Controllable Conversational Theme Detection Track at DSTC 12

Igor Shalyminov, Hang Su, Jake Vincent, Siffi Singh, Jason Cai, James Gung, Raphael Shu, Saab Mansour

Amazon

shalymin@amazon.com

Abstract

Conversational analytics has been on the forefront of transformation driven by the advances in Speech and Natural Language Processing techniques. Rapid adoption of Large Language Models (LLMs) in the analytics field has taken the problems that can be automated to a new level of complexity and scale.

In this paper, we introduce *Theme Detection* as a critical task in conversational analytics, aimed at automatically identifying and categorizing topics within conversations. This process can significantly reduce the manual effort involved in analyzing expansive dialogs, particularly in domains like customer support or sales. Unlike traditional dialog intent detection, which often relies on a fixed set of intents for downstream system logic, themes are intended as a direct, user-facing summary of the conversation's core inquiry. This distinction allows for greater flexibility in theme surface forms and user-specific customizations.

We pose *Controllable Conversational Theme Detection* problem as a public competition track at Dialog System Technology Challenge (DSTC) 12 — it is framed as joint clustering and theme labeling of dialog utterances, with the distinctive aspect being controllability of the resulting theme clusters' granularity achieved via the provided user preference data.

We give an overview of the problem, the associated dataset and the evaluation metrics, both automatic and human. Finally, we discuss the participant teams' submissions and provide insights from those. The track materials (data and code) are openly available in the GitHub repository.

1   Introduction

Conversational analytics — at the intersection of Speech and Natural Language Processing — has undergone rapid transformation due to advances in both fields. *Automatic Speech Recognition (ASR)* now enables accurate transcription of conversations across diverse domains and durations. Simultaneously, *Natural Language Processing* (especially *Information Retrieval*) has enabled large-scale analysis of conversational data, revealing patterns such as word usage, emotional tone, and discussed topics. More recently, *Large Language Models (LLMs)* have elevated the complexity and quality of analysis tasks. For instance, large-scale text embedding models (Wang et al., 2024) significantly enhance document similarity search by capturing semantic meaning beyond surface forms.

In this paper, we propose the task of *Theme Detection*, a key problem in conversational analytics. Themes reflect the high-level topics discussed in conversations and aid in categorizing them by function — e.g., customer support, sales, or marketing. Automatically identifying and labeling themes can greatly reduce the manual effort required to analyze long conversations.

While related to dialog intent detection, theme detection serves a different purpose. Intents are typically tied to a fixed schema and used for downstream system logic. In contrast, themes are final outputs for users (e.g., analysts), summarizing the customer's inquiry and supporting diverse surface forms and customizations.

We introduce the task of *Controllable Conversational Theme Detection* as a new track in the Dialog System Technology Challenge (DSTC) 12. Building on the DSTC 11 track on Open Intent Induction (Gung et al., 2023b), our challenge adds two major innovations: (1) joint theme detection and labeling, and (2) controllable theme granularity. The latter enables customization of theme clusters based on user preferences — motivated by real-world use cases where businesses may want finer or coarser thematic distinctions.

This task is designed for a zero-shot setting on unseen domains. Models will be guided by user preference data (detailed in Section 4) to align both labels and cluster granularity. While especially

compelling in the context of LLMs, the proposed setup does not require their use.

2 Related Work

In this section, we discuss prior work related to the distinctive aspects of our proposed task.

2.1 Unsupervised dialog theme / intent detection

The task of open conversational intent induction was introduced in a DSTC 11 track by Gung et al. (2023b), which focused on utterance clustering in two setups of varying complexity: (1) intent detection with pre-defined intentful utterances to be clustered, and (2) open intent induction, which required identifying and clustering such utterances.

In contrast, our task involves a single setup with pre-defined themed utterances, and the goal is to jointly cluster and label them according to specific evaluation metrics. Unlike intent induction, we do not restrict the surface form of theme labels. Instead, labels are assessed based on their structural quality and functional usefulness for analysis (see Section 6 and Appendix B).

2.2 Controllable clustering

Our goal of controllable theme granularity builds on the concept of *constrained clustering*. A comprehensive taxonomy of constraint-based clustering tasks is provided by González-Almagro et al. (2025). We adopt instance-level pairwise constraints ("should-link" / "cannot-link"), implementing a semi-supervised clustering approach where supervision comes from labeled utterance pairs. This setup has been well-studied, from early work by Basu et al. (2004) to more recent approaches by Zhang et al. (2019) and Viswanathan et al. (2024).

2.3 Clustering with LLMs

The use of LLMs for utterance clustering has gained traction. Zhang et al. (2023) propose using hard triplets ("does A match B better than C?") derived from a teacher LLM to fine-tune a smaller embedding model and refine clusters via a hierarchical method similar to HAC (Manning et al., 2008). While this method enables controllable clustering guided by LLMs, it focuses solely on clustering — cluster labeling remains out of scope. In contrast, our task requires *labeled theme clusters*, combining clustering with label generation to better reflect real-world needs.

Viswanathan et al. (2024) provide a thorough study on integrating LLMs into clustering workflows. They identify three points of intervention: (1) *pre-clustering*, using LLMs to generate keywords and enrich input texts; (2) *during clustering*, by expanding human-provided pairwise constraints; and (3) *post-clustering*, correcting uncertain assignments with LLM-based prompting. Although their framework aligns well with our goals, their focus remains on clustering rather than labeling.

3 Task Description

The task of *Controllable Conversational Theme Detection* is defined as follows. The input data are:

1. a dataset of conversations with some utterances within them labeled as "themed" (those conveying the customer's requests, possibly several per conversation)

2. a set of preference pairs covering a sample of all the themed utterances and representing what pairs should belong to the same theme and which should not. — which we refer to as "should-link" and "cannot-link" pairs, respectively

3. a theme label writing guideline outlining the requirements to a label as both a linguistic expression and an analytical tool.

The goal of the task is to:

- cluster the themed utterances so that each cluster represents a meaningful semantic / thematic group, is distinguishable from other theme clusters and satisfies the should-link / cannot-link requirements on its utterances (if it contains utterances included in the preference data)

- give each theme a short, concise and actionable natural language label (more detail on our evaluation criteria is given in Section 6).

