DO STOP ME NOW: DETECTING BOILERPLATE RESPONSES WITH A SINGLE ITERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Language Models (LLMs) often expend significant computational resources generating boilerplate responses, such as refusals, simple acknowledgements and casual greetings, which adds unnecessary cost and latency. To address this inefficiency, we propose a simple yet highly effective method for detecting such responses after only a single generation step. We demonstrate that the log-probability distribution of the first generated token serves as a powerful signal for classifying the nature of the entire subsequent response. Our experiments, conducted across a diverse range of small, large, and reasoning-specialized models, show that the first-token log-probability vectors form distinctly separable clusters for different response types. Using a lightweight k-NN classifier, we achieve high accuracy in predicting whether a response will be a substantive answer or a form of boilerplate response, including user-specified refusals. The primary implication is a practical, computationally trivial technique, optimizing LLM inference by enabling early termination or redirection to a smaller model, thereby yielding significant savings in computational cost. This work presents a direct path toward more efficient and sustainable LLM deployment.

1 INTRODUCTION

Large Language Models (LLMs) have revolutionized artificial intelligence with their ability to understand and generate human-like text across diverse applications, from conversational agents to code assistants. This remarkable capability comes with a significant challenge: the high computational and financial costs associated with llm inference for each user query (Regmi & Pun, 2024). These costs are especially wasteful in real-world scenarios where LLMs generates unwanted or predictable outputs. For instance, OpenAI CEO Sam Altman has publicly stated that politeness expressions like ”please” and ”thank you” have cost the company tens of millions of dollars, due to the electricity consumption associated with generating these boilerplate responses (USA Today, 2025). This highlights a critical inefficiency: LLMs often produce unnecessary tokens that consume resources without contributing to the user's core intent. The ability to accurately and cost-effectively characterize LLM responses *prior to or early in their generation* is thus paramount for optimizing inference costs, reducing latency, and enhancing the overall sustainability of LLM-powered systems.

To address this inefficiency, recent research purposes several novel methods, specifically concerning refusal-to-comply (i.e. due to safeguards). For example, strategies like refusal tokens (Jain et al., 2025) involve prepending special tokens during training that the model learns to generate first when a refusal is appropriate. Other studies show that LLMs encode global attributes of their future responses in hidden representations even before any tokens are generated, enabling emergent response planning (Dong et al., 2025). Another study (Arditi et al., 2024), reveals that refusal behavior is classifiable by a one-dimensional subspace in LLM activations, leading to the development of a ”refusal metric”. This metric, derived from summing probabilities assigned to specific ”refusal tokens” at early generation stages, serves as an efficient proxy for measuring refusal likelihood without full response generation, aligning closely with our objectives for cost-effective content filtering.

This work builds upon these advancements and shows that the **log-probabilities of the first generated token** are sufficient for accurate **prediction of multiple response types**, including user-specified refusals. By focusing on early prediction and detection of boilerplate content (such as refusals, gratitude acknowledgements and other non-task-solving elements), we aim to significantly

reduce the computational and environmental footprint of LLM inference, ultimately leading to more efficient and sustainable AI systems.

2 PRELIMINARIES

2.1 DEFINING BOILERPLATE RESPONSES

We define boilerplate responses as responses that are interchangeable within their class. For example, **refusal type response** (i.e. *"I'm sorry, I cannot help you with ____"* or *"You're welcome! I'm glad I could assist you with ____"*) are interchangeable up to context. Efficient and cost-effective identification of boilerplate enables optimizing inference costs in real-world applications. This involves distinguishing content such as refusals, gratitude, or simple acknowledgements from genuinely meaningful, task-solving outputs. Recent advancements in LLM research offer several avenues to address this challenge, often by leveraging the internal mechanisms of these models for early prediction or classification.

3 RELATED WORK

3.1 DIALOGUE ACT CLASSIFICATION FOR CONVERSATIONAL INTENT

In conversational settings, Dialogue Act Classification (DAC) identifies the communicative intent behind utterances (Zhangwenbo & Yuhan, 2025; Aljanaideh, 2025). LLMs exhibit zero-shot DAC capabilities, which can be refined iteratively with online feedback without requiring labeled data (Zhangwenbo & Yuhan, 2025). This allows for the early classification of dialogue acts, including those that signify conversational fillers, gratitude, or simple acknowledgements, thereby identifying non-meaningful conversational turns.

