informatics

MDPI

Article

Global Embeddings, Local Signals: Zero-Shot Sentiment Analysis of Transport Complaints

Aliya Nugumanova, Daniyar Rakhimzhanov * and Aiganym Mansurova *

Big Data and Blockchain Technologies Research Innovation Center, Astana IT University, Astana 010000, Kazakhstan; a.nugumanova@astanait.edu.kz

* Correspondence: d.rahimzhanov@astanait.edu.kz (D.R.); 231945@astanait.edu.kz (A.M.)

Abstract

Public transport agencies must triage thousands of multilingual complaints every day, yet the cost of training and serving fine-grained sentiment analysis models limits real-time deployment. The proposed "one encoder, any facet" framework therefore offers a reproducible, resource-efficient alternative to heavy fine-tuning for domain-specific sentiment analysis or opinion mining tasks on digital service data. To the best of our knowledge, we are the first to test this paradigm on operational multilingual complaints, where public transport agencies must prioritize thousands of Russian- and Kazakh-language messages each day. A human-labelled corpus of 2400 complaints is embedded with five open-source universal models. Obtained embeddings are matched to semantic "anchor" queries that describe three distinct facets: service aspect (eight classes), implicit frustration, and explicit customer request. In the strict zero-shot setting, the best encoder reaches 77% accuracy for aspect detection, 74% for frustration, and 80% for request; taken together, these signals reproduce human four-level priority in 60% of cases. Attaching a single-layer logistic probe on top of the frozen embeddings boosts performance to 89% for aspect, 83–87% for the binary facets, and 72% for end-to-end triage. Compared with recent fine-tuned sentiment analysis systems, our pipeline cuts memory demands by two orders of magnitude and eliminates task-specific training yet narrows the accuracy gap to under five percentage points. These findings indicate that a single frozen encoder, guided by handcrafted anchors and an ultra-light head, can deliver near-human triage quality across multiple pragmatic dimensions, opening the door to low-cost, language-agnostic monitoring of digital-service feedback.

Keywords: sentiment analysis; frustration detection; complaint classification; instruction-tuned embeddings; fine-grained sentiment analysis

Academic Editors: Patricia Anthony and Jing Zhou

Received: 3 July 2025
Revised: 31 July 2025
Accepted: 12 August 2025
Published: 14 August 2025

Citation: Nugumanova, A.; Rakhimzhanov, D.; Mansurova, A. Global Embeddings, Local Signals: Zero-Shot Sentiment Analysis of Transport Complaints. *Informatics* **2025**, *12*, 82. https://doi.org/10.3390/informatics12030082

1. Introduction

Public transport providers often receive customer complaints via online platforms and social media [1]. Mining these complaints can provide actionable insights into service issues and passenger experience. Previous work in text analytics for transportation has primarily focused on sentiment analysis as a polarity classification task, typically identifying positive, neutral or negative sentiments in customer feedback or extracting a limited basic range of emotions, such as joy, sadness, and anger [2–5]. Most of these studies rely on conventional machine learning pipelines that combine lexicon-based features with linear or shallow neural classifiers.

However, complaints are not merely texts expressing dissatisfaction; rather, they are socially situated acts that position the speaker in relation to institutional structures and expectations. Traditional sentiment analysis, with its reliance on fixed polarity categories or constrained emotional lexicons, fails to capture this complexity [6]. It typically reduces multi-faceted communicative acts to simplistic labels, often strips away the social dynamics, and interprets underlying emotions as mutually exclusive, contrary to psychological frameworks [6].

Recent advances in large language models (LLMs) offer new opportunities for fine-grained sentiment analysis [7–10]. Extended context windows, rich world knowledge, and prompt-following capabilities of LLMs make them uniquely suited to extract pragmatic cues such as complaint aspects, emotion types and intensities, or latent intentions. Nowadays, generative LLMs like GPT or Claude Sonnet can directly execute complex prompts, performing fine-grained sentiment analysis without fine-tuning or supervision. However, the practical deployment of these models faces two major constraints: latency and cost. Generative LLMs typically operate via API-based inference and consume high memory and compute per query. This makes them prohibitively expensive and impractical for large-scale, near-real-time complaint triage in operational settings.

