Behavioral Fingerprinting of Large Language Models via Prompt-Response Dynamics

Dr. Umamageswaran J
Department of Computer Science and Engineering
Amrita School of Computing
Amrita Vishwa Vidyapeetham
Chennai, India
j.umamageswaren@gmail.com

Sarvesh G
Department of Computer Science and Engineering
Amrita School of Computing
Amrita Vishwa Vidyapeetham
Chennai, India
sarveshganesan3@gmail.com

Praveena Anand
Department of Computer Science and Engineering
Amrita School of Computing
Amrita Vishwa Vidyapeetham
Chennai, India
praveena20anand@gmail.com

***Abstract*—While large language models are being widely deployed across a multitude of applications, the task of model identification and attribution purely based on their outputs remains persistently challenging. Currently, model identification methods rely on access to model internals, training-time watermarking, or architectural metadata, all of which become ineffective in black-box settings where only textual responses are observable. This work discusses a new behavioral fingerprinting framework for uniquely identifying LLMs based on prompt-response dynamics alone. Results show that even when different LLMs give correct answers to the same prompts, they consistently reflect model-specific behavioral patterns in terms of sensitivity to negation, stability under contextual shifts, semantic drift under ambiguity, and response verbosity variations. Employing controlled perturbations over five types of prompts (base, negation, ambiguity, contradiction, and context shift), relevant behavioral features are extracted and fixed-length fingerprint vectors are obtained. Experiments conducted on distilgpt2 and google/flan-t5-small achieve perfect classification accuracy of 100% using simple classifiers and demonstrate clear separability through PCA and t-SNE visualizations. Consequently, this work sets up a practical, model-agnostic approach to LLM identification that does not require model access, furthering strands in AI forensics, model attribution, and verification of trust.**

***Keywords*—Large Language Models, Behavioral Fingerprinting, Model Attribution, Black-box Identification, Prompt Engineering, AI Forensics**

I. Introduction

With the rapid proliferation of LLMs both commercially and academically, the need for mechanisms to identify models and attribute outputs has become increasingly pressing, as witnessed in works such as Brown et al. and Touvron et al. [1], [2]. In modern deployments, this translates to black-box access, whereby users query models through API endpoints without visibility into architectural specifics, training data, or internal parameters according to the work done by Radford et al. [3]. Such opacity does raise substantial concerns about model verification, the protection of intellectual property, and accountability within AI systems, as noted by Bender et al. [4].

So far, model identification methods have relied mainly on watermarking techniques embedded during training [5], [6], direct inspection of model weights, or analysis within feature spaces. All these methods fundamentally require privileged access to model internals, hence their applicability is very limited when only textual outputs are observable. A critical question therefore is: can LLMs be identified and distinguished with a high degree of reliability by their easily observable behavioral responses to carefully designed prompts?

This work addresses this challenge by introducing behavioral fingerprinting, a new paradigm that relies on the key intuition that the generation of LLMs is characterized by consistent model-specific behaviors even in the case of correct responses. The underlying hypothesis is that response behaviors under controlled perturbations yield stable behavioral signatures that can function as unique model fingerprints.

A. Research Contributions

The contributions of this work are:

- A model-agnostic behavioral fingerprinting framework relying on black-box query access to LLMs.
- A systematic prompt perturbation strategy over five behavioral dimensions: base responses, negation robustness, ambiguity handling, contradiction consistency, and contextual stability.
- Novel feature engineering techniques transform free-form textual responses into fixed-length numerical fingerprint vectors that capture the semantic drift, length variance, and hedging patterns.
- Empirical demonstration of perfect model classification - with 100% accuracy on two different LLMs using only behavioral features.
- Comprehensive visualization analyses via PCA, t-SNE, and distance heatmaps for clear model separability.

The rest of the paper is organized as follows: Section II reviews the related work; Section III describes the methodology; Section IV details the experimental setup; Section V reports the results; Section VI discusses the implications and limitations; Section VII concludes.