3.2 DISTINGUISHING MEANINGFUL CONTENT FROM BOILERPLATE ELEMENTS

A key challenge is the precise differentiation between essential, reasoning tokens and repetitive, non-critical boilerplate tokens (e.g., formatting, transitional phrases like *"Based on the user's request..."*) (Ye et al., 2025). The Shuffle-Aware Discriminator (SHAD) offers an automated and adaptive solution by exploiting predictability differences after shuffling input-output combinations: boilerplate tokens remain predictable, while reasoning tokens do not.

3.3 OPERATIONAL EFFICIENCY METHODS FOR COST REDUCTION

Beyond direct classification, other techniques aim to reduce LLM inference costs by avoiding unnecessary computation. Semantic caching mechanisms, such as GPT Semantic Cache, leverage query embeddings to identify semantically similar questions and retrieve pre-generated responses, significantly reducing API calls and improving response times, especially for repeated boilerplate queries (Regmi & Pun, 2024). Additionally, advancements in multi-token prediction enable LLMs to jointly predict several subsequent tokens in a single inference step (Orgad et al., 2025). While this primarily speeds up generation, it could potentially be adapted to quickly scan for and flag boilerplate patterns, allowing for early termination. Another approach is to use model routing, sending simpler prompts (such as those leading to boilerplate responses) to a smaller, cheaper model (Ding et al., 2024).

3.4 EARLY PREDICTION OF OUTPUT CHARACTERISTICS

A foundational concept in this area is emergent response planning, where LLMs' internal hidden representations can encode global attributes of an entire future response before any tokens are generated (Dong et al., 2025). By probing these pre-generation representations, it is possible to predict various characteristics of the upcoming output, such as response length, reasoning steps, answer confidence, or factual consistency. This capability is instrumental for pre-generation resource allocation optimization, allowing systems to anticipate the nature of a response and potentially avoid

costly full generations of boilerplate content. Complementary to this, methods like TRAIL (Shahout et al., 2025) leverage recycled LLM layer embeddings to dynamically predict remaining output length with low overhead and high accuracy, refining predictions at each token generation step. This approach, similar to others employing separate lightweight LLMs or BERT models (Devlin et al., 2019) for length prediction, aims to optimize scheduling and reduce latency, indirectly supporting early termination for short, non-meaningful responses.

3.5 Refusal Detection at Inference Time

For specific boilerplate types like refusals, specialized mechanisms have been developed. The Refusal Tokens strategy proposes prepending a special `[refuse]` token (or category-specific tokens) to responses during training, allowing the model to learn to generate this token first when a refusal is appropriate (Jain et al., 2025). At test-time, the softmax probability of this refusal token quantifies the likelihood of a refusal, enabling calibrated control over refusal rates without retraining. This enables a "cheap sweep" by allowing identification of optimal refusal thresholds with a single forward pass, avoiding full response generation. Notably, this method allows for fine-grained control over various refusal types if multiple tokens are used.

Similarly, (Arditi et al., 2024) demonstrates that refusal behavior in LLMs is mediated by a one-dimensional subspace within their residual stream activations. This allows for the derivation of a "refusal metric" by summing the probabilities assigned to a predefined set of "refusal tokens" (e.g., "I'm sorry", "I cannot") at the first token position of the prompt. This metric serves as an efficient proxy for estimating the likelihood of a model refusing an instruction without requiring full response generation. This approach is the most closely related to our work, as it focuses on leveraging early signals from the model's generation output to predict the nature of the upcoming response. Yet, unlike our work, it requires manual listing of these "refusal tokens".

Beyond explicit refusals, research into LLM transparency also explores their internal signals for other content characteristics. Studies on LLM hallucinations indicate that truthfulness information is concentrated in "exact answer tokens" within the generated response (Orgad et al., 2025). Probing classifiers trained on intermediate representations of these tokens can predict errors, suggesting that LLMs encode information about their own truthfulness. While focused on error detection, this highlights the general potential to probe internal states for the "meaningfulness" of a response.

4 Our Method

When using LLMs to generate responses, they do so one token at a time. At each iteration, *all possible tokens* are assigned probabilities, but only one is selected. Thus, examining all token probabilities of a *single iteration* provides an overview of all the possible subsequent responses. We hypothesize that by using the (log-)probabilities of the first token generation, it is possible to classify certain response types. To validate it, we measure the similarities between log-probabilities of the first token generated by different prompts, designed to induce boilerplate responses versus others that suppose to elicit more detailed responses related to the chat. We perform these validation over different models, both small and large language models, as the token-probability-space is not comparable between different models.