3.1 Controlling theme granularity

In the way we intend to control theme granularity, we loosely follow the Stage 2 approach of Zhang et al. 2023. That work described a data-efficient approach with user preference data in the should-link / cannot-link form. As such, if user preferences indicate that the utterances *"I want to purchase pet*
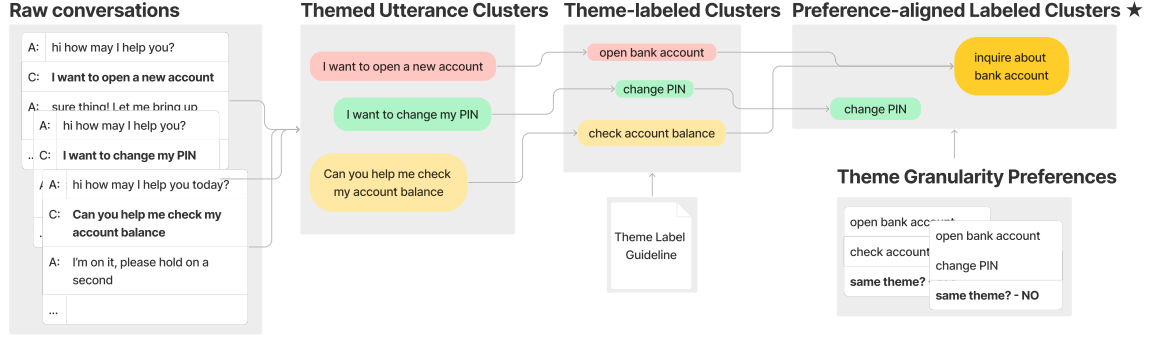
Figure 1: Diagram of the proposed task in the form of an example processing pipeline. The inputs to the "system" are raw conversations, user preferences on the theme granularity and theme label guidelines; the output is preference-aligned utterance clusters with the corresponding theme labels (marked with ★)

insurance" and *"I want to purchase travel insurance"* should belong to the same theme, all the utterances like these two would be associated to the single theme whose label semantically unifies both of the two utterances' meanings e.g. *"purchase insurance"* or some close paraphrase of it. On the other hand, if the preferences elicit that *"I want to find the closest branch"* and *"Give me the directions to the closest ATM"* should not belong to the same theme, the corresponding themes *"find branch"* and *"find ATM"* as well as the clusters of utterances belonging to them should be kept as separate. Some example usages of such data include contrastive fine-tuning of utterance representation as done by e.g. Chu et al. (2023) and Zhang et al. (2021) or adjusting the initial clusters/themes, as depicted in Figure 1.

3.2 Expected result

A successful completion of the task would assume assigning each utterance a theme label so that:

- theme labels are concise, exhaustively cover all the examples and are mutually exclusive,

- label wording conforms to the Theme label writing guideline (Appendix B),

- theme granularity matches the 'gold' held-out assignment which is supposed to be inferred from the provided user preference samples.

A visualization of the overall task is presented in Figure 1 where we depict a potential sequential pipeline as an example. The actual submissions can vary in architecture and the types of models used. We intend the problem to be solved in a zero-shot weakly supervised way, in the sense that all the training/development data provided to the participants has no domain overlap with the test data (more detail on the data in Section 4), and the only supervision signals provided are 1) user preference data covering a sample of the dataset and 2) theme label writing guideline.

While the input data suggests LLM-based solutions, we encourage the participants to use techniques from both LLM-based and traditional Machine Learning paradigms that adequately correspond to the problem specifics.

4 Data

We build our task on top of the NatCS (Gung et al., 2023a,b), a multi-domain dataset of human-human customer support conversations — the dataset statistics per domain are provided in Table 2.

We intend for the participants' submissions to work in a zero-shot setup naturally supported within the LLM-centered framework. As such, we provide the three original NatCS domains: **Banking**, **Finance** and **Insurance** — for the participants to use for the training/development purposes and assess the domain generalization of their approaches.

Our theme labels closely resemble the original intent annotations in NatCS, though those were altered in the following ways:

1. intent labels' surface form was rewritten where needed to conform with the theme label writing guideline (see Appendix B),

2. for each original intent label, we provide two theme labels, a more specific one and a more vague one, for the flexibility of evaluation,

Table 1: User Preference Data Statistics

Domain	# Should-link pairs	% data covered	# Cannot-link pairs	% data covered
Banking	164	10.04%	164	10.04%
Finance	173	10%	173	10%
Insurance	155	8.99%	126	7.30%
Travel (held out)	77	10.07%	76	9.93%

Table 2: Dialog Dataset Statistics

Domain	# Dialogs	# Themed utterances
Banking	980	1634
Finance	3000	1725
Insurance	836	1333
Travel (held out)	999	765

3. intent clustering itself was altered to reflect our task's custom theme granularity,

4. some noisy intent annotations were corrected or otherwise dropped.

The held out test domain, **Travel**, is publicly released for the first time in this challenge and has little to no overlap with the train/dev data.

Also introduced in this challenge is theme granularity preference data on top of NatCS dialogs, its statistics are shown in Table 1. We generated preference pairs in the following way.

Should-link pairs: we clustered themed utterances (we leave the specifics of the clustering algorithm behind to prevent evaluation metric hacking) and sampled pairs that belong to the same cluster in the gold assignment but to the different clusters as per the algorithm, with sampling weights set to the normalized cosine distances between the points in the pair (further points that should be in the same theme are more interesting). **Cannot-link pairs**: similarly, we sampled pairs of utterances that belong to different clusters in the gold assignment but to the same cluster as per the algorithm. Sample weights set to $1 - dist(utt_a, utt_b)$ normalized to make a probability distribution, where utt_a and utt_b are the utterances in the pair and $dist$ is cosine distance.

In each case, our target amount of pairs to generate corresponds to 10% of all the themed utterances in the dataset, and preference pairs cover no more than 30% of any given gold cluster's utterances.

5 Baseline and Experimental Setup

We provided the participants with a baseline solution that combines traditional machine learning approaches with LLM-based techniques. As such, the entire baseline workflow consists of 3 stages:

1. **Utterance clustering**. Each themed utterance is embedded with SentenceBERT (`all-mpnet-base-v2` model is used, Reimers and Gurevych 2019), then the embeddings are clustered using the K-means algorithm (Jin and Han, 2010) with 10 clusters by default and the `k-means++` initializer (Arthur and Vassilvitskii, 2007).

2. **Theme cluster adjustment to user preferences**. We apply a naïve algorithm that re-assigns cluster labels for every utterance id containing in the should-link / cannot-link sets. For every $< utt_i, utt_j >$ pair in the should-link set, if they are assigned to different clusters, utt_j is re-assigned to utt_i's cluster. In turn, for every $< utt_m, utt_n >$ in the cannot-link set, utt_n is re-assigned to the cluster with the second closest centroid to it. Evidently, the baseline cluster adjustment algorithm doesn't have any generalization outside of the given preference sets.

