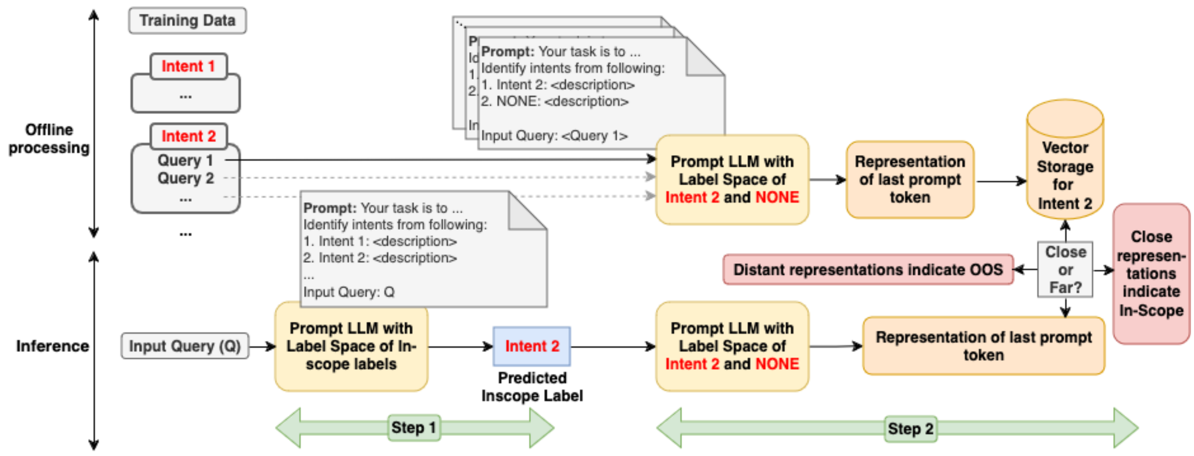

Figure 4: OOS detection using LLM's internal representations

		Overall Accuracy	F1 Score	Inscope Accuracy	Out of Scope Recall
SOF Mattress	**Mistral-7B**	0.705	0.699	0.842	0.465
	Ours	0.748	0.751	0.767	0.715
Curekart	**Mistral-7B**	0.601	0.615	0.863	0.376
	Ours	0.761	0.766	0.736	0.782
Power Play11	**Mistral-7B**	0.357	0.384	0.689	0.205
	Ours	0.780	0.739	0.411	0.950

Table 5: Comparison of our two step methodology with baseline across HINT3 datasets

representations in Step 2. But since we just need to do a forward pass for encoding the prompt, it is significantly faster than autoregressive generation.

Additionally, our proposed OOS detection methodology using LLM's internal representations can be used to improve OOS detection performance of both fine-tuned and non-fine-tuned (base instruct tuned) LLMs. We choose to experiment and show results on non-fine-tuned LLM in Sec 4.2.2 because that is a more practical scenario (as fine-tuning and deployment of a separate instance of LLM for every TODS is prohibitively expensive), but the methodology is generic enough to be used with fine-tuned LLMs as well.

4.2.2 Experiments and Results

Setup. We experiment with base instruct tuned Mistral-7B since its weights are open source. We use cosine similarity for comparing representations in Step 2 and take mean of scores over all training sentences of the predicted intent.

Results. Table 5 compares results of our methodology against baseline LLM methodology discussed in Sec 3.1.2 for HINT3 datasets. We see >5% improvement in performance across datasets at ∼300ms additional latency cost on 1 32GB V100 GPU because encoding the prompt through LLM is cheap. There is drop in in-scope performance as well but that is overcome by significant gains in OOS recall to lead to better overall performance. If needed, threshold in Step 2 of our methodology can be chosen such that drop in in-scope performance is less than an upper limit which in-turn would limit the gains in OOS performance though.

5 Conclusion

Various idiosyncrasies of intent detection task like varying scope of intents within a dataset, need to reject out of scope queries, imbalanced datasets and low resource regime make it a challenging task. In this work we evaluate multiple open source and closed source SOTA LLMs across multiple internal and external datasets for the task of intent detection using adaptive ICL and CoT prompting, compare them with SetFit models and discuss their performance/latency trade-offs. We build a hybrid system which routes queries to LLM when needed and achieves balance between performance and cost. We also propose a novel two step methodology which improves overall LLM performance by >5% across datasets and share insights on how varying scope of intents and number of labels in label space affect LLM performance. We hope our work will be useful for the community to build better TODS.

Limitations

While our current work has broad applicability for the design of accurate and computationally efficient task-oriented dialog systems, there are a few limitations:

Interactive Intent Design. Our current work assumes that intents are specified one-time in the form of examples by human experts, which has been the norm for designing task-oriented conversational assistants. However, there is potential for leveraging LLMs for an interactive class design process. In the future, we plan to investigate the benefits of enabling domain experts to directly interact with these LLMs to interactively define and refine the scope of intents.

Multilingual Support. While our current empirical evaluation was primarily focused on English datasets, the SOTA LLMs we explore already provide multilingual support. To fully harness the potential of our approach, we aim to generalize our ideas to the multilingual setting and evaluate them on diverse dialog datasets across various languages.

Alternative Hybrid Strategies. In the current work, we employ a cascade routing strategy that uses SetFit's uncertainty to combine the SetFit models and LLMs yielding promising results. However, there are additional hybrid strategies worth exploring. Drawing inspiration from active learning literature, we could investigate alternative utility functions, such as information gain to determine when to invoke the LLM alongside the SetFit model. We also plan to compare our approach with model distillation strategies, where the LLM is used to generate synthetic training data to enhance the SetFit models.