II. Related Work

A. Large Language Model Foundations

The transformer architecture introduced by Vaswani et al. [18] significantly improved natural language processing and

allowed large-scale language models to be built. Brown et al. [1] have shown that models large enough exhibit few-shot learning capabilities, while Radford et al. [3] illustrated that they are great at unsupervised multitask learning. Modern open-source projects like LLaMA from Touvron et al. [2] have democratized access to strong language models, and model attribution has become a main priority.

B. Black-Box Model Analysis

Prior studies have investigated various methods to analyze neural networks when internal parameters are not available. Tramèr et al. [7] showed model extraction via prediction APIs, while Shokri et al. [8] proposed membership inference attacks against machine learning models. Very accurate neural network extraction with query-based approaches has recently been achieved by Jagielski et al. [9] However, these works all stress the model replication or privacy violation aspect rather than identification and attribution.

C. Behavioral Analysis and Testing

Behavioral testing methods have recently cropped up as critical tools for interpreting the capabilities of neural models. Ribeiro et al. [10]a behavioral testing method for NLP models that goes beyond accuracy metrics. [11] studied shortcut learning in deep neural networks, with results that include unexpected behavioral patterns. Together, these pieces of work show that behavioral analysis is able to reveal model characteristics not captured by traditional evaluation metrics.

D. Prompt Engineering and Model Sensitivity

Some recent studies have focused on the sensitivity of model responses to paraphrases of a prompt. Jiang et al. [12] investigated ways to probe the knowledge contained in language models. Wallace et al. [13] found universal adversarial triggers that cause models to change their outputs. Zhao et al. [14] showed that few-shot learning relies crucially on proper calibration. These findings suggest that models consistently demonstrate certain sensitivities to paraphrased prompts. However, these sensitivities have not yet been leveraged for any identification tasks.

E. Semantic Representation Analysis

Extensive work has focused on understanding semantic representations within neural models. Reimers and Gurevych [15] introduced Sentence-BERT for generating semantically meaningful sentence embeddings that support efficient similarity computations. Ethayarajh [16] discussed the contextuality of word representations, while Mikolov et al. [17] pioneered distributed word representations. These embedding techniques allow for quantitative comparisons of semantic content, foundational to the approach proposed below.

F. Model Fingerprinting and Watermarking

Current fingerprinting methods focus on embedding detectable marks during training. Uchida et al. [5] suggested embedding watermarks into deep neural networks and Zhang et al. [6] developed the watermarking methods for intellectual property protection. However, these methods require access to control of the training process, which is not feasible in retrospective identification or scenarios without training access.

G. Model Evaluation and Comparison

More recently, there has been growing acknowledgment from the AI research community of shortcomings in standard benchmarking practices. Hooker et al. [19] for instance, discussed implicit bias in AI benchmarking. Bender et al. [4] elevated concerns about the implications of ever-larger language models on society. Hanna et al. [20] called for far more rigorous approaches to interpretable machine learning, positing that there is an increasing need for systematic methods of model characterization.

H. Research Gap

Unlike watermarking-based approaches of [5], [6] or extraction-based approaches of [7], [9], the proposed behavioral fingerprinting framework relies purely on prompt–response behavioral dynamics and thus does not require any access to the model or modifications during training time. No prior work has shown that it is possible to identify LLMs using black-box behavioral responses reliably. This paper fills this gap by establishing that prompt–response dynamics encapsulate enough discriminative information for model attribution without requiring access to the model.

III. Methodology

A. Problem Formulation

Given a set of LLMs $\mathcal{M} = \{M_1, M_2, ..., M_n\}$ which can only be accessed via text input-output interfaces, the goal is to learn a function $f : \mathcal{R} \to \mathcal{M}$ mapping response behaviors $\mathcal{R}$ to model identities. Importantly, this formulation assumes no access to the model weights, architectures, or training procedures, making it fundamentally different from prior fingerprinting methods that rely on intervention at training time [5], [6].