3. **Theme label generation**. We used an LLM with the prompt as in Appendix B — the default model used in the baseline implementation is `Mistral-7B-Instruct-v0.3` (Jiang et al., 2023). No limitation on the number of in-context utterances was set.

6 Evaluation

Theme assignment that is the result of our task's solution can be assessed from two perspectives: from the controlled clustering perspective and from the theme label generation perspective — our evaluation metrics reflect these two perspectives.

6.1 Automatic evaluation

Automatic evaluation metrics are mainly used for the development purposes and were provided to the participants as part of the starter code.

Table 3: Automatic Evaluation — Theme Label Metrics. Here and below, results in **bold** are the best, underlined are those above baseline.

Team ID	R-1	R-2	R-L	Cos sim	BERT P	BERT R	BERT F1	LLM s1	LLM s2	LLM avg
Team A	32.70%	4.60%	29.82%	59.51%	89.82%	91.20%	90.35%	46.01%	56.47%	51.24%
Team B	5.03%	0.00%	5.03%	37.08%	85.22%	88.02%	86.53%	12.03%	0.13%	6.08%
Team C	**45.22%**	23.81%	**45.10%**	**69.91%**	**95.02%**	**94.69%**	**94.71%**	**100.00%**	**99.48%**	**99.74%**
Team D	34.57%	21.31%	34.27%	55.93%	92.52%	91.48%	91.91%	80.39%	76.60%	78.50%
Team E	42.28%	16.50%	41.22%	62.48%	93.85%	92.84%	93.27%	93.46%	95.69%	94.58%
Team F	23.10%	0.79%	21.14%	46.02%	85.67%	89.29%	87.19%	4.05%	3.53%	3.79%
Baseline	43.74%	**24.56%**	42.87%	59.68%	89.25%	89.87%	89.52%	20.39%	39.48%	29.93%
BL-prefs	29.27%	4.21%	24.69%	48.79%	85.31%	87.77%	86.44%	12.81%	18.43%	15.62%

6.1.1 Clustering metrics

- **NMI** score (Vinh et al., 2010) — *Normalized Mutual Information* is a function that measures the agreement of the two cluster assignments, reference and predicted, ignoring permutations. Normalization is performed over the mean of the entropies of the two assignments

- **ACC** score (Huang et al., 2014) evaluates the optimal alignment between the reference cluster assignment and the predicted one, with the alignment obtained using the Hungarian algorithm.

6.1.2 Label generation metrics

We evaluate the predicted labels for theme clusters in two general ways: 1) similarity to the reference labels, 2) adherence to the theme label guideline.

Similarity of a predicted label to the references is calculated in the following way:

$$Score_i(Y_i, \hat{y}_i) = \max_j sim(Y_{i,j}, \hat{y}_i)$$

where Y_i are the reference labels for the i-th utterance (we provide two labels with a more specific and a more vague wording, respectively), y_i is the predicted label for the same utterance and sim is one of the similarity functions listed below.

- **Cosine similarity** — the semantic similarity measure over SentenceBERT embeddings (`all-mpnet-base-v2` model is used, Reimers and Gurevych 2019) of the reference and predicted labels,

- **ROUGE** score (Lin, 2004) — an token-level N-gram overlap metric useful for comparing short and concise word sequences,

- **BERTScore** (Zhang et al., 2020) combines the agility of embedding-based similarity and the interpretability of token-level overlap. The model tokenizes each utterance and generates a contextual embedding for each token. Then, a cosine similarity $sim_{i,j}$ is calculated between i-th token of the reference and j-th token of the prediction. We report BERTScore **Precision** (for each token in the prediction, finding the reference token with the highest similarity), **Recall** (for each token in the reference, finding the prediction token with the highest similarity) and **F1 score**.

Adherence to the guideline is evaluated with an LLM-as-a-Judge prompted with a version of the guideline attached in the Appendix B (it was provided for the participants). For the usage with the LLM, it was split into three sections spanning structural and functional criteria, i.e. how good the label is as a linguistic expression and how good it is as an analytical tool, respectively. For the sake of preventing evaluation metric hacking, we shared a different / condensed version of the guideline to the participants and kept the full version held out. For evaluation during the development phase, our provided code used a self-hosted solution with `vicuna-13b-v1.5` (Zheng et al., 2023) as the default LLMaaJ backbone. In the automatic evaluation of the final submissions, we used Claude 3.5 Sonnet (Anthropic, 2024). Our repository contains both the public version of the label style evaluation prompt (3 condensed sections optimized for usage with public self-hosted LLMs) and its held out version (2 expanded sections optimized for usage with Claude, uploaded after the end of the competition).

6.2 Human Evaluation

All submissions underwent expert human evaluation in order to verify automated evaluation results

and to expand the automated evaluation methodology to more precisely assess each solution's performance. The evaluation dimensions were divided into two broad categories covering formal and functional criteria, and each of these areas had additional subdimensions to be rated by evaluators in a binary fashion (*pass/fail*) using criteria distributed into into two broad categories: Structural/Functional. The structural criteria were based on the theme labeling guidelines provided to participants.

Structural Criteria (Theme Label as a Linguistic Expression): Conciseness & Word Choice, Grammatical Structure

Functional Criteria[1] (Theme Label as an Analytical Tool): Semantic Relevance, Analytical Utility, Granularity, Actionability, Domain Relevance, Thematic Distinctiveness.

The guidelines for each of these dimensions, along with the positive and negative examples provided to evaluators (with reasoning), are laid out in Appendix C. The theme labeling guidelines, upon which the structural criteria were based, are defined in Appendix B. The annotation task was completed in a single-pass way by two members of the track organizing team.

7   Results and Analysis

Table 6: Automatic Evaluation — Clustering Metrics

Team ID	ACC	NMI
Team A	48.37%	42.02%
Team B	17.91%	1.97%
Team C	**67.97%**	**70.39%**
Team D	51.76%	47.71%
Team E	35.82%	47.73%
Team F	26.67%	9.06%
Baseline	53.2%	50.59%
BL-prefs	47.97%	45.39%

We received submissions from 6 participant teams. During the development, the teams were free to use the provided public data across 3 domains for creating their own train / development setups and testing e.g. out-of-domain generalization of their approaches. The test domain was made public during the last week of the competition. When submitting the inference results via an online form, the participant teams were asked to provide a brief info about their approaches. Below are the questions and the summaries of the submitted answers:

What LLM type did you use? (*Open-source — self-hosted / Proprietary via API / No LLM / Other*)

Teams A, C and F used a proprietary API; teams B, D and E used an open-source self-hosted LLM.