4.1 Dataset

We create a unique dataset of ~3k different chats of different lengths and different classes. We define several types of classes:

- **Refusal**: Chats or messages an assistant will refuse to answer due to internal learnt safeguards.
- **Thanks**: Chats ending with the user thanking the assistant for its assistance. These usually make the assistant respond with common phrases like *"You're welcome!"* or *"My pleasure!"*
- **Hello**: Chats starting with the user saying *"Hello!"*, *"Hi!"* or similar texts.

- **Chat**: All other chats that do not belong to above classes, where the user and the assistant are having a regular conversation.

The dataset was created in the following way:

1. We use the AdvBench dataset (Zou et al., 2023), containing harmful prompts, and classify these as Refusal.
2. We then sampled ~500 random prompts from the Alpaca dataset (Taori et al., 2023), and classify these as Chat. The input-prompts of the Alpaca dataset are split into two columns: *instruction* and *input*, where in most cases, the *input* column is empty. We explicitly used only the *instruction* column as the prompt, thus creating some cases of chats with missing context.
3. Per each of the prompts we now have, we prompted an LLM to respond, and recorded the response. We then asked an LLM to continue the conversation as the user would, and if the model refused to reply - steer the conversation to a legitimate follow up question. We combined the User-Assistant-User interactions as new examples, and labeled them as Chat.
4. For the original harmful prompts, we asked an LLM to hypothesize what was the legitimate prompt the user asked *prior* to the harmful prompt. We then asked an LLM to respond the the legitimate prompt, and added these User(safe)-Assistant-User(harmful) interactions as Refusal.
5. Next, we asked an LLM to come with 250 "benign" thank-you prompts a user might send to an assistant (i.e. *"Thank you for your help!"*). We then sampled 500 User-Assistant-User chats, replaced the last message with a random thank-you prompt, and labeled them as Thanks.
6. Finally, we created a list of ~30 "benign" prompts which can be used to initiate a conversation with an assistant without any additional request (such as *"Hello!"* or *"Good morning!"*). We labeled these as Hello.

The result is a dataset of ~3k chats, containing both single prompt chats and User-Assistant-User chats. The dataset is available at `ANON_URL`[1].

4.2 Response Type Clustering and Classification

We prompt selected language models with the chats from our dataset, and then record the log-probabilities vectors of the first token generated as the LLM's response. We visualize the results using 2D t-SNE (van der Maaten & Hinton, 2008). To quantitatively evaluate the effectiveness of our approach, we trained k-Nearest Neighbors (k-NN) classifiers (Cover & Hart, 1967) on the first-token log-probability embeddings for each model category. The k-NN algorithm was chosen for its simplicity and interpretability, enabling direct measurement of the clusters separability. For each model, we performed 5-fold stratified cross-validation to ensure robust evaluation across all response types (Chat, Hello, Refusal, Thanks). We fixed $k = 3$ for all models to enable direct comparison across different architectures. All reported metrics (accuracy, precision, recall, F1-score) are cross-validation results averaged across the 5 folds, providing more reliable estimates than single train-test splits. We report macro-averaged precision, recall, and F1-score to account for class imbalance, particularly for the Hello class which represents only ~1.4% of the dataset.

5 Experiments

We verify our hypothesis on three scenarios: **Small Language Models**, **Reasoning Models** and **Large Language Models**.

5.1 Small Language Models

We perform an evaluation of the first-token log-probabilities of the following models:

[1]Dataset is attached to anonymous submission, dataset link is hidden for anonymity

- Llama 3.2 3B (Meta AI, 2024)
- Qwen 2.5 1.5B (Qwen Team, 2024; Yang et al., 2024)
- Gemma 3 1B (Gemma Team et al., 2025)

A 2D t-SNE plot of these models' log-probabilities is shown in Figure 1 and Table 1. We clearly see a separation between the classes.

