	
4	tiny_gpt2	-0.09526	0.866592	0.911832	0.095256	0.771336	
5	phi	-0.10262	0.845146	0.916522	0.102623	0.742523	
6	bert	-0.14015	0.857747	0.92256	0.140153	0.717594	
7	gemini	0.20909	0.883061	0.933359	0.20909	0.673971	
8	mistral	0.210096	0.866094	0.930794	0.210096	0.655998	
9	distilbert	0.241003	0.87836	0.924402	0.241003	0.637357	
10	sentence_	-0.24906	0.845993	0.936869	0.249065	0.596928	
11	roberta	-0.32886	0.85778	0.92937	0.328858	0.528922	
12	gpt	0.509455	0.846959	0.925393	0.509455	0.337504	
13							

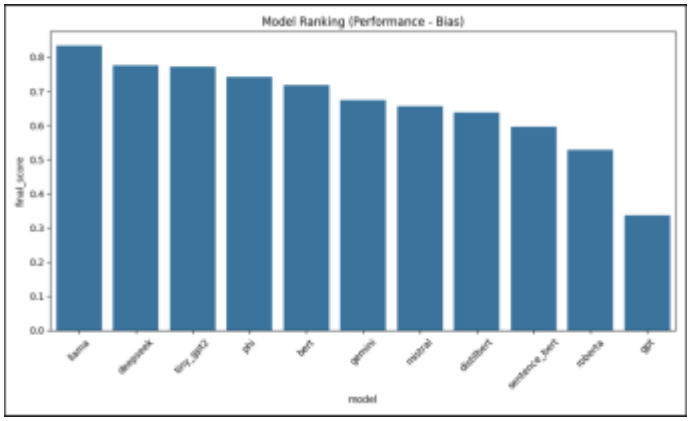

Fig. 10. Combined Score

REFERENCES

[1] S. L. Blodgett, S. Barocas, H. I. Daume´, and H. Wallach, "Language (technology) is power: A critical survey of "bias" in nlp," *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 5454–5476, 2021.

[2] T. J. Hu *et al.*, "Generative language models exhibit social identity biases," *Nature Human Behaviour*, vol. 8, pp. 1795–1806, 2024.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.

[4] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," in *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[5] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," in *Advances in Neural Information Processing Systems (NeurIPS) Workshop on Energy Efficient Machine Learning*, 2019.

[6] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 3982–3992.

[7] M. Nadeem, A. Bethke, and S. Reddy, "Stereoset: Measuring stereotypical bias in pretrained language models," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2021, pp. 5356–5371.

[8] J. Zhao, T. Wang, M. Yatskar, V. O. Celikyilmaz, and K.-W. Chang, "Gender bias in coreference resolution: Evaluation and debiasing methods," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2018, pp. 15–20.

[9] A. Caliskan, J. J. Bryson, and A. Narayanan, "Semantics derived automatically from language corpora contain human-like biases," *Science*, vol. 356, no. 6334, pp. 183–186, 2017.

[10] C. May, A. Wang, J. Barnes, D. Roth, L. Zettlemoyer, and S. Riehle, "Measuring categorical bias in contextualized word representations," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019, pp. 3143–3155.

[11] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," OpenAI, Tech. Rep., 2019.

[12] E. M. Bender and B. Friedman, "Data statements for natural language processing: Toward mitigating system bias and enabling better science," *Transactions of the Association for Computational Linguistics*, vol. 6, pp. 587–604, 2021.

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez,

L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.

[14] J. Zhao, Y. Zhou, Z. Li, K.-W. C. Wang, and T. A. Wang, "Gender bias in contextualized word embeddings," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019, pp. 629–634.

[15] E. Sheng, K.-W. Chang, P. Natarajan, and N. Peng, "The woman worked as a babysitter: On biases in language generation," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 3407–3412.

[16] S. Kiritchenko and S. M. Mohammad, "Examining gender and race bias in two hundred language models of sentiment analysis," in *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics (*SEM)*, 2018, pp. 163–177.

[17] X. Lu and D. Gildea, "Gender bias in neural natural language processing," in *Proceedings of the 1st Workshop on Gender Bias in Natural Language Processing*, 2020, pp. 1–11.

[18] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai, "Man is to computer programmer as woman is to homemaker? debiasing word embeddings," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 29, pp. 4349–4357, 2016.

[19] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning, "A large annotated corpus for learning natural language inference," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2015, pp. 632–642.

[20] H. Touvron, T. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhojanapalli *et al.*, "Llama 2: Open foundation and fine-tuned chat models," 2023, arXiv:2307.09288.

[21] A. Q. Jiang *et al.*, "Mistral 7b," 2023, arXiv:2310.06825.