Ethics Statement

Our motivation for the current work is to develop computationally efficient and accurate solutions for intent detection, leveraging prior research on sentence transformers and generative language models. As the focus is on intent classification rather than generation, the typical risks associated with generative content do not directly apply. However, as with any machine learning system, there are other important considerations, such as potential biases in the training data or constituent pre-trained models, the possibility of misuse, and challenges in establishing full accountability. Since our approach incorporates generative LLMs, any application of the proposed ideas needs to be mindful of any biases present in those models. Overall, the proposed methodological innovations are intended for benign applications and are not associated with any direct negative social impact. The datasets used in this research include public benchmarks and proprietary datasets from safe ecommerce categories, with personally identifiable information (PII) redacted to ensure customer privacy. To enable reproducibility, we plan to share these datasets as a community after internal approvals.

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2623–2631.

Anthropic. 2023. Welcome to claude. https://docs.anthropic.com/claude/docs/intro-to-claude. Accessed: 30-04-2024.

Gaurav Arora, Chirag Jain, Manas Chaturvedi, and Krupal Modi. 2020. HINT3: Raising the bar for intent detection in the wild. In *Proceedings of the First Workshop on Insights from Negative Results in NLP*, pages 100–105, Online. Association for Computational Linguistics.

BAAI. 2023. Bge retriever. https://huggingface.co/BAAI/bge-base-en-v1.5. Accessed: 30-04-2024.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. 2020. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 38–45, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Dialogflow. 2010. Intent in dialogflow. https://cloud.google.com/dialogflow/cx/docs/concept/intent. Accessed: 11-07-2024.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. *Preprint*, arXiv:1506.02142.

Gartner. 2022. Gartner predicts conversational ai will reduce contact center agent labor costs by $80 billion in 2026. https://www.gartner.com/en/newsroom/press-releases/2022-08-31-gartner-predicts-conversational. Accessed: 11-07-2024.

Maarten Grootendorst. 2020. Keybert: Minimal keyword extraction with bert.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. 2019. An evaluation dataset for intent classification and out-of-scope prediction. 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)*, pages 1311–1316, Hong Kong, China. Association for Computational Linguistics.

AWS LEX. 2017. Intent in aws lex. https://docs.aws.amazon.com/lex/latest/dg/how-it-works.html. Accessed: 11-07-2024.

Bing Liu and Ian R. Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. *CoRR*, abs/1609.01454.

Bo Liu, Liming Zhan, Zexin Lu, Yujie Feng, Lei Xue, and Xiao-Ming Wu. 2024. How good are llms at out-of-distribution detection? *Preprint*, arXiv:2308.10261.

Meta. 2023. Meta llama. https://llama.meta.com/. Accessed: 11-07-2024.

Mistral. 2023. Mistral ai models. https://docs.mistral.ai/getting-started/models/. Accessed: 11-07-2024.

OpenAI. 2022. Introducing chatgpt. https://openai.com/index/chatgpt. Accessed: 30-04-2024.

OpenAI, Josh Achiam, and Steven Adler et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. 2023. Is chatgpt a general-purpose natural language processing task solver? *Preprint*, arXiv:2302.06476.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *arXiv preprint arXiv:2004.09297*.

Sentence Transformers. 2021. Sentence transformer mpnet base v2. https://huggingface.co/sentence-transformers/all-mpnet-base-v2. Accessed: 10-07-2024.

Lewis Tunstall, Nils Reimers, Unso Eun Seo Jo, Luke Bates, Daniel Korat, Moshe Wasserblat, and Oren Pereg. 2022a. Efficient few-shot learning without prompts. *Preprint*, arXiv:2209.11055.

Lewis Tunstall, Nils Reimers, Unso Eun Seo Jo, Luke Bates, Daniel Korat, Moshe Wasserblat, and Oren Pereg. 2022b. Setfit - efficient few-shot learning with sentence transformers. https://github.com/huggingface/setfit. Accessed: 10-07-2024.

Pei Wang, Keqing He, Yejie Wang, Xiaoshuai Song, Yutao Mou, Jingang Wang, Yunsen Xian, Xunliang Cai, and Weiran Xu. 2024. Beyond the known: Investigating llms performance on out-of-domain intent detection. *Preprint*, arXiv:2402.17256.

Haode Zhang, Yuwei Zhang, Li-Ming Zhan, Jiaxin Chen, Guangyuan Shi, Xiao-Ming Wu, and Albert Y. S. Lam. 2021. Effectiveness of pre-training for few-shot intent classification. *CoRR*, abs/2109.05782.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A survey of large language models. *Preprint*, arXiv:2303.18223.

A Appendix

A.1 AID3 Dataset

ALC contains upper funnel shopping queries for 1 HCTP[7] category while **ADP** contains lower funnel queries for 6 HCTP categories. **OADP** also contains lower funnel queries from >10 HCTP categories.

A.2 Experiment Setup

For training SetFit models, we use SetFit library (Tunstall et al., 2022b) for implementation. Hyperparameter search space for SetFit model's training is given in Table 6.

For **negative augmentation**, we use KeyBERT (Grootendorst, 2020) for identifying keywords. For every identified keyword, random 50% of the times we completely remove it, and remaining 50% of the times we replace it with a randomly generated string of 5 characters. For eg: "looking for a gaming laptop" can get converted into "looking for a" or "looking for a XYCVD QSDER" or "looking for a RTYUH". Since these augmented OOS sentences have similar lexical pattern as in-scope training sentences, these are expected to help the model avoid latching onto any spurious patterns and help overall learning, which shows up in results as well (See 3.4). If U is the set of randomly sampled augmentations to add to train set, then we keep $|U| = 0.2*|D|$, where $|D|$ is size of train set.

For **choosing ICL examples** for LLMs, we do grid search over ideal number of ICL examples and retriever threshold whose search space is shown in Table 7. We keep ordering of labels in the prompt

[7] High Consideration Technical Products