5.1.1 Refusal due to Incapability

When examining Chat samples located much closer to the Refusal class, we find mostly the chats with the missing context we created by omitting the *input* column from the Alpaca dataset. The following list provides a few examples of such incomplete chat messages found more closely to the Refusal class:

- *"Based on the provided input paragraph, provide a summary of its content."*
- *"Translate the sentence below into Japanese."*
- *"Describe the character of the protagonist in the given TV show."*
- *"Group the given list into 3 Groups."*

The assistants are now incapable of replying to these prompts, as they now miss critical context. They therefore trigger a refusal response from the assistants, as they try to explain to the user that they cannot assist - not due to safeguards, but due to lack of context.

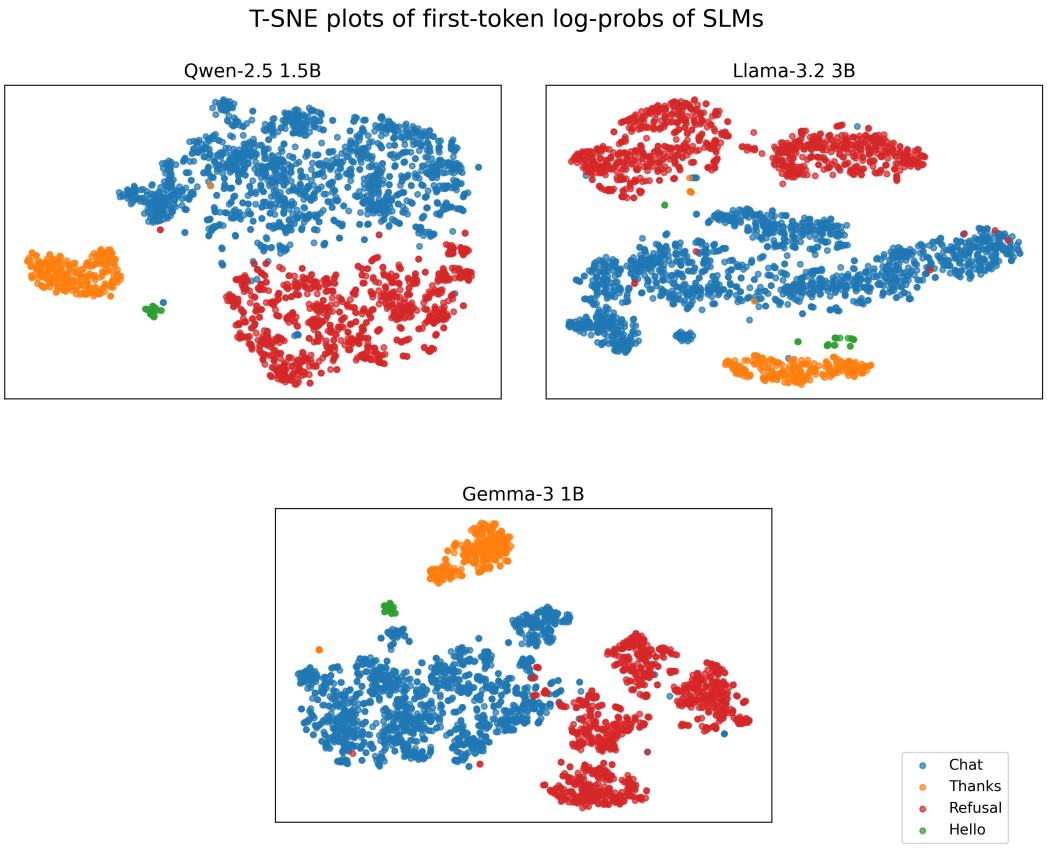

Figure 1: 2D T-SNE plot of first-token log-probabilities for Small Language Models (Llama 3.2 3B, Qwen 2.5 1.5B, Gemma 3 1B). Each point represents a chat, colored by class.

Table 1: Small Language Models Performance on Type Classification (k=3)

Model	Accuracy	Precision	Recall	F1	Chat F1	Hello F1	Refusal F1	Thanks F1
Qwen2.5-1.5B	0.997	0.991	0.998	0.994	0.998	1.000	0.998	0.998
Llama-3.2-3B	0.995	0.996	0.984	0.990	0.998	1.000	0.998	0.996
Gemma-3-1B-IT	0.994	0.997	0.997	0.997	0.998	1.000	0.997	1.000

5.1.2 Refusal due to System Prompt

So far, chats marked as Refusal are those AI assistants tend to refuse to reply to due to internal trained safeguards. We wish to check whether this method applies too when the refusal is due to an arbitrary system prompt, stating that the assistant cannot reply to certain scenarios.

