



April, 2026

CSC_5AI27_TP

Large-Scale Generative Models

Project

Names: Mohamed Almahdi, Nicolas Braga, Margherita Necchi

Dialogue Intent Classification

Abstract

Using the CLINC150 benchmark, we build and compare six intent classification systems across three paradigms: a TF-IDF + Logistic Regression baseline, a fine-tuned BERT encoder, and four LLM prompting strategies (Zero-Shot, Few-Shot v1/v2, Chain-of-Thought) applied to the Mistral-small API with no training. A unified evaluation pipeline enables fair cross-system comparison despite heterogeneous output formats and test-set sizes. Few-Shot v2 achieves the best accuracy (76.0%) and Macro-F1 (0.839) on 500 test samples; BERT and TF-IDF reach equivalent accuracy on the full 5 500-sample test set at orders-of-magnitude lower inference cost.

Problem Statement

This project aims to build a system that identifies the intent of a user utterance in a dialogue system. Intent classification is a core component of conversational AI, where the system must determine the user's goal in order to generate an appropriate response or trigger an action.

Given a user input such as *"Book a flight to Paris tomorrow"*, the system should classify the intent as `flight_booking`. The task is formulated as a text multi-class classification problem: each input utterance is assigned exactly one label corresponding to the user's intention, chosen from a fixed set of 151 classes. Three challenges motivate our comparative study:

- **Fine-grained label space.** 150 in-domain intents with heavily overlapping surface forms (e.g. `cancel` vs. `cancel_reservation`).
- **Out-of-scope (OOS) detection.** Real dialogue systems must also recognise utterances that match no known intent.
- **Resource trade-offs.** Training-based approaches require labelled data and compute; prompting-based approaches trade these for per-query API cost and latency.

Method / Overall Architecture

We evaluate three model families; full descriptions and prompt templates are provided in Appendix A.

Baseline: TF-IDF + Logistic Regression

As an initial benchmark for the intent classification task on CLINC150, we implement a pipeline based on Term Frequency–Inverse Document Frequency (TF-IDF) features combined with a logistic regression classifier. The objective is to establish a simple yet interpretable reference point before experimenting with more complex neural architectures, and to quantify the predictive power of traditional bag-of-words representations on this benchmark.

Each utterance is encoded as a sparse high-dimensional vector using TF-IDF vectorisation. Unigram and bigram features are included in order to capture not only individual word relevance but also short phrasal patterns that are often informative for intent recognition, such as “*transfer money*” or “*book flight*”. TF-IDF was selected because it provides a computationally efficient method for emphasising discriminative words while reducing the influence of highly frequent but less informative terms.

Once vectorised, a multinomial logistic regression model is trained to map the TF-IDF features to the 151 intent categories. Logistic regression was chosen as the baseline classifier due to its strong performance on sparse text representations, its simplicity of implementation, and its ability to provide a linear decision boundary that is easy to analyse. The optimisation objective is to minimise the cross-entropy loss over the intent labels, allowing the model to learn the association between lexical patterns and intent classes. After training, the model is evaluated on the validation set using classification accuracy as the primary metric.

Although this baseline approach does not capture contextual or semantic relationships beyond surface-level lexical features, it serves as a meaningful starting point that quantifies the predictive power of bag-of-words representations on CLINC150, and establishes a reference against which the performance gains of more advanced architectures can be measured.

BERT

Following the TF-IDF baseline, we implement a transformer-based model using BERT [?]. Specifically, we use the **bert-base-uncased** variant. The motivation for moving to BERT is that, unlike the sparse TF-IDF representation, BERT generates dense contextual embeddings in which the meaning of each word is informed by the surrounding words in the utterance, making it significantly better suited for distinguishing intents that may share overlapping vocabulary but differ in semantic context.

The raw utterances from CLINC150 are first tokenised using the pre-trained BERT tokeniser, which converts text into subword tokens based on the WordPiece algorithm and appends the special classification token [CLS] at the beginning of each sequence and the separator token [SEP] at the end. Since utterances vary in length, each tokenised sequence is padded or truncated to a fixed maximum sequence length to ensure uniform batch processing. Corresponding attention masks are generated to distinguish actual tokens from padding tokens so that the model only attends to meaningful input during training.