Universal text embeddings such as E5 [11], BGE-M3 [12], GTE [13], or GIST [14] alleviate these constraints while retaining key benefits: instruction-tuned or self-knowledge-distilled training, zero-shot generalization, and seamless domain transfer. Cao [15] defines universal text embeddings as unified and comprehensive models that can address a multitude of input text lengths, downstream tasks, domains, and languages. These models generate compact embeddings that run efficiently on local GPUs and even consumer-grade hardware. Their efficiency makes them suitable for lightweight classification pipelines in which each complaint is encoded together with a task-specific signal prompt or query and then assessed through nearest prototype matching or similarity thresholding. For instance, ref. [16] shows that E5-large embedding model combined with a logistic regression head achieves an F1 score of 90.4% on the Stanford Sentiment Treebank, only 0.8 percentage points (pp) below GPT-3.5-turbo. Moreover, embeddings are generated in just a few milliseconds per sentence on a commodity GPU, whereas inference with LLMs typically incurs latencies that are two orders of magnitude higher and entails per-token API charges.

The most notable aspect of using lightweight classification via embeddings is that the embedding model remains frozen, i.e., its weights are not updated, while only a single linear layer (a classification head) is trained, which significantly reduces the computational cost. The frozen strategy keeps the embedding model fixed and uses it solely to map input texts into dense vector representations that are then fed to lightweight classifiers such as logistic regression, random forests, or shallow neural networks. Therefore, freezing the encoder removes the need to back-propagate through hundreds of millions of parameters; only a classification head is trained, making learning faster, more memory-efficient, and less prone to overfitting. This strategy mirrors the protocol of the MTEB benchmark [17], where models are purposely not fine-tuned so that performance reflects the intrinsic semantic quality of the embeddings rather than the capacity of a task-specific head.

By contrast, the unfrozen strategy fine-tunes the embedding model, allowing for the internal representations to shift and align more closely with the specific requirements of the target task. This strategy typically yields higher ceilings in accuracy and F-score across benchmarks [18,19]. The trade-off is a much larger computational footprint and the risk of catastrophic forgetting where the model loses previously acquired general linguistic knowledge due to over-specialization.

This study investigates the capacity of five state-of-the-art universal embeddings (multilingual-E5-large-instruct, E5-mistral-7b-instruct, GTE-Qwen2-1.5B-instruct, BGE-M3,

and LaBSE) to perform multi-faceted classification of public transport complaints under a frozen encoder setting. Two analytical regimes are examined, both sharing the frozen encoder but differing in the presence of trainable parameters:

- Zero-shot learning. The model is used exactly as released; complaint embeddings are compared to handcrafted semantic anchors and class decisions are made through nearest prototype matching. No task-specific weights are introduced or updated.
- Lightweight supervised learning (frozen encoder + linear probe). The model remains frozen, yet a single-layer logistic classifier is trained on top of the complaint embeddings to learn optimal decision boundaries from labelled data. Only this shallow head ($\approx$0.1% of the backbone parameters) is updated.

In both regimes, facet-level signals are extracted by feeding the frozen encoder with concise, task-specific prompts. This motivates a unified pipeline that leverages state-of-the-art universal embeddings to classify multiple facets of public transport complaints, without training a separate model for each facet.

Building on this motivation, we formulate two research questions:

- RQ 1. How accurately do universal embedding models classify passenger complaints across the three facets (service aspect, latent frustration, explicit request) in a strictly zero-shot setting, with no fine-tuning and no additional classification head?
- RQ 2. How much additional accuracy can be gained by training a lightweight supervised multi-faceted classifier (e.g., linear probe) on top of the same frozen embeddings?

The contribution of this study lies in a comparative evaluation of universal embedding models under minimal supervision for a realistic, multi-faceted complaint classification task. Our approach highlights the untapped potential of modern instruction-tuned embeddings for scalable, low-cost public service analytics. The operational scheme of our multi-faceted classification is grounded in sociolinguistic theory, which views complaints not simply as reports of service failure but as socially situated speech acts. Drawing on Goffman's dramaturgical perspective [19], we treat each complaint as a staged interaction in which the passenger presents the complaint aspect (the first facet), negotiates face by expressing or masking frustration (the second facet), and, when appropriate, moves the interaction forward by formulating an explicit request (the third facet). Inspired by the "one encoder, any task" principle introduced by Su et al. [20], we explore whether a single embedding model can serve multiple analytical perspectives simultaneously: in short, "one encoder, any facet".

The remainder of this article is structured around the two research questions outlined above. The Related Work Section reviews existing research on fine-grained sentiment analysis to position and differentiate the contribution of our study. The Materials and Methods Section begins with a description of our complaint dataset, followed by the rationale for selecting the three analytical facets: service aspect, latent frustration, and explicit request. We then present the five universal embedding models evaluated in this study, along with the zero-shot classification protocol and the supervised baselines used for comparison. The Results and Analysis Section reports both zero-shot and supervised classification performance across all models and facets. The Discussion Section interprets these findings considering model architecture, instruction tuning, and embedding behavior. Finally, the Conclusion Section summarizes key takeaways and outlines directions for future research.