In the experiment shown in Figure 2, we send the assistant a request for a recipe for a Black Forest Cake. We did so twice per model, once with a system prompt explicitly stating that the assistant *cannot* provide a recipe for a Black Forest Cake, and once without any additional restrictions. Figure 2 shows the same 2D T-SNE plot of the first-token log-probabilities of SLM, along with the log-probabilities of the first token of each of the Black Forest Cake prompt variations. We clearly see that when the model was instructed not to provide a recipe, the first token log-probabilities are closer to the Refusal class center-of-mass. While not shown here, other such experiments yielded similar results.

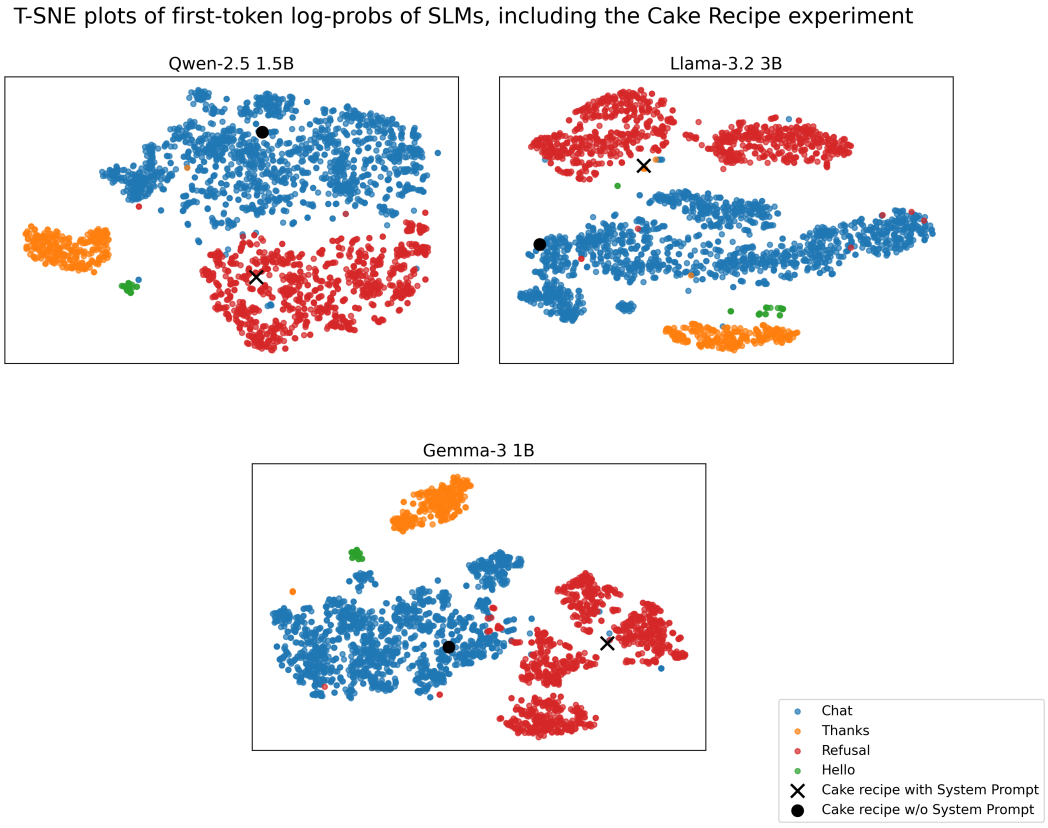

Figure 2: 2D T-SNE plot of first-token log-probabilities for Small Language Models, including the Cake Recipe experiment. The black circle represents a request for a recipe for a Black Forest Cake, and the black cross represents the same requests, but with a system prompt instructing the assistant not to provide the recipe.

5.2 Reasoning Models

The examination of reasoning models adds a bit more complexity, as they tend to begin with a thinking phase which includes self-explaining and "self-talking" (i.e. *"So, the user asked..."*, *"I need to..."*, etc.). Since our classification applies to the first token of the **response**, we slightly adjusted the chat inputs by adding an empty-thinking phase to the assistant's message (i.e. *"<think></think>"*), and only then checked the log-probabilities of the first token.