How large of an LLM did you use? (*<30B / 30—100B / >100B / Unknown (proprietary API) / No LLM / Other*)

Team A, C and F's model size is unknown; teams B, D and E used a model with <30B parameters.

Did you use any conversational information (previous / past context of the utterance)? Please specify if yes

Team C used the context window of 5 turns; Team E used conversational context within the predicted topic segment.

What clustering algorithm did you use?

Team A used HDBScan (Campello et al., 2013); Teams B and D used K-Means (Jin and Han, 2010); Team C used ClusterLLM (Zhang et al., 2023); Team F experimented with K-Means, DBSCAN and HDBSCAN; Team E used Spectral Clustering (Shi and Malik, 2000).

What text embedding model did you use?

Teams A and C used Instructor model (Su et al., 2023); Teams B, D and E used SentenceBERT as per the baseline.

Did you use an embedding dimensionality reduction technique? (Please specify which one if yes)

Teams A and E used UMAP (McInnes et al., 2018).

Did you use a data augmentation technique (please specify what kind)?

Team A used Speech Acts as a data augmentation; Team B used SimCSE (Gao et al., 2021); Team E used contrastive learning to augment the limited unlabeled data.

How did you use the should-link / cannot-link pairs?

Teams A, B and D used the baseline approach. Team C used an LLM to re-assign the clusters for all the utterances from the should-link pairs. For

---

[1]All functional criteria dimensions were evaluated at the level of the utterance except for *Thematic Distinctiveness*, which was evaluated for each cluster label.

Table 4: Human Evaluation — Per-utterance Functional Metrics

Team ID	Semantic Relevance	Analytical Utility	Granularity	Actionability	Domain Relevance
Team A	77.25%	63.66%	22.75%	56.21%	79.74%
Team B	64.97%	12.94%	0.00%	4.05%	97.78%
Team C	**89.67%**	**82.75%**	47.84%	**74.77%**	**98.82%**
Team D	68.76%	63.66%	26.41%	60.26%	94.25%
Team E	86.27%	54.64%	22.48%	54.51%	91.11%
Team F	45.23%	41.57%	7.71%	41.57%	67.45%
Baseline	86.61%	66.84%	**47.98%**	66.84%	89.6%

Table 5: Human Evaluation — Per-cluster Metrics

Team ID	Structural		Functional
	Conciseness	Grammatical Structure	Thematic Distinctiveness
Team A	83.33%	**100.00%**	75.76%
Team B	**100.00%**	33.33%	0.00%
Team C	**100.00%**	**100.00%**	**91.11%**
Team D	91.67%	66.67%	90.91%
Team E	93.65%	93.65%	78.34%
Team F	95.00%	**100.00%**	72.63%
Baseline	80%	30%	**91.11%**

the cannot-link pairs, the LLM was used to identify the utterance of the pair not belonging to the cluster, and then to make the re-assignment. Team E trained a reward model from the should-link and cannot-link pairs that was later incorporated into the clustering algorithm to impose soft constraints.

Did you use the theme label styleguide — if yes, how?

Team C used General Schema to extract verbs and nouns for each utterance in the cluster, then using those, they generated theme labels. Theme D instructed the labeling LLM to generate Verb-Object pairs. Teams E and F used the provided styleguide itself. Team F added it directly into the labeling LLM's prompt, and Team E modified and simplified it first.

Short (1-2 paragraph) description of your approach

Team A proposed a cluster-then-label framework for thematic clustering of utterances. First, they compute utterance embeddings using either Sentence Transformers, InBedder, or Instructor models depending on the embedding type. they then apply clustering (KMeans or HDBSCAN with UMAP-based dimensionality reduction) to group thematically similar utterances. Clustering is refined using manually provided should-link and cannot-link preference pairs, ensuring better alignment with human notions of similarity. After clustering, each

cluster is labeled automatically by prompting an LLM (ChatGPT or Gemini Flash) with a batch of utterances, extracting a theme label and brief explanation. The resulting predicted labels are assigned back to utterances, forming the final output for evaluation. This approach leverages both unsupervised structure discovery and lightweight LLM-based supervision for scalable and interpretable theme labeling.

Team B used SCCL (Zhang et al., 2021) and applied SimCSE for data augmentation. After training the SCCL, they clustered the utterances with K-Means. They performed hyperparameter search for the number of clusters based on the Silhouette score and set it to 7. User preference data was not used.

Team C first extracted keyphrases from conversations using an LLM. They also determined the numbers of clusters based on the Silhouette coefficient. Clustering was performed using ClusterLLM, and the embedder was fine-tuned on the clustered utterances. Subsequently, among the two candidates with the highest preference pair accuracy, the candidate with the greater number of clusters was selected as the final model. Utterances were then adjusted according to the preference pairs. Finally, for the clustered utterances, a general schema was extracted in terms of verbs and nouns, and based on both the schema and the utterance content, the final theme labels were generated.

Team D explored two approaches. The first one involved designing a prompting strategy to generate concise labels in a Verb–Object format (e.g., *"update address"*, *"book flight"*), allowing for more structured and comparable cluster representations. The second approach used `LLaMA-3.1-8B-Instruct` to evaluate whether two utterances (with dialog history) belonged to the same cluster, based on their distance from the cluster center. The second method showed limited performance, and they submitted results using the first one, with a more robust prompting-based labeling strategy.

Team E propose PrefSegGen, a preference-aware topic segmentation and generation framework that addresses low-resource conversational theme understanding by integrating topical-structured context modeling with user-preference-aligned theme generation. First, they introduce a novel two-stage self-supervised contrastive learning topic segmentation framework to obtain the topic segment to which the target utterance belongs under low-resource conditions. It initially leverages the unlabeled dialogues to pre-train topic encoders (`bert-base-uncased` & `sup-simcse-bert-base-uncased`) on coherence and similarity patterns, followed by supervised fine-tuning with minimal labeled data to enhance segmentation precision. Subsequently, they incorporate a reward-guided clustering mechanism to guarantee that the generated themes are both contextually grounded and preference-aligned. A reward model, trained on should-link and cannot-link pairs, dynamically assigns linkage weights that reflect semantic proximity in line with user expectations. These weights guide spectral clustering after UMAP-based embedding reduction. Crucially, for each target utterance, they utilize its segmented topical context as input when prompting `LLaMA3-8B-Instruct`, coupled with the official style guide, to generate hierarchical theme labels. An ensemble refinement process further enhances topic consistency by filtering low-frequency labels, yielding final outputs that are structurally coherent, context-aware, and tailored to user preferences.