These processed inputs are fed into the pre-trained **bert-base-uncased** encoder, which consists of multiple bidirectional transformer layers that produce contextual embeddings for every token in the sequence. For the classification task, the final hidden representation of the [CLS]

token is used as a fixed-length embedding summarising the semantic content of the entire utterance. This representation is then passed through a fully connected linear classification layer whose output dimension matches the 151 intent classes.

The model is fine-tuned end-to-end on the CLINC150 training split, meaning that both the pre-trained BERT parameters and the classification head are updated during supervised training. The cross-entropy loss function compares predicted logits against true intent labels; the AdamW optimiser updates the parameters with a deliberately small learning rate to avoid destabilising the pre-trained representations. Training is conducted over several epochs with periodic evaluation on the validation set to monitor performance. Compared to the TF-IDF baseline, this architecture introduces a much richer representation of language by capturing syntactic dependencies and semantic nuances within each utterance, allowing the model to better differentiate between intents that contain similar words but express different user goals.

LLM Prompting (Mistral-small, zero training)

All experiments use the Mistral-small API on 500 randomly sampled test utterances; no labelled training data is required. Three approaches were developed, each building on the previous one.

Approach 1: Zero-Shot. The model receives the full list of 151 intent names and must predict the label in a single API call. A key finding is that *prompt wording alone* has a large impact: three formulations were tested on the same model and task (Table 1), yielding up to +6 pp accuracy for free.

Table 1: Zero-shot prompt variants: same model, same task, different wording.

Variant	Prompt style	Accuracy
V1 — Direct	“Classify this message ... Reply with only the intent name.”	65%
V2 — Explicit Role	“You MUST choose exactly one intent. Do NOT add explanations.”	71%
V3 — Natural Question	“What does the user want to do? Pick from these intents.”	69%

V2 (explicit-role framing) was retained as the base prompt for all subsequent approaches.

Approach 2: Targeted Few-Shot. Classic full few-shot would require $151 \times 3 = 453$ labelled examples per prompt, exceeding the free API tier’s context limit. The targeted pipeline solves this with two API calls per sample: (i) a zero-shot call narrows the field to the 5 most probable candidate intents; (ii) a second call provides 3 training examples per candidate (15 total) and asks the model to select among them. This preserves in-context learning while reducing example count from 453 to 15. A second prompt-wording iteration (v2) further improved results.

Approach 3: Chain-of-Thought (CoT). The model is instructed to reason step by step (*topic* $\rightarrow$ *user goal* $\rightarrow$ *best matching intent*) before committing to a label. Explicit intermediate reasoning reduces ambiguity on semantically adjacent intents without requiring any labelled examples.

Experimentation

Dataset & Dataset Statistics

We use **CLINC150** [?], a standard benchmark for intent classification in task-oriented dialogue. The dataset contains 22 500 in-domain utterances spanning 150 intent classes across 10 domains

(banking, travel, food, smart home, utilities, entertainment, and more), plus 1 200 out-of-scope (OOS) utterances (151 classes in total). The corpus is perfectly balanced: 100 training examples per in-domain intent. An 80/20 train-validation split is used for supervised learning. Classical models are evaluated on the full 5 500-sample test set; LLM experiments use 500 randomly sampled utterances.

Table 2: CLINC150 dataset statistics.

Split	In-domain samples	OOS samples
Train	15 000	100
Validation	3 000	100
Test (full)	4 500	1 000
<i>LLM subset</i>	<i>500</i>	—

Experimental Results based on Evaluation Metrics

A shared evaluation pipeline normalises every model’s output to a common schema [text, true_label, pred_label, correct, is_oos, model], decoding numeric intent IDs (classical models) and string labels (LLMs) into a comparable form. Reported metrics: **Accuracy**, **Macro-F1** (treats all 151 classes equally), **Weighted-F1** (weighted by class support), and **OOS Accuracy**.

Table 3: Global evaluation metrics. LLM models evaluated on 500 samples; Regex and BERT on 5 500 samples. Best result per column in **bold**.