We evaluated the following reasoning models:

- DeepSeek-R1 8B (Llama Distillation) (DeepSeek-AI, 2025)
- Phi-4 Reasoning Plus (Abdin et al., 2025)

Results are shown in Figure 3 and Table 2. Here too, we see a clear separation between the classes.

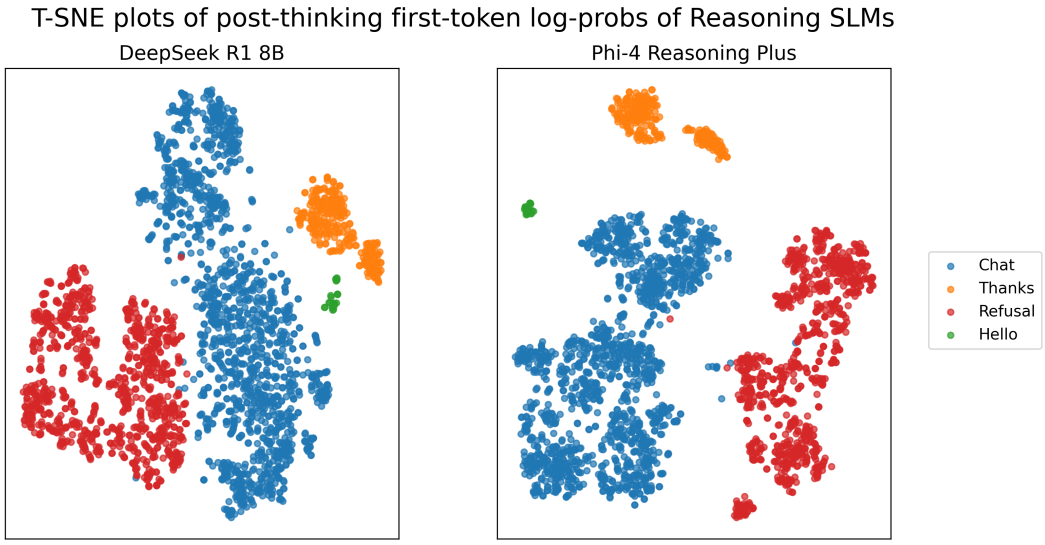

Figure 3: 2D T-SNE plot of post-empty thinking (*"<think></think>"*) first-token partial log-probabilities for Reasoning Models (DeepSeek-R1 8B, Phi-4 Reasoning Plus). Each point represents a chat, colored by class.

Table 2: Reasoning Models Performance on Type Classification (k=3)

Model	Accuracy	Precision	Recall	F1	Chat F1	Hello F1	Refusal F1	Thanks F1
Phi-4-Reasoning+	0.998	0.999	0.999	0.999	0.999	1.000	0.999	1.000
DeepSeek-R1-8B	0.998	0.998	0.989	0.993	0.999	1.000	0.999	1.000

5.3 Large Language Models

We perform an evaluation of the first-token log-probabilities of the following cloud-based models:

- OpenAI GPT-4o[2] (OpenAI et al., 2024)
- Gemini 2.0 Flash (Hassabis & Kavukcuoglu, 2024)

Unlike open-source models, these models do not provide the full log-probabilities of generated tokens, but only the top 20. We reconstructed the partial log-probabilities vectors, as the APIs of

[2]Running on Microsoft Azure

these models provide both the log-probability and token IDs (token indices) of the top 20 tokens.[3] Therefore, the vectors displayed in Figure 4 and Table 3 are *trimmed* log-probabilities vectors.

Even with the trimmed log-probabilities, we still see clear separation between the classes.