Team F employed a large language model (LLM) to annotate utterances based on preference signals, and subsequently attempted to merge clusters according to the LLM-based annotation.

Our evaluation results reveal that **Team C**'s approach achieved the highest accuracy across the board on both human and automatic metrics. It was tied with **Team B** on Label Conciseness, and with **Teams A** and **F** on Grammatical Structure. Although only **Team C**'s approach achieves **100%** on both, signifying that its label generation works in full accordance with the styleguide. **Team C** was also the only one to surpass the baseline on automatic clustering metrics. **Team E** achieved the second best overall performance in both automatic and human evaluation and **Team D** placed third.

It is noteworthy that for all the three winning places, the ranking induced by the automatic metrics matched that by the humans — indicating that 1) automatic similarity metrics are applicable for short text, and 2) automatic evaluation of higher-level concepts like our label guideline is sufficiently accurate with frontier LLMs-as-Judges.

8   Conclusions

In this paper, we introduced *Theme Detection* as a critical task in conversational analytics, and the associated *Controllable Conversational Theme Detection* competition track at Dialog System Technology Challenge (DSTC) 12 — where joint theme clustering and cluster label generation was further combined with the custom theme cluster granularity controllable via the provided preference data.

We gave an overview of the competition setup including the problem, the benchmark dataset and the details of evaluation, both automatic and human. We presented the participant team's submissions and gave an analysis of the insights from those.

We hope that this new problem, together with the dataset and the insights obtained from the competition will foster further research and advancements in Conversational AI.

9   Acknowledgements

References

Anthropic. 2024. Introducing Claude 3.5 Sonnet. https://www.anthropic.com/news/claude-3-5-sonnet. Accessed: 2025-06-13.

David Arthur and Sergei Vassilvitskii. 2007. k-means++: the advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM*

Symposium on Discrete Algorithms, SODA '07, page 1027–1035, USA. Society for Industrial and Applied Mathematics.

Sugato Basu, Arindam Banerjee, and Raymond J. Mooney. 2004. Active semi-supervision for pairwise constrained clustering. In *Proceedings of the Fourth SIAM International Conference on Data Mining, Lake Buena Vista, Florida, USA, April 22-24, 2004*, pages 333–344. SIAM.

Ricardo J. G. B. Campello, Davoud Moulavi, and Joerg Sander. 2013. Density-based clustering based on hierarchical density estimates. In *Advances in Knowledge Discovery and Data Mining*, pages 160–172, Berlin, Heidelberg. Springer Berlin Heidelberg.

Caiyuan Chu, Ya Li, Yifan Liu, Jia-Chen Gu, Quan Liu, Yongxin Ge, and Guoping Hu. 2023. Multi-stage coarse-to-fine contrastive learning for conversation intent induction. In *Proceedings of The Eleventh Dialog System Technology Challenge*, pages 31–39, Prague, Czech Republic. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Germán González-Almagro, Daniel Peralta, Eli De Poorter, José Ramón Cano, and Salvador García. 2025. Semi-supervised constrained clustering: an in-depth overview, ranked taxonomy and future research directions. *Artif. Intell. Rev.*, 58(5):157.

James Gung, Emily Moeng, Wesley Rose, Arshit Gupta, Yi Zhang, and Saab Mansour. 2023a. NatCS: Eliciting natural customer support dialogues. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9652–9677, Toronto, Canada. Association for Computational Linguistics.

James Gung, Raphael Shu, Emily Moeng, Wesley Rose, Salvatore Romeo, Arshit Gupta, Yassine Benajiba, Saab Mansour, and Yi Zhang. 2023b. Intent induction from conversations for task-oriented dialogue track at DSTC 11. In *Proceedings of The Eleventh Dialog System Technology Challenge*, pages 242–259, Prague, Czech Republic. Association for Computational Linguistics.

Peihao Huang, Yan Huang, Wei Wang, and Liang Wang. 2014. Deep embedding network for clustering. In *2014 22nd International Conference on Pattern Recognition*, pages 1532–1537.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Xin Jin and Jiawei Han. 2010. *K-Means Clustering*, pages 563–564. Springer US, Boston, MA.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

L. McInnes, J. Healy, and J. Melville. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *ArXiv e-prints*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Jianbo Shi and J. Malik. 2000. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2023. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1102–1121, Toronto, Canada. Association for Computational Linguistics.

Nguyen Xuan Vinh, Julien Epps, and James Bailey. 2010. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research*, 11(95):2837–2854.

Vijay Viswanathan, Kiril Gashteovski, Kiril Gashteovski, Carolin Lawrence, Tongshuang Wu, and Graham Neubig. 2024. Large language models enable few-shot clustering. *Transactions of the Association for Computational Linguistics*, 12:321–333.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Improving text embeddings with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11897–11916, Bangkok, Thailand. Association for Computational Linguistics.

Dejiao Zhang, Feng Nan, Xiaokai Wei, Shang-Wen Li, Henghui Zhu, Kathleen McKeown, Ramesh Nallapati, Andrew O. Arnold, and Bing Xiang. 2021. Supporting clustering with contrastive learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5419–5430, Online. Association for Computational Linguistics.

Hongjing Zhang, Sugato Basu, and Ian Davidson. 2019. A framework for deep constrained clustering - algorithms and advances. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2019, Würzburg, Germany, September 16-20, 2019, Proceedings, Part I*, volume 11906 of *Lecture Notes in Computer Science*, pages 57–72. Springer.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Yuwei Zhang, Zihan Wang, and Jingbo Shang. 2023. ClusterLLM: Large language models as a guide for text clustering. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13903–13920, Singapore. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

A Cluster Labeling Prompt

```
<task>
You are an expert call center assistant.
You will be given a set of utterances in
<utterances> </utterances> tags, each
one on a new line.
The utterances are part of call center
conversations between the customer and
the support agent.
Your task is to generate a short label
describing the theme of all the given
utterances. The theme label should be
under 5 words and describe the desired
customer's action in the call.

<guidance>
Output your response in the following
way.

<theme_label_explanation>
Your short step-by-step explanation
behind the theme
</theme_label_explanation>

<theme_label>
your theme label
</theme_label>

</guidance>
</task>

H:
<utterances>
{utterances}
```

```
</utterances>
```

B Theme Label Writing Guideline

An acceptable theme label is structurally and semantically well-formed according to the rules outlined in this appendix. *Structurally well-formed* means that the words and their arrangement in the theme label are acceptable. *Semantically well-formed* means that the meaning and usability of the theme label are acceptable.