Model	Accuracy	Macro F1	Weighted F1	Precision (W)
Zero-Shot	0.650	0.695	0.618	0.745
Few-Shot v1	0.742	0.823	0.699	0.861
Chain-of-Thought	0.758	0.813	0.740	0.844
Few-Shot v2	0.760	0.839	0.722	0.870
Regex (TF-IDF + LR)	0.750	0.815	0.711	0.800
BERT	0.750	0.815	0.741	0.800

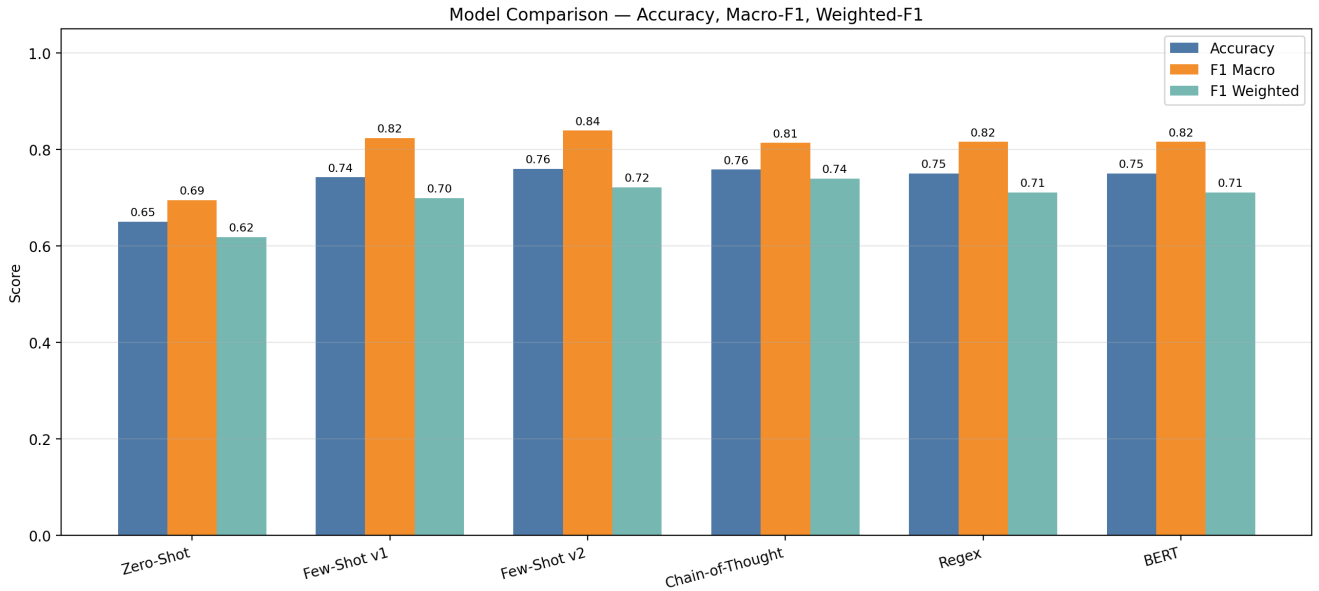


Figure 1: Accuracy, Macro-F1, and Weighted-F1 across all six systems.

Key observations: (i) **Few-Shot v2** is the top performer (76.0%, Macro-F1 = 0.839); (ii) **classical models match LLMs** on the full test set (both at 75.0%) at a fraction of the cost; (iii) **prompt wording alone adds +6 pp** (Zero-Shot V1 → V2); (iv) **Zero-Shot is the weakest** approach (65.0%), confirming CLINC150 is too fine-grained for pure instruction-following.

LLM Approaches: Step-by-Step Progression

Table 4 shows how each successive LLM approach builds on the previous one. The largest single gain (+9.2 pp) comes from introducing targeted few-shot examples; prompt reformulation alone adds +6 pp at zero extra cost.

Table 4: LLM approach progression on 500 test samples (Mistral-small).

Approach	Accuracy	vs Zero-Shot V1
Zero-Shot V1 (baseline prompt)	65.0%	—
Zero-Shot V2 (explicit-role prompt)	71.0%	+6.0 pp
Targeted Few-Shot v1	74.2%	+9.2 pp
Targeted Few-Shot v2 (best prompt)	76.0%	+11.0 pp
Chain-of-Thought	75.8%	+10.8 pp