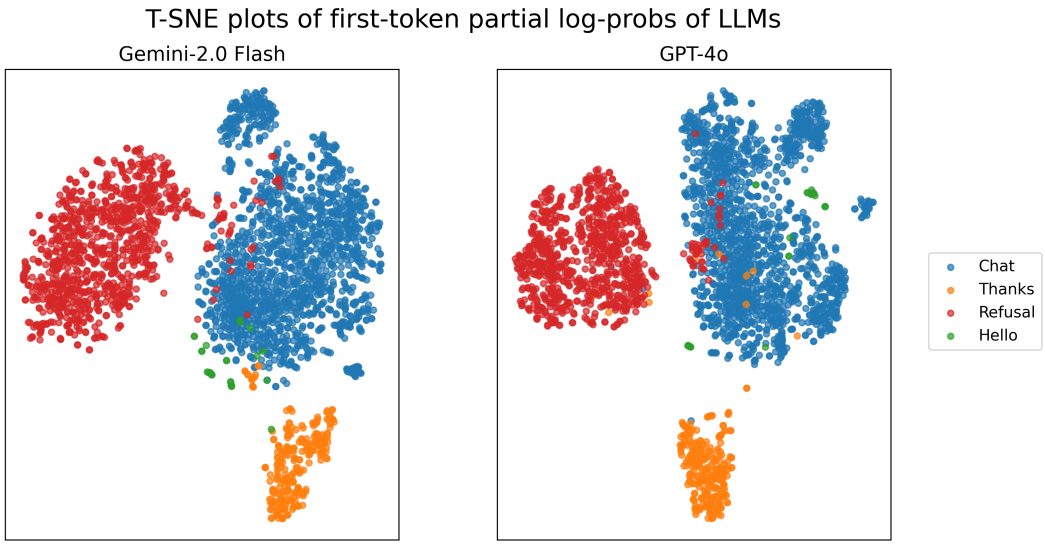

Figure 4: 2D T-SNE plot of first-token partial log-probabilities for Large Language Models (GPT-4o, Gemini-2.0 Flash). Each point represents a chat, colored by class.

Table 3: Large Language Models Performance on Type Classification (k=3)

Model	Accuracy	Precision	Recall	F1	Chat F1	Hello F1	Refusal F1	Thanks F1
Gemini-2.0-Flash	0.979	0.989	0.844	0.884	0.993	0.826	0.995	0.993
GPT-4o	0.974	0.983	0.914	0.941	0.992	0.941	0.987	0.982

6 Conclusion

In this work, we addressed the significant computational inefficiency of Large Language Models generating boilerplate responses. We introduced a simple yet highly effective method to predict the nature of an entire response by analyzing the log-probability distribution of just the first generated token.

Our comprehensive experiments across a diverse range of small, large, and reasoning-specialized models consistently demonstrated that the log-probabilities of the first token form distinct, separable clusters for different response types, such as substantive answers, refusals, simple acknowledgements or greetings. We showed that a lightweight k-NN classifier can leverage these clusters to achieve high accuracy in predicting the response category after a single generation step. Furthermore, our method successfully identifying refusals prompted by both inherent model safeguards and arbitrary, user-defined system prompts.

The primary implication of our findings is a practical and computationally trivial technique to optimize LLM inference. By enabling early termination of unwanted boilerplate generation, this approach offers substantial savings in computational cost and latency. This work presents a direct path toward more efficient, economical, and sustainable deployment of LLM systems, paving the way for more responsive and cost-effective applications.

[3]For GPT-4o, we used `tiktoken` to retrieve the token IDs from the token string-values: *github.com/openai/tiktoken*

Future work could focus on applying this technique to a wider range of boilerplate categories, multi-language scenarios, and exploring its effectiveness in multi-modal contexts.

REFERENCES

Marah Abdin, Sahaj Agarwal, Ahmed Awadallah, Vidhisha Balachandran, Harkirat Behl, Lingjiao Chen, Gustavo de Rosa, Suriya Gunasekar, Mojan Javaheripi, Neel Joshi, Piero Kauffmann, Yash Lara, Caio César Teodoro Mendes, Arindam Mitra, Besmira Nushi, Dimitris Papailiopoulos, Olli Saarikivi, Shital Shah, Vaishnavi Shrivastava, Vibhav Vineet, Yue Wu, Safoora Yousefi, and Guoqing Zheng. Phi-4-reasoning technical report, 2025. URL https://arxiv.org/abs/2504.21318.

Ahmad Aljanaideh. Speech act patterns for improving generalizability of explainable politeness detection models. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 18945–18954, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.970. URL https://aclanthology.org/2025.findings-acl.970/.

Andy Arditi, Oscar Balcells Obeso, Aaquib Syed, Daniel Paleka, Nina Rimsky, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=pH3XAQME6c.

T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967. doi: 10.1109/TIT.1967.1053964.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *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)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423/.

Dujian Ding, Ankur Mallick, Chi Wang, Robert Sim, Subhabrata Mukherjee, Victor Rühle, Laks V. S. Lakshmanan, and Ahmed Hassan Awadallah. Hybrid LLM: Cost-efficient and quality-aware query routing. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=02f3mUtqnM.