B.1 Theme labels exclude unneeded and undesirable words.

Theme labels should be concise (2–5 words long). They should only include essential words (see B.2 and B.2.1 below). Essential words will primarily include content (open-class) words. Function (closed-class) words should be excluded. Prepositions may be included as needed but should be avoided when there is a synonymous alternative label without a preposition.

Theme labels should also exclude context-sensitive words like pronouns (*him*, *her*, *them*, *it*, *us*, etc.) and demonstratives (*this*, *that*, *those*, etc.).

B.2 Word types

- Content/open-class words:

    - nouns (*items*, *insurance*, *information*, *order*, etc.)
    - main verbs (*check*, *inquire*, *add*, *explore*, etc.)
    - adjectives (*new* patient, *missing* item, etc.)
    - other modifying words (*shipping* information, *product* options, etc.)

- Function/closed-class words:

    - articles/determiners (*the*, *a*, etc.)
    - auxiliary verbs (*have* or *be*, as in *I have eaten* or *I am eating*)
    - copulas
    - negation (*not* or *-n't*, as in *not on time* or *didn't arrive*)
    - conjunctions (*and*, *or*, *but*, etc.)
    - complementizers (clause-embedding uses of *that*, *for*, *if*, *whether*, *because*, etc.)
    - modals (*can*, *could*, *will*, *would*, *may*, *might*, *must*, *shall*)

- question words (*who*, *what*, *where*, *when*, *how*, *why*)

- Context-sensitive words:

  - pronouns (*she*, *he*, *they*, *it*, *her*, *his*, etc.)
  - demonstratives (*this*, *these*, *that*, *those*, etc.)
  - temporal adverbs (*yesterday*, *tomorrow*, *next week*, etc.)
  - other context-sensitive language
    * *one*, as in *I'm looking for a nearby branch. Can you find* one*?*
    * deleted nouns (noun ellipsis), as in *I found his order, but not yours __.*

B.2.1 Examples

For a theme covering order tracking:

- **Good**: track order

- **Good**: track shipment

- **Bad**: track an order (includes an article)

- **Bad**: track their order (includes a pronoun)

For a theme covering finding the nearest branch of a chain:

- **Good**: find nearest branch

- **Good**: find closest branch

- **Bad**: find nearest one (includes context-sensitive *one*)

- **Bad**: check if there's a nearby branch (includes a complementizer *if*; includes a form of *be*)

B.3 Theme labels are verb phrases that classify events.

A verb phrase begins with a verb and may include arguments or modifiers of the verb (such as a direct object). The verb should be in its citation form, lacking any complex morphology such as tense or agreement suffixes. The citation form of a verb is what would normally follow the infinitive *to*, such as *sign up* in *I'd like to* sign up. Theme labels should not be other phrase types, such as noun phrases.

The verb phrase should describe a class of events. Events are things that can be said to **happen**, unlike states (e.g. *learn* [event] vs. *know* [state]), entities

(e.g. *redeem* [event] vs. *redemption* [entity]), properties (e.g. *complain* [event] vs. *angry* [property]), and claims (*report defect* [event] vs. *product is defective* [claim]).

B.3.1 Examples

For a theme covering membership sign-ups:

- **Good**: sign up for membership (verb phrase; describes a kind of *signing up* event)

- **Bad**: signing up for membership (verb phrase, but verb is not in citation form)

- **Bad**: membership sign-up (noun phrase; describes a kind of entity)

- **Bad**: memberships (noun phrase; describes a kind of entity)

For a theme covering requests to check in early at a hotel:

- **Good**: request early check-in (verb phrase; describes a kind of requesting event)

- **Bad**: requested early check-in (verb phrase, but verb is not in citation form)

- **Bad**: request for early check-in (noun phrase; describes a kind of entity)

- **Bad**: customer wants early check-in (this is a claim)

For a theme covering reporting a defective product:

- **Good**: report defective product (verb phrase; describes events)

- **Bad**: reporting defective product (verb phrase, but verb is not in citation form)

- **Bad**: believe product is defective (verb phrase, but describes a state rather than an event)

- **Bad**: defective product (noun phrase; describes a kind of entity)

B.4 Theme labels are informative and actionable yet sufficiently general.

Theme labels should be informative enough to substantially narrow down the set of possible customer issue resolution steps (the steps to resolve the problem/need that drove the customer to make contact). For example, *check balance* is probably associated with a standard procedure for checking the balance

of a range of customer account types, but *perform check* is so broad that it could be associated with an extremely diverse group of issue resolutions. Non-actionable theme labels may be excessively vague or uninformative, and hence not very useful.

B.4.1 Examples

For a theme covering appointment-scheduling themes:

- **Good**: schedule appointments

- **Bad**: ask about appointments (probably too general)

- **Bad**: schedule appointment for next week (too specific)

- **Bad**: schedule appointment for elderly parent (too specific)

For a theme covering adding a recognized user to an existing account or policy:

- **Good**: add user

- **Bad**: add one (too general)

- **Bad**: add oldest child (too specific)

For a theme covering user password issues:

- **Good**: reset password

- **Good**: troubleshoot password

- **Bad**: secure account (too general)

- **Bad**: reset password again (too specific)

For a theme covering credit or debit card charge disputes:

- **Good**: dispute charge

- **Bad**: complain about charge (too general)

- **Bad**: file card complaint (too general)

- **Bad**: dispute charge for defective blender (too specific)

C Human Evaluation Guidelines

C.1 Structural Dimensions

C.1.1 Conciseness & Word Choice

Options: Pass (1) / Fail (0)
Definition: The following criteria are consolidated by the evaluator into one Pass/Fail rating for Conciseness & Word Choice:

1. **Label length**: Is the label concise, containing only 2–5 words?

   - **Pass**: update billing address
     Fail: update customer's residential billing address for future statements
     Rationale: The good example uses 3 words, within the required 2-5 word range. The bad example uses 8, making it unnecessarily verbose when the core intent can be expressed more concisely.
   - **Pass**: access account statement
     Fail: statement
     Rationale: The good example uses 3 words, adhering to the 2-5 word guideline. The bad example uses only one word, which lacks sufficient specificity to be useful as a theme label.