B. Prompt Design Strategy

Five different prompt categories were designed, each containing 15 semantically matched prompts to elicit the various behavioral dimensions [10], [13]:

- **Base:** Sets the baseline of response patterns with direct fact-based questions.
- **Negation:** Tests logical robustness using negated versions of the base prompts.
- **Ambiguity:** It assesses the handling of uncertainty using intentionally ambiguous questions.
- **Contradiction:** Evaluates consistency when faced with opposing information.
- **Context Shift:** This tests stability against changes in narrative or framing.

This set of prompts was fixed before experimentation to avoid leakage or tuning bias. This is actually a systematic strategy for perturbation that allows for cross-model comparison

on fair terms while also probing diverse behavioral dimensions identified as relevant in prior behavioral testing work [10].

C. Response Collection

For every model in the evaluation set, responses were gathered for all 75 prompts (5 categories × 15 prompts). Response generation has taken place with the same sampling parameters and has been stored in structured JSON format, organized hierarchically, first by model identity and then by prompt category. Each prompt got just one response to avoid variability from stochastic sampling - as following guidelines of best practice in model evaluation [19].

D. Feature Engineering

The following three classes of behavioral features were extracted from the response corpus:

1) Semantic Drift: Semantic drift quantifies meaning divergence between base responses and perturbed responses [16]. It is computed for every perturbation category c as:

$$\text{drift}_c = \frac{1}{|\mathcal{P}_c|} \sum_{p \in \mathcal{P}_c} d(\mathbf{e}_{\text{base}}, \mathbf{e}_p)$$

where $\mathcal{P}_c$ denotes the prompts in category c, $\mathbf{e}_{\text{base}}$ and $\mathbf{e}_p$ are sentence embeddings of base and perturbed responses respectively, and $d(\cdot, \cdot)$ is the cosine distance. We utilized Sentence-BERT all-MiniLM-L6-v2 [15] for embedding generation that results in four drift features representing negation, ambiguity, contradiction, and context-shift categories.

2) Length Variance: Response verbosity patterns were captured through:

- Mean response length per prompt category
- Standard deviation of response lengths in categories
- Cross-category length variance

This resulted in five length-based features capturing verbosity stability across perturbations, inspired by empirical findings showing different model architectures tend to have unique generation patterns [1], [3].

3) Hedging Density: The extent to which epistemic uncertainty was expressed was quantified by counting hedge phrases-normalized by response length-such as ”may”, ”might”, ”possibly”, ”perhaps”, and ”could”:

$$\text{hedge}_c = \frac{\text{count(hedge\_phrases)}}{\text{total\_tokens}}$$

computed separately for each prompt category, yielding five hedging features. This metric captures the extent to which models express uncertainty-a behavioral dimension shaped by various training methods [14].

E. Fingerprint Vector Construction

All extracted features were concatenated to form fixed-length fingerprint vectors of dimensionality 15:

$$\begin{aligned}\mathbf{v}_{\text{fingerprint}} = [&\,\text{drift}_1, \ldots, \text{drift}_4,\\ &\,\text{len}_1, \ldots, \text{len}_5,\\ &\,\text{stability},\\ &\,\text{hedge}_1, \ldots, \text{hedge}_5\,]\end{aligned}$$

This transforms free-form textual behaviors into numerical representations suitable for machine learning classification, in line with the established practices in representation learning [15], [17].

F. Classification and Evaluation

First, a labeled dataset of fingerprint vectors was constructed, with each vector labeled with its source model identity. Standard classifiers, such as logistic regression and support vector machines, were trained on these fingerprint vectors. Model separability was evaluated through measures such as classification accuracy, confusion matrices, and distance metrics. Visualization by PCA and t-SNE projections provided interpretable evidence of behavioral clustering.