Zhichen Dong, Zhanhui Zhou, Zhixuan Liu, Chao Yang, and Chaochao Lu. Emergent response planning in llms, 2025. URL https://arxiv.org/abs/2502.06258.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry, Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi, Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini, Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivakumar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eugene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna

Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wieting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh, Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine, Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael Moynihan, Min Ma, Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Nilay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Rubenstein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu, Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti Sheth, Siim Põder, Sijal Bhatnagar, Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry, Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein Zhu, Zichuan Wei, Zoltan Egyed, Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Jean-Baptiste Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

Demis Hassabis and Koray Kavukcuoglu. Introducing gemini 2.0: Our new ai model for the agentic era. https://blog.google/technology/google-deepmind/google-gemini-ai-update-december-2024/, December 2024. Published December 11, 2024 on The Keyword (Google Blog).

Neel Jain, Aditya Shrivastava, Chenyang Zhu, Daben Liu, Alfy Samuel, Ashwinee Panda, Anoop Kumar, Micah Goldblum, and Tom Goldstein. Refusal tokens: A simple way to calibrate refusals in large language models, 2025. URL https://openreview.net/forum?id=Qny1ufReka.

Lingrui Mei, Shenghua Liu, Yiwei Wang, Baolong Bi, Ruibin Yuan, and Xueqi Cheng. Hiddenguard: Fine-grained safe generation with specialized representation router, 2025. URL https://openreview.net/forum?id=NgCNMlTXx9.

Meta AI. Llama 3.2 connect 2024: Vision, edge & mobile devices. https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/, September 2024. Meta AI blog post.

Sander Noels, Guillaume Bied, Maarten Buyl, Alexander Rogiers, Yousra Fettach, Jefrey Lijffijt, and Tijl De Bie. What large language models do not talk about: An empirical study of moderation and censorship practices, 2025. URL https://arxiv.org/abs/2504.03803.

OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Madry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler,

Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mely, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug Li, Duc Phong Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Haiming Bao, Haitang Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather Whitney, Heewoo Jun, Hendrik Kirchner, Henrique Ponde de Oliveira Pinto, Hongyu Ren, Huiwen Chang, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian O'Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quinonero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeveld, Josh Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai Fricke, Kai Hayashi, Karan Singhal, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button, Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Long Ouyang, Louis Feuvrier, Lu Zhang, Lukas Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Madelaine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljubeh, Mateusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Yatbaz, Meng Jia Yang, Mengchao Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Oom Temudo de Castro, Mikhail Pavlov, Miles Brundage, Miles Wang, Minal Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Natalie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nik Tezak, Niko Felix, Nithanth Kudige, Nitish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Philippe Tillet, Prafulla Dhariwal, Qiming Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Rapha Gontijo Lopes, Raul Puri, Reah Miyara, Reimar Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi, Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Cunninghman, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. Gpt-4o system card, 2024. URL https://arxiv.org/abs/2410.21276.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. LLMs know more than they show: On the intrinsic representation of LLM hallucinations. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=KRnsX5Em3W.

Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL https://qwenlm.github.io/blog/qwen2.5/.

Sajal Regmi and Chetan Phakami Pun. Gpt semantic cache: Reducing llm costs and latency via semantic embedding caching, 2024. URL https://arxiv.org/abs/2411.05276.

Rana Shahout, eran malach, Chunwei Liu, Weifan Jiang, Minlan Yu, and Michael Mitzenmacher. DON't STOP ME NOW: EMBEDDING BASED SCHEDULING FOR LLMS. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=7JhGdZvW4T.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

USA Today. "please, thank you": Chatgpt, openai energy costs. https://www.usatoday.com/story/tech/2025/04/22/please-thank-you-chatgpt-openai-energy-costs/83207447007/, April 2025. Accessed: 2025-08-14.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

Ziang Ye, Zhenru Zhang, Yang Zhang, Jianxin Ma, Junyang Lin, and Fuli Feng. Disentangling reasoning tokens and boilerplate tokens for language model fine-tuning. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20939–20957, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.1078. URL https://aclanthology.org/2025.findings-acl.1078/.

Zhangwenbo Zhangwenbo and Wang Yuhan. Act2P: LLM-driven online dialogue act classification for power analysis. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20494–20504, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.1052. URL https://aclanthology.org/2025.findings-acl.1052/.

Andy Zou, Zifan Wang, J. Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models, 2023.