2. **Function word exclusion**: Does the label exclude unnecessary function words (articles, auxiliary verbs, etc.)?

   - **Pass**: add dependent coverage
     Fail: add the dependent to coverage
     Rationale: The good example correctly excludes function words like articles ("the"), focusing only on essential content words. The bad example unnecessarily includes "the", which should be excluded according to guidelines.
   - **Pass**: troubleshoot internet connection
     Fail: troubleshoot why internet is not working
     Rationale: The good example properly excludes function words, while the bad example improperly includes function words "why," "is," and "not" which should be excluded for conciseness.

3. **Avoidance of context sensitivity**: Does the label exclude context-dependent words (pronouns, demonstratives, temporal adverbs, etc.)?

- **Pass**: return defective product
  Fail: return this item
  Rationale: The good example avoids context-sensitive words like "this" and uses the general term "product" that can apply across contexts. The bad example includes the context-sensitive demonstrative "this," which requires a specific context to understand its meaning.
- **Pass**: reschedule appointment
  Fail: reschedule it for tomorrow
  Rationale: The good example uses general terminology applicable to any appointment, while the bad example includes both the pronoun "it" and the temporal adverb "tomorrow," both of which are dependent on conversation context for their meaning.

4. **Preposition usage**: Are prepositions included only when necessary?

- **Pass**: transfer funds
  Fail: transfer from account
  Rationale: The good example avoids unnecessary prepositions by using a concise verb-object structure. The bad example unnecessarily includes the preposition "from" when the more concise alternative without the preposition works just as well.
- **Pass**: join rewards program
  Fail: sign up for rewards program
  Rationale: The good example avoids prepositions entirely, while the bad example unnecessarily includes the preposition "for" when alternatives without prepositions are available and equally clear.

C.1.2 Grammatical Structure

Options: Pass (1) / Fail (0)
Definition: The following criteria are consolidated by the evaluator into one Pass/Fail rating for Grammatical Structure:

1. **Verb phrase structure**: Is the label a verb phrase?

- **Pass**: cancel flight
  Fail: flight cancellation
  Rationale: The good example correctly follows the verb phrase requirement by starting with a verb ("cancel") followed by a noun ("flight"). The bad example uses a noun phrase ("flight cancellation") instead.
- **Pass**: redeem rewards
  Fail: rewards redemption process
  Rationale: The good example uses a verb phrase beginning with the verb "redeem". The bad example fails by using a noun phrase with "redemption" as the head noun rather than using a verb form.

2. **Citation form**: Does the verb appear in its citation form (without tense or agreement morphology)?

- **Pass**: change delivery address
  Fail: changing delivery address
  Rationale: The good example correctly uses the citation form of the verb "change" without any tense or agreement morphology. The bad example fails by using the -ing form "changing" rather than the required base form.
- **Pass**: cancel subscription
  Fail: canceled subscription
  Rationale: The good example properly uses the citation form of the verb "cancel" without inflectional endings. The bad example incorrectly uses the past tense form "cancelled" instead of the citation form.

3. **Event classification**: Does the verb phrase describe a class of events, rather than states, entities, properties, or claims?

- **Pass**: verify warranty coverage
  Fail: warranty coverage
  Rationale: The good example describes an event (the act of verifying) rather than an entity. The bad example describes an entity (the warranty coverage itself) rather than an event, violating the requirement that theme labels classify events. Note: The bad example would also be ruled out by the verb phrase requirement.
- **Pass**: express dissatisfaction
  Fail: customer is dissatisfied
  Fail: is dissatisfied
  Rationale: The good example describes an event (the act of expressing) rather than a state. The first bad example is

structured as a claim about the customer, rather than describing en event. The second bad example is a verb phrase but describes the wrong kind of situation: a state, rather than an event.

- **Pass**: complain about faulty product (event)
  Fail: angry about faulty product (property)
  Rationale: The good example describes an event (the act of complaining) rather than a property. The bad example describes a property or attribute of the customer, rather than an event describing the customer's intent.

C.2 Functional Dimensions

C.2.1 Semantic Relevance

Options: Pass (1) / Fail (0)
Definition: Does the label accurately capture the core intent/topic of the utterance it represents? Theme labels are expected to provide a gist of the dialogue from the customer's inquiry perspective.

- **Pass**: request card security support (For customer utterance: "I received a notification that my credit card might have been compromised. I need to know what steps I should take.")
  Rationale: This theme label demonstrates good semantic relevance by accurately capturing the core intent of the customer's inquiry—addressing a potential security issue—rather than focusing on peripheral aspects like the notification itself.

- **Fail**: express frustration (For customer utterance: "I've been on hold for 45 minutes trying to get help with activating my new debit card. This is ridiculous!")
  Rationale: This theme label fails the semantic relevance test because it focuses on the customer's emotional state rather than their actual intent, which is to activate their debit card. The frustration is secondary to the core purpose of the contact.

- **Pass**: book accommodation
  Fail: inquire about Chicago
  Rationale: The good example correctly identifies the core intent (booking a hotel room), while the bad example misidentifies the intent as seeking information about Chicago when

the location is just a detail/slot related to the booking request.

C.2.2 Analytical Utility

Options: Pass (1) / Fail (0)
Definition: Does the label provide meaningful categorization that could directly support a reviewer or analyst's workflow when reviewing conversation data? Themes, which should be ready for presentation to the user/analyst, are supposed to highlight the topics discussed in the conversation that are useful for categorizing and further analyzing them according to the nature of the conversation.

- **Pass**: troubleshoot checkout error
  For customer utterance: "I'm getting error code E-503 when trying to complete my purchase on your website. I've tried three different browsers."
  Rationale: This theme label has good analytical utility because it categorizes the issue in a way that would allow analysts to, e.g., identify patterns in checkout problems, prioritize technical fixes, and track the frequency of specific error types.

- **Fail**: customer contact
  For customer utterance: "I ordered a blue shirt in size medium last week, but you sent me a red one instead. I'd like to exchange it."
  Rationale: This theme label lacks analytical utility because it's too broad to provide meaningful categorization. It fails to identify the specific issue (there's an order fulfillment error) in a way that could help improve operations or track problem patterns.

- **Pass**: downgrade service plan
  Fail: smart thermostat model TH8000 connection failure with iOS app version 3.2.1
  Rationale: The good example provides useful categorization at the right level of detail for business analysis. The bad example is too specific with technical details that would fragment similar issues into tiny categories, making pattern identification difficult.