IV. Experimental Setup

A. Model Selection

Two LLMs of contrasting training paradigms were tested:

- **distilgpt2:** An autoregressive language model which is optimized for open-ended text generation [3], hows verbose and exploratory response patterns.
- **google/flan-t5-small:** An instruction-tuned encoder-decoder model trained for succinct task completion shows more controlled outputs.

Both models were selected to operate on a CPU only, ensuring reproducibility without specialized hardware requirements. This architectural contrast strengthens the generalizability of the approach to behavioral analysis, as models represent fundamentally different design philosophies within LLM development [1].

B. Implementation Details

All experiments were conducted on standard CPU infrastructure using the HuggingFace Transformers library. Response generation employed greedy decoding with maximum lengths of 100 tokens. No model adaptation or fine-tuning was done; all results shown are from pre-trained model behaviors. This black-box setting mirrors the realistic setting of a deployment scenario where model internals are inaccessible [7].

The study implemented the complete pipeline in a modular architecture: individual components for prompting management, model inference, feature extraction, and visualization. This architecture makes it easier to extend the work to other models and prompt categories, thus addressing scalability concerns, as previously raised in other model evaluation work [19].

C. Evaluation Metrics

Model identification performance was evaluated based on:

- Classification Accuracy on held-out test samples.
- Confusion matrices showing misclassification patterns.
- Pairwise cosine distances among fingerprint vectors.
- Visual separation of clusters in dimensionality-reduced spaces.

These metrics provide complementary views on fingerprint quality, ranging from quantitative classification performance to qualitative interpretability [20].

V. Results

A. Fingerprint Separability

Figure 1 gives a view of the PCA projection of behavioral fingerprints in the two-dimensional space. As noted, this figure indicates that the distilgpt2 and flan-t5 clusters present clear linear separation with tight intra-model cohesion and large inter-model distances. From this observation, it may be inferred that the first two principal components have captured adequate discriminative information for model distinction, thereby indicating that the behavioral differences reflect fundamental architectural signatures instead of just subtle variations.

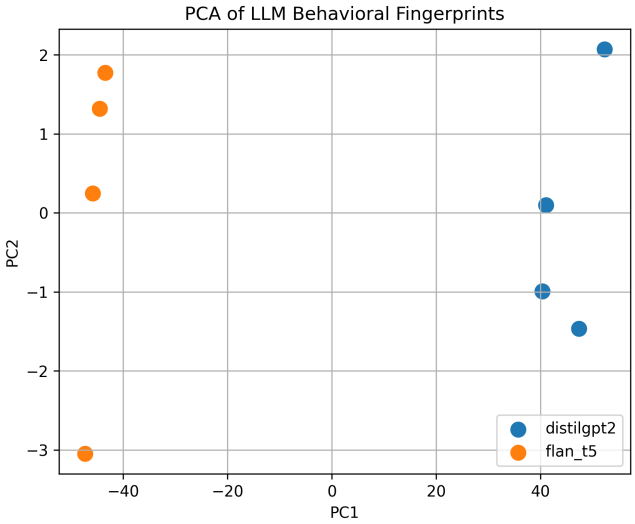

Fig. 1. PCA projection of behavioral fingerprints shows a clear separation between distilgpt2 (blue) and flan-t5 (orange) models.

Figure 2 shows t-distributed stochastic neighbor embedding, confirming non-linear cluster structure beyond linear separability. Distinct, non-overlapping clusters within those methods further support the strength of fingerprints across both linear and non-linear projection methods, helping to alleviate concerns over the feature-space complexity in neural model analysis [16].