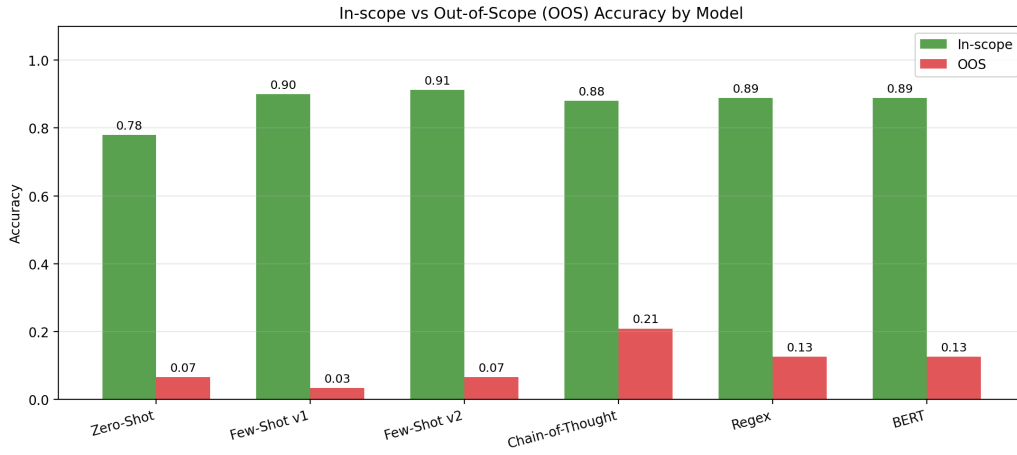


Figure 2: In-scope vs. OOS accuracy. LLM approaches tend to force-assign an in-scope intent rather than abstaining.

Error Analysis

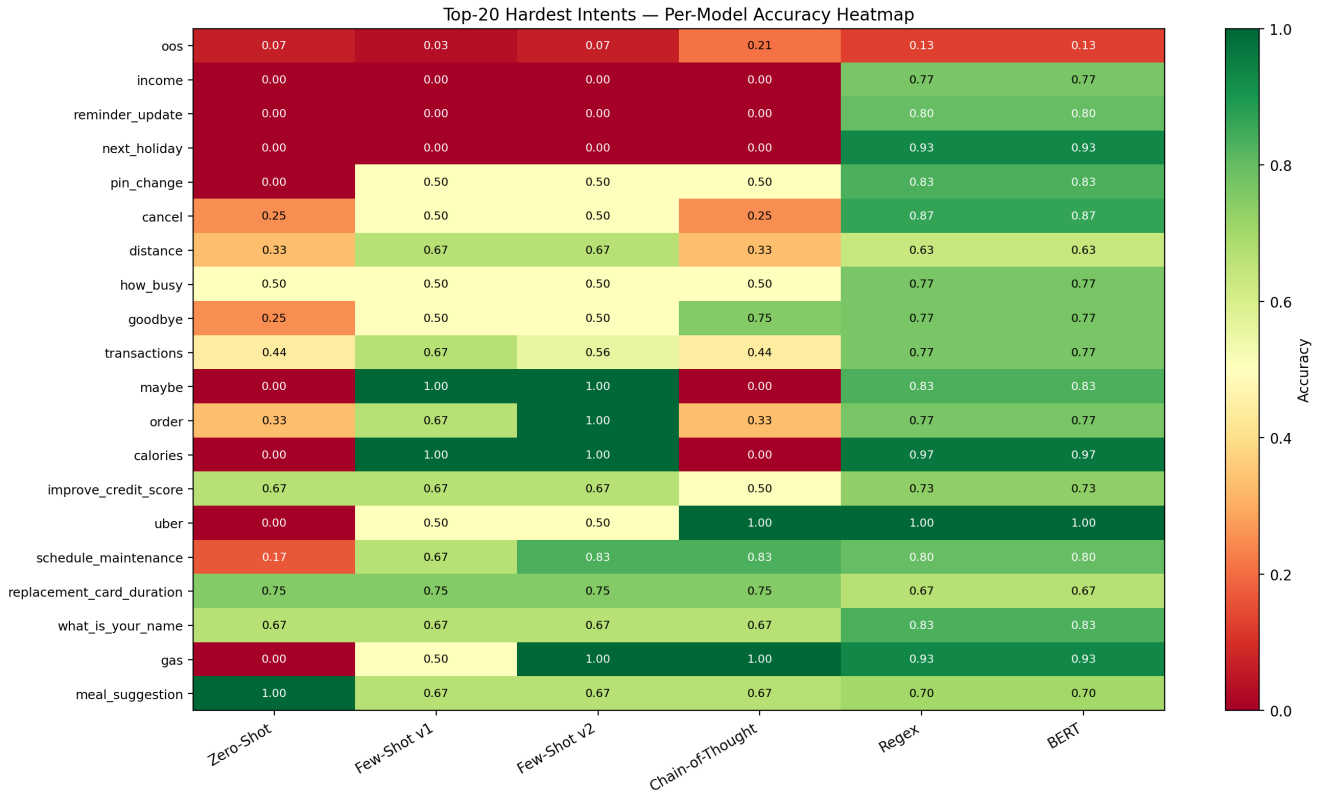


Figure 3: Heatmap of the 20 hardest intents (lowest mean accuracy across all models). Red = 0.0, green = 1.0.