C.2.3 Granularity

Options: Pass (1) / Fail (0)
Definition: Does the label maintain appropriate specificity, as determined by its closeness to the provided gold labels? (Submission authors are expected to infer ideal granularity from the provided user preference data.)

- **Pass**: update payment information
  Fail: manage account
  Rationale: The good example demonstrates appropriate granularity by categorizing the issue at a level that's neither too broad nor too specific. The bad example is too broad, grouping potentially diverse issues that would benefit from more specific categorization.

- **Pass**: troubleshoot device connectivity
  Fail: resolve Sony WH-1000XM4 headphones pairing failure with streaming app on Android 16 beta
  Rationale: The good example shows appropriate granularity by categorizing at a level that groups similar technical problems. The bad example has excessive granularity, including specific device models and OS versions that would create overly-fragmented categories.

C.2.4 Actionability

Options: Pass (1) / Fail (0)
Definition: Does the label provide sufficient information to categorize customer issues for resolution? Theme labels should be informative enough to substantially narrow down the set of possible customer issue resolution steps.

- **Pass**: dispute transaction
  Fail: seek assistance
  Rationale: The good example demonstrates good actionability by clearly identifying a specific process (transaction dispute) with established resolution procedures. The bad example is too vague to suggest any specific resolution path.

- **Pass**: trace missing shipment
  Fail: discuss app features
  Rationale: The good example shows good actionability by identifying a specific issue (shipment tracking problem) that points to clear resolution steps. The bad example has poor actionability because "discuss" doesn't point to a specific resolution-related action, and "app features" is too broad.

C.2.5 Domain Relevance

Options: Pass (1) / Fail (0)
Definition: Does the label reflect domain-specific terminology and concepts appropriate to the conversation context? Theme labels should reduce

manual analysis by utilizing domain-relevant and context-relevant terminology.

- **Pass**: verify coverage details
  For customer utterance: "I need to know if my insurance policy covers damage from a burst pipe in my basement."
  Rationale: This theme label demonstrates good domain relevance by using terminology ("verify coverage") that's specific to the insurance industry and reflects how claims and policy questions are typically categorized in that domain.

- **Pass**: transfer prescription
  For customer utterance: "I want to transfer my prescription from my old pharmacy to your location. Can you help with that?"
  Rationale: This theme label shows good domain relevance by using standard pharmacy industry terminology ("transfer prescription") that accurately reflects how this process is categorized and handled within the healthcare/pharmacy domain.

- **Fail**: change money amount
  For customer utterance: "I need to increase my 401(k) contribution percentage starting with my next paycheck."
  Rationale: This theme label lacks domain relevance because it uses overly-generic terminology instead of financial industry-specific language. A more domain-relevant label would be "adjust retirement contribution" or "modify investment allocation."

- **Fail**: fix travel problem
  For customer utterance: "My flight was delayed and I missed my connection. I need to be rebooked on the next available flight.")
  Rationale: This theme label has poor domain relevance because it doesn't use airline industry terminology. A more domain-relevant label would be "rebook missed connection", "accommodate disrupted itinerary", etc.

C.2.6 Thematic Distinctiveness

Options: Pass (1) / Fail (0)
Definition: Does the label create a clear boundary that differentiates one theme from the other themes in the dataset? Theme labels should exhaustively cover all the examples AND be mutually exclusive.

- **Pass**: report stolen card
  In this context: Dataset already contains theme labels "report lost card" and "report fraudulent transaction"
  For customer utterance: "Someone stole my wallet and I need to block my credit card immediately."
  Rationale: This theme label demonstrates good thematic distinctiveness by creating a clear boundary between related but distinct issues: lost cards (misplaced by owner), stolen cards (taken by someone else), and fraudulent transactions (unauthorized use).'

- **Fail**: inquire about refund
  In this context: Dataset already contains theme label "request refund"
  For customer utterance: "I returned my purchase last week but haven't seen the money back in my account yet."
  Rationale: This theme label fails the thematic distinctiveness test because it doesn't create a clear boundary between refund requests and refund status checks. The new utterance is about tracking a refund in progress, which should be a distinct category (e.g. "check refund status". Instead, this category could be compatible with utterances that are already covered by "request refund".

- **Pass**: change delivery location
  Fail: reset account
  In this context: Dataset already contains theme labels "schedule delivery", "reschedule delivery", "reset password", and "update account information"
  Rationale: The good example shows appropriate thematic distinctiveness by creating a clear boundary between different delivery modification types. The bad example blurs the boundary between password resets and other profile updates, creating confusion about categorization.

D   Input/Output Data Examples

Below is an input datapoint for a dialogue with one utterance marked as themed. For the train/dev domains, the theme labels will be available as in the example below. For the test domain, only the flag that an utterance is themed will be provided.

```
{
  "conversation_id": "Banking_123",
```

```
"turns": [
    {
        "speaker": "Agent",
        "utterance": "Thank you for
            calling Intellibank. This is
            Melanie. How can I help you
            ?"
    },
    {
        "speaker": "Customer",
        "utterance": "Yeah, hey. This is
            John Smith. I've got a
            quick question."
    },
    {
        "speaker": "Agent",
        "utterance": "OK, John. What can
            I help you with?"
    },
    {
        "speaker": "Customer",
        "utterance": "Yeah I need to
            know what your ATM
            withdrawal limits are for
            the day.",
        "theme_label": "get daily
            withdrawal limit",
    },
    {
        "speaker": "Agent",
        "utterance": "Certainly. Our ATM
            withdrawal limit is on a
            per day basis and it is up
            to two hundred dollars."
    },
    {
        "speaker": "Customer",
        "utterance": "Oh perfect,
            perfect. Yeah, I think I'll
            just see if I can head down
            to the ATM now. Thank you."
    },
    {
        "speaker": "Agent",
        "utterance": "OK, thank you. You
            have a great day."
    },
    {
        "speaker": "Customer",
        "utterance": "You too."
    }
]
}
```

Below is an input datapoint with the example user preference on clustering granularity:

```
{
  "utterance_a": {
    "utterance": "Yeah, so I need to
        change the account number thing
        that I put in whenever I go to
        the ATM."
    "conversation_id": "Banking_123",
    "turn_id": 4
  },
  "utterance_b": {
    "utterance": "OK. Excellent. Thank
        you Ms. Crystal. And while I got
        you on the phone I see it's
```

```
            been a little bit since you've
            authenticated your account here.
             Would you like to add a PIN
            number to your account for
            security reasons?"
        "conversation_id": "Banking_345",
        "turn_id": 10
    },
    "belong_to_same_theme": "yes"
}
```