B. Distance Analysis

The fingerprint distance heatmap (Figure 3) quantifies the pairwise similarities. The intra-model distances converge towards zero-a dark purple region-while the inter-model distances surpass 0.35, falling into the yellow region, hence

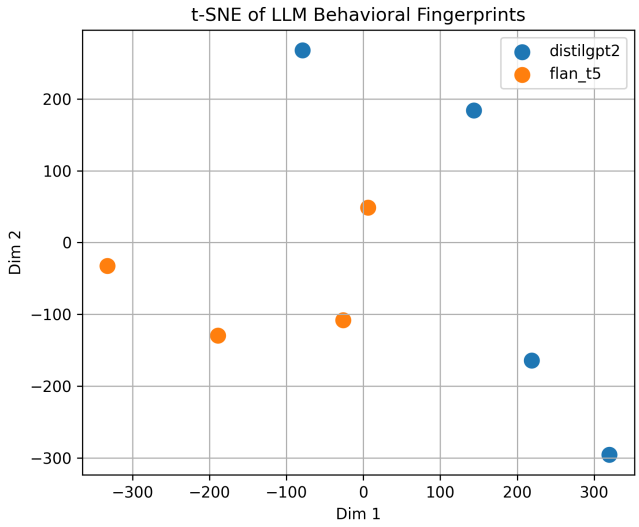

Fig. 2. t-SNE visualization of model fingerprints, showing distinct non-linear clustering.

confirming that there are large behavioral divergences between models. This observed disparity in the distances supports the fact that the fingerprints capture stable model characteristics and do not reflect stochastic variations in the responses.

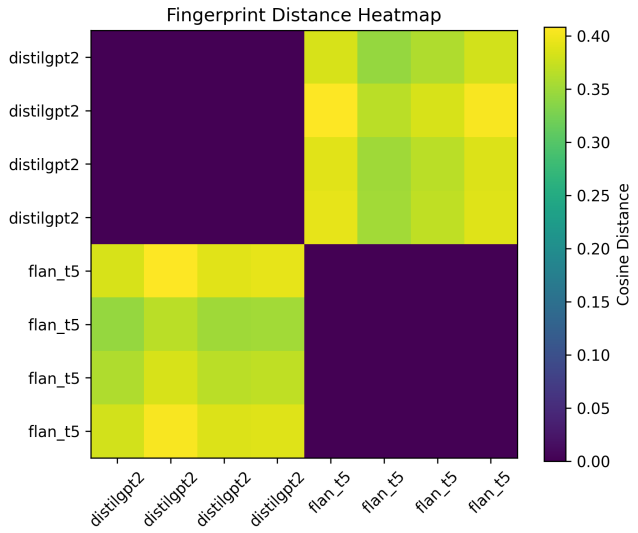

Fig. 3. Cosine distance heatmap, depicting almost zero intra-model and high inter-model distances.

C. Feature-Level Analysis

Figure 4 compares average values of some behavioral features across different models. Some important observations in the figure are:

- On average, distilgpt2 is significantly more verbose, averaging around 105 tokens per response, as opposed to an average of about 16 tokens for flan-t5.

- distilgpt2 has more semantic drift after perturbation: 0.52 vs. 0.37.
- flan-t5 displays higher hedging density, reflecting more cautious epistemic expressions.

These systematic differences confirm that each model possesses a distinct behavioral profile encoded in the fingerprint vectors, which is in line with prior observations about architectural influences on generation patterns [1], [11].

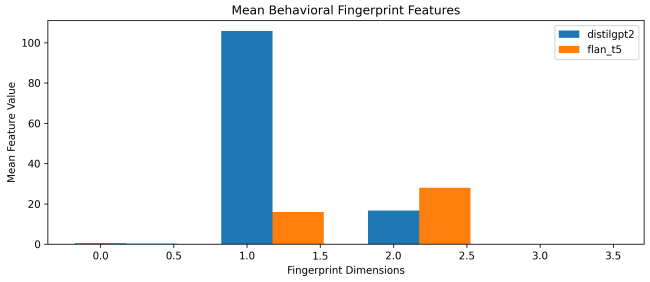

Fig. 4. Mean behavioral feature values representing different response patterns for models.