A persistent set of intents is universally hard across all six systems. Failures share three root causes: (i) **semantic proximity** to a sibling intent (e.g. `change_language` vs. `change_accent`); (ii) **OOS utterances** consistently misclassified as in-scope: no model abstains reliably without an explicit mechanism; (iii) **abstract or meta intents** (e.g. `meaning_of_life`). Per-intent accuracy

does not correlate with test-set support, confirming that difficulty is driven by semantic ambiguity rather than data sparsity (see Fig. 7 in Appendix B).

Discussion

Table 5: Qualitative comparison along practical deployment dimensions.

Criterion	BERT / TF-IDF	Zero-Shot	Few-Shot / CoT
Accuracy (test)	75%	65–71%	74–76%
Training required	Yes (BERT)	No	No
Inference cost	Very low	Medium	Medium–High
Latency / query	<1 ms	~1–2 s	~2–5 s
New intent	Retrain	Prompt edit	Prompt + examples
OOS detection	Threshold	Poor	Poor

Strengths of the LLM approach.

- 75.8% accuracy *with no fine-tuning*: the model is used directly off-the-shelf via the API.
- Prompt engineering is fast and flexible: adding or renaming an intent requires only a prompt edit, no retraining.
- The targeted few-shot pipeline is an original, efficient solution to the context-length bottleneck of standard few-shot (453 $\rightarrow$ 15 examples per prompt).

Comparison with expectations. Prompting-based methods reach competitive accuracy without any training data, validating that LLMs generalise well to fine-grained classification with careful prompt engineering. The +6 pp gain from reformulating the zero-shot prompt is striking: wording is as important as model architecture. Classical models remain fully competitive at a fraction of the inference cost and with native OOS support via confidence thresholding.

Limitations.

- API rate limits on the free tier constrained LLM experiments to 500 samples, making a strictly fair comparison with the 5 500-sample classical evaluation inexact.
- Ambiguous and out-of-scope intents remain hard: no prompting strategy provides reliable OOS abstention.
- The targeted few-shot pipeline requires 2 API calls per sample, doubling latency and cost relative to zero-shot.

Perspectives. Promising directions include: using a larger model (Mistral-large or GPT-4) to raise the accuracy ceiling, embedding-based semantic example selection to improve few-shot quality dynamically, combining prompting with fine-tuning (e.g. instruction-tuning a small open model on CLINC150), and evaluating LLM approaches on the full 5 500-sample test set for a fairer head-to-head comparison.

Appendix A: Prompt Templates

A.1 Zero-Shot V2 (base prompt for all LLM approaches)

You are an intent classification system.

Classify this message into one of these intents: restaurant_reviews, weather, translate, ... [full list of 151 intent names]

User: "how do I say hello in Spanish"

You MUST choose exactly one intent. Do NOT add explanations.

Reply with only the intent name.

A.2 Targeted Few-Shot: Stage 2 prompt structure

You are an intent classification system.

Here are some examples for the most likely intents:

Intent: translate

— "how do I say goodbye in French"

— "what is the Spanish word for hello"

— "translate this sentence to German"

Intent: change_language

— ... [3 examples per candidate, 5 candidates total = 15 examples]

Now classify this message: "how do I say hello in Spanish"

You MUST choose one of the 5 candidate intents above.

A.3 Chain-of-Thought prompt

Analyze this user message step by step:

- 1. What topic is the user talking about?*
- 2. What does the user want to do?*
- 3. Which intent from the list matches best?*

User: "how do I say hello in Spanish"

After your reasoning, write:

Intent: <intent_name>

Appendix B: Additional Evaluation Figures

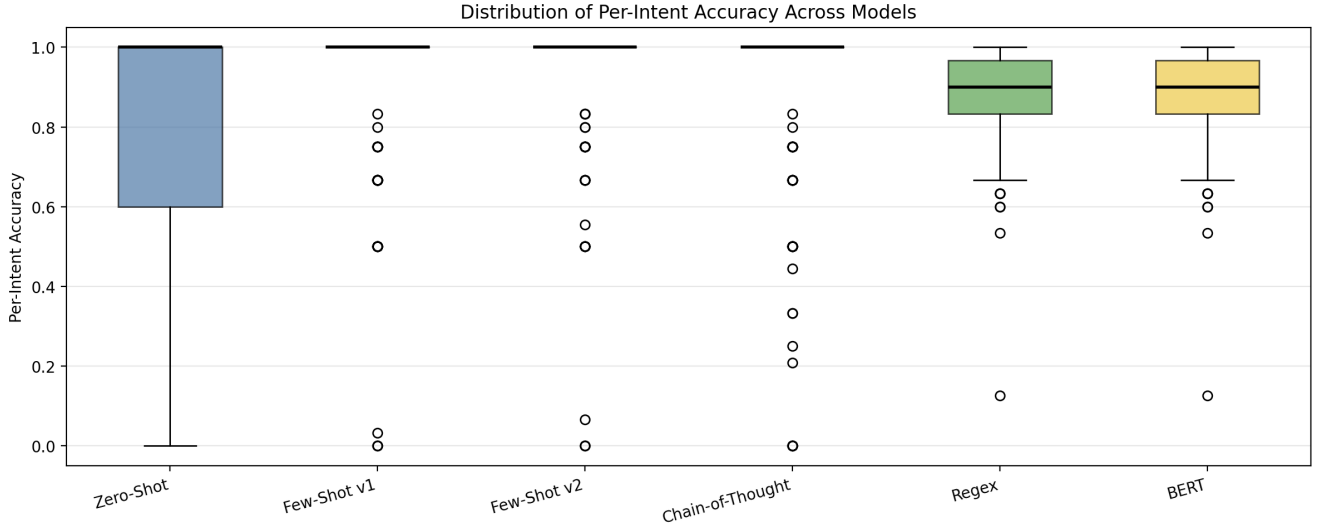
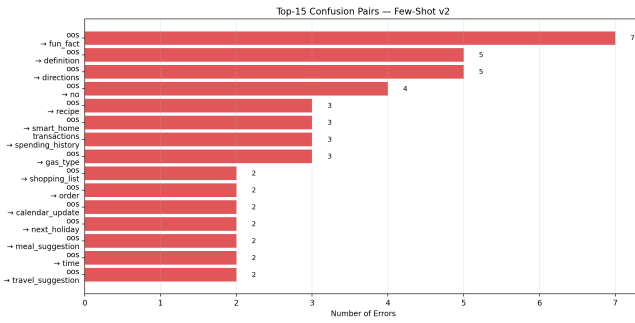
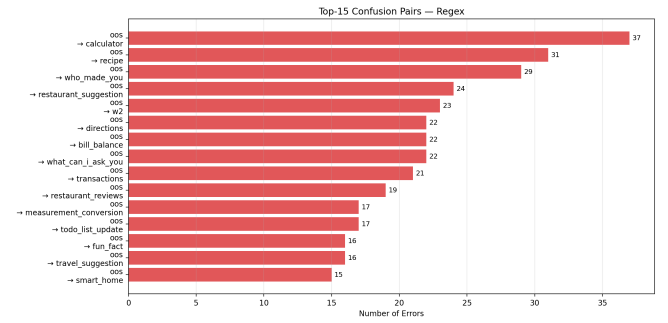


Figure 4: Box plot of per-intent accuracy across all models. Few-Shot v2 and CoT display the highest median and smallest spread; Zero-Shot has the widest distribution, reflecting high variance across the 151 classes.



(a) Best LLM: Few-Shot v2



(b) Best classical: BERT / TF-IDF

Figure 5: Top-15 most frequent confusion pairs. LLM errors cluster on OOS utterances mapped to in-scope intents; classical model errors concentrate on fine-grained intra-domain distinctions.

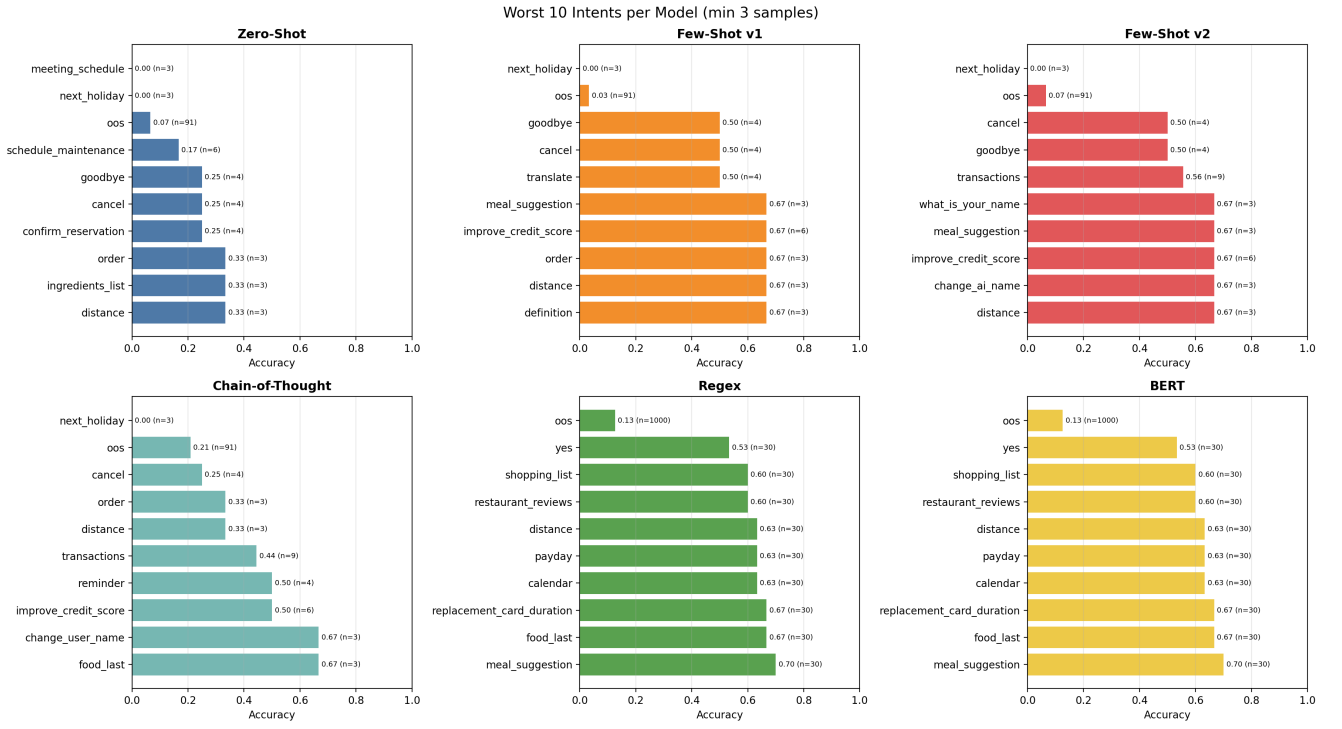


Figure 6: Top-10 worst intents per model. LLM-based approaches fail more on abstract or meta intents; classical models fail on fine-grained intra-domain distinctions, suggesting complementary blind spots that an ensemble could exploit.