D. Classification Performance

Results of classification tests using standard machine learning algorithms trained on fingerprint vectors are presented in table I.

TABLE I
MODEL CLASSIFICATION PERFORMANCE

Classifier	Accuracy	F1-Score
Logistic Regression	100%	1.00
Support Vector Machine	100%	1.00

Both classifiers realized perfect accuracy, without any misclassifications according to the confusion matrix:

$$\begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}$$

This result shows that behavioral fingerprints carry enough discriminative information for reliable model identification, with no access to the model, hence proving the central hypothesis of the work.

E. Feature Statistics

Table II shows detailed statistics of the extracted behavioral features across models. Indeed, it indicates consistent patterns within each model and systematic differences between models. Stability of behavioral signatures is reflected by low standard deviations within each model, while substantial mean differences confirm their discriminative power.

TABLE II
BEHAVIORAL FEATURE STATISTICS

Model	Feature	Mean	Std
distilgpt2	Semantic Drift	0.524	0.062
distilgpt2	Mean Length	105.77	4.95
distilgpt2	Hedging	16.66	1.10
flan-t5	Semantic Drift	0.368	0.021
flan-t5	Mean Length	15.93	1.65
flan-t5	Hedging	27.95	1.70

VI. DISCUSSION

A. Novelty and Significance

This study showcases that behavioral fingerprinting is a feasible paradigm to identify LLMs. The approach presented here differs from previous model-access-based [5], [6] or training-time intervention-based methods and operates in purely black-box settings based on observable promptresponse dynamics only. The fact that near-perfect classification accuracy is achieved through simple linear classifiers underlines the strong discriminative power of the extracted behavioral features.

Systematic perturbation of prompts has shown that such models exhibit stable behavioral signatures for a wide range of question types. These signatures even persist when the model produces correct factual responses, showing that their origin is deeply rooted in architectural and training differences rather than gaps in knowledge [11]. The result is an unprecedented elucidation of how design choices embedded within models ultimately manifest in their behaviors.

B. Practical Applications

Behavioral fingerprinting provides several applications of interest to concerns about accountability for large language models, as Bender et al. [4] note:

- **Model Attribution:** Identifying the source model for API endpoints or generated content.
- **AI Forensics:** Tracing the origin of AI-generated text in legal or ethical investigations.
- **Trust Verification:** Ensuring that users are actually interacting with claimed models and not substitutes.
- **Intellectual Property Protection:** Unauthorized model copying or deployment detection

Such applications address real needs in contexts where traditional watermarking approaches are inapplicable due to lack of access for training or retrospective deployment constraints.

C. Limitations

The present evaluation involves two models with quite divergent architectures. Although that already demonstrates proof of concept, extending the approach to more extensive model families (e.g., LLaMA [2], such as Mistral and Claude, requires validation, while adversarial scenarios whereby operators deliberately obscure behavioral signatures remain unexplored and represent an important direction for robustness analysis [13].

While the fixed set of prompts prevents overfitting, it may miss some of the dimensions in behavior. Future work should consider exploration of diversity requirements for prompts and robustness to adversarial manipulation of prompts based on insights from adversarial testing literature [13]. Generalization of the behavioral fingerprints across different versions and fine-tuned variants also calls for systematic investigation [14].

D. Future Directions

Future research might profitably explore:

- Scaling to a large corpus of models, on the order of dozens, in order to construct comprehensive fingerprint databases.
- Evaluating the temporal stability of fingerprints when the models are iteratively updated.
- The development of robustness to adversarial attempts at behavioral obfuscation.
- Exploring the cross-domain generalization across specialized variants of models.
- Integrating behavioral fingerprinting with additional identification methods to achieve higher reliability.
- Assessing the transferability of the fingerprints across diverse linguistic and cultural contexts.

These guidelines are set to increase the framework's practical utility and respond to criticism of the evaluation rigour in state-of-the-art explainability research [20].