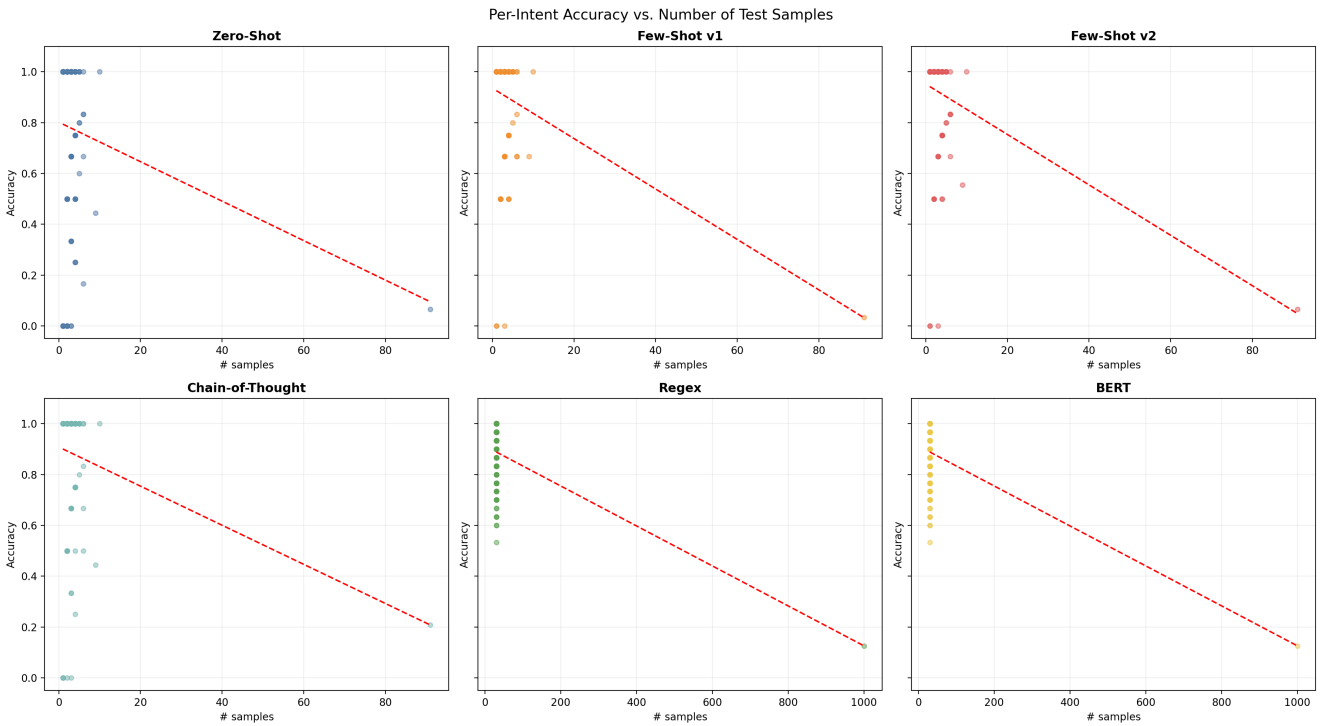
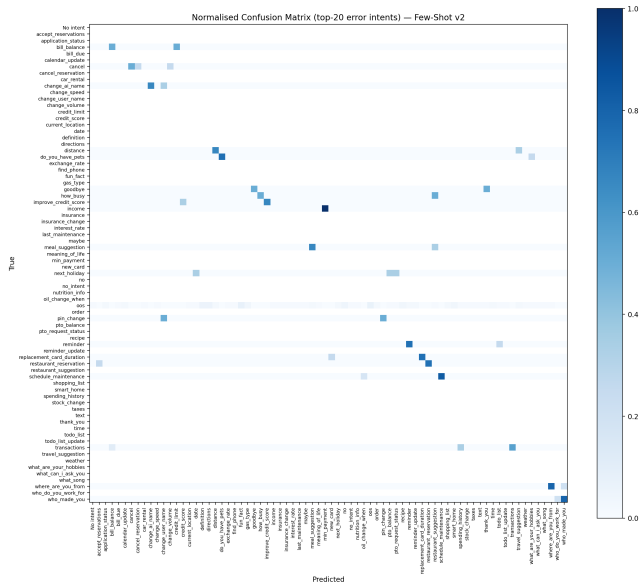
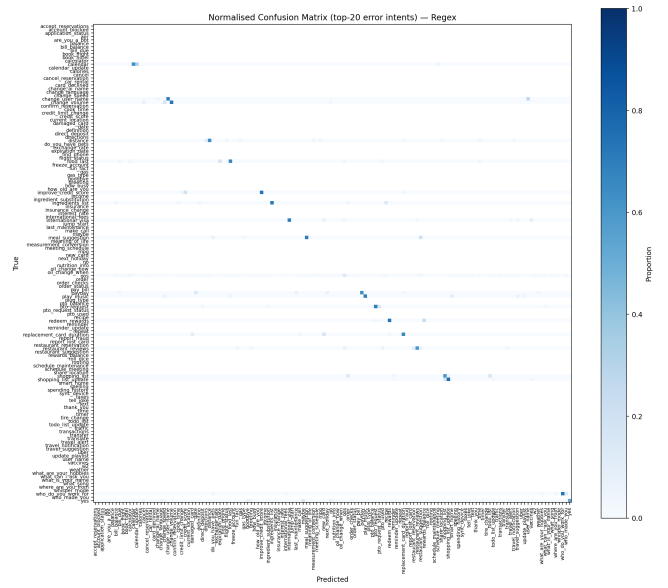


Figure 7: Per-intent accuracy vs. number of test samples per intent with a linear trend line per model. The absence of a strong positive correlation confirms that difficulty is driven by semantic ambiguity, not data sparsity.



(a) Few-Shot v2



(b) BERT / TF-IDF

Figure 8: Full confusion matrices for the best LLM and best classical model.