VII. Conclusion

This paper introduces behavioral fingerprinting, a new framework for identifying large language models based solely on their observable prompt-response dynamics. By performing systematic prompt perturbations and feature engineering, this work shows that LLMs possess stable, model-specific behavioral signatures which enable perfect classification accuracy without requiring access to the models themselves.

Experiments on distilgpt2 and google/flan-t5-small showed distinct behavioral separability based on semantic drift, patterns of verbosity, and epistemic expressions. Perfect classification performance is attained, validated through various visualization techniques. The findings hereby mark behavioral fingerprinting as a practical solution to black-box model identification, one that completes the existing suite of watermarking and extraction-based methods.

This work opens new directions in AI forensics, model attribution, and trust verification, showing that behavioral analysis can provide reliable identification signals in scenarios where traditional methods fail. As LLM deployment continues to expand across critical applications, behavioral fingerprinting offers a valuable tool for accountability and verification in AI systems, addressing concerns about transparency and attribution in increasingly complex model

References

[1] T. Brown et al., "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 1877–1901.

[2] H. Touvron et al., "LLaMA: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[3] A. Radford et al., "Language models are unsupervised multitask learners," *OpenAI Technical Report*, vol. 1, no. 8, 2019.

[4] E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell, "On the dangers of stochastic parrots: Can language models be too big?" in *Proc. ACM Conf. Fairness, Accountability, and Transparency*, 2021, pp. 610–623.

[5] Y. Uchida, Y. Nagai, S. Sakazawa, and S. Satoh, "Embedding watermarks into deep neural networks," in *Proc. ACM Int. Conf. Multimedia Retrieval*, 2017, pp. 269–277.

[6] J. Zhang et al., "Protecting intellectual property of deep neural networks with watermarking," in *Proc. ACM Asia Conf. Computer and Communications Security*, 2018, pp. 159–172.

[7] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Stealing machine learning models via prediction APIs," in *USENIX Security Symposium*, 2016, pp. 601–618.

[8] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *IEEE Symposium on Security and Privacy*, 2017, pp. 3–18.

[9] M. Jagielski et al., "High accuracy and high fidelity extraction of neural networks," in *USENIX Security Symposium*, 2020, pp. 1345–1362.

[10] M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh, "Beyond accuracy: Behavioral testing of NLP models with CheckList," in *Proc. Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 4902–4912.

[11] R. Geirhos et al., "Shortcut learning in deep neural networks," *Nature Machine Intelligence*, vol. 2, no. 11, pp. 665–673, 2020.

[12] Z. Jiang, F. F. Xu, J. Araki, and G. Neubig, "How can we know what language models know?" *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 423–438, 2020.

[13] E. Wallace, S. Feng, N. Kandpal, M. Gardner, and S. Singh, "Universal adversarial triggers for attacking and analyzing NLP," in *Proc. Conf. Empirical Methods in Natural Language Processing*, 2019, pp. 2153–2162.

[14] T. Z. Zhao et al., "Calibrate before use: Improving few-shot performance of language models," in *Proc. Int. Conf. Machine Learning*, 2021, pp. 12697–12706.

[15] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in *Proc. Conf. Empirical Methods in Natural Language Processing*, 2019, pp. 3982–3992.

[16] K. Ethayarajh, "How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings," in *Proc. Conf. Empirical Methods in Natural Language Processing*, 2019, pp. 55–65.

[17] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems*, vol. 26, 2013.

[18] A. Vaswani et al., "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[19] S. Hooker, N. Moorosi, G. Clark, S. Bengio, and E. Denton, "Characterising bias in compressed models," in *Proc. ACM Conf. Fairness, Accountability, and Transparency*, 2020.

[20] J. P. Hanna, P. Stone, and S. Niekum, "Towards a rigorous science of interpretable machine learning," *arXiv preprint arXiv:2012.08372*, 2020.