2. Related Work

2.1. The Concept of Fine-Grained Sentiment Analysis

While traditional sentiment analysis typically produces binary or ternary polarity classifications (positive/negative/neutral), fine-grained sentiment analysis aims to capture more specific emotions and their intensity levels. For instance, ref. [21] emphasizes that detecting sentiment intensity is a core task in fine-grained sentiment analysis. IBM [22] and Amazon [23] similarly define fine-grained sentiment analysis as the process of grouping text by distinct emotions and assigning each a corresponding intensity level or grade. Many business and research contexts adopt a five-class intensity scheme (from very negative to very positive), which is explicitly referred to as fine-grained sentiment analysis in the industry literature [24,25]. Dong [26] introduces multi-dimensional sentiment analysis, along with aspect-level and intensity analysis, as one of the core tasks of fine-grained sentiment analysis.

Some recent studies equate fine-grained sentiment analysis with aspect-based sentiment analysis, which focuses on detecting sentiment toward specific entities or attributes within text, rather than an overall polarity [27–30]. Other contemporary approaches extend fine-grained sentiment analysis to understand the intent or nature of the opinion expressed, especially in customer feedback [31,32]. This goes beyond what the sentiment is, to why or in what manner it is expressed. For example, identifying whether a negative sentiment is specifically a complaint, a request for help, or sarcasm can be seen as part of fine-grained opinion analysis. Ref. [31] applies fine-grained sentiment analysis to a domain-specific task by analyzing automotive complaint reports in order to detect potential product harm issues. By mining each complaint in detail and assessing its sentiment and urgency, their system provides an early warning for product crises, which is conceptually close to our work.

As evidenced above, fine-grained sentiment analysis serves as an umbrella term encompassing multiple forms of granularity. In the academic literature, it may refer to analyzing specific targets or aspects, assigning more precise sentiment scores, classifying nuanced emotions, or combining these approaches. Recent studies explicitly recognize that granularity spans several analytical dimensions rather than a single concept. The shift from coarse document-level sentiment to aspect, intensity, emotion, and intent analysis reflects an ongoing pursuit of more detailed and comprehensive sentiment understanding.

2.2. Fine-Grained Sentiment Analysis for Public Sector Applications

A systematic survey of transport sentiment research conducted this year [33] showed that machine learning baselines consistently lagged behind deep learning models by double-digit F1 scores. The field therefore pivoted to domain-adapted transformers: the first multi-label corpus of Indian transit grievances enabled RoBERTa to recognize fifteen complaint types with a macro-F1 of 0.71, ten points above classical pipelines [34]. Fine-tuning on in-domain text soon became the default. Washington's MetRoBERTa model, trained on six years of metro CRM logs, pushed topic-level accuracy toward 90% and illustrated how local data can yield transit-aware language models ready for operational dashboards [35].

Researchers next addressed the limitations of vanilla transformers by introducing structural bias. The dual-channel SSFF-GCN architecture integrated dependency syntax with contextual semantics, achieving a new state-of-the-art performance above 80% macro-F1 on benchmark aspect tasks, highlighting that graph-aware encoders can effectively capture long-range relations without compromising general-purpose embeddings [27]. In scenarios with limited annotated data, such as healthcare complaint logs, researchers turned to cross-domain transfer [36].

Large language models then introduced a frozen alternative to supervised fine-tuning [37–40]. Study [37] employs GPT-4 Turbo with a tailored prompt to identify

transport-specific complaint categories in Indian Twitter/X data and generate concise cluster summaries. The LLM-based summarizer improves ROUGE-L from 0.45 (LexRank) to 0.86, showing that a frozen model guided by domain-aware instructions can outperform extractive methods for grievance monitoring. In healthcare, Li et al. [38] prompt ChatGPT with aspect-based templates and chain-of-thought cues to analyze 504k patient reviews. Compared to direct zero-shot prompting, this raises weighted precision from 0.890 to 0.944 and macro-F1 above 0.91, confirming that prompt scaffolding enables clinic-grade facet detection without fine-tuning. A follow-up study [39] extends to multi-dimensional dissatisfaction signals, using ChatGPT-3.5 with International Classification of Diseases mapping to label aspect, sentiment, and disease code in one pass. The system achieves 0.907 weighted precision and ~0.89 accuracy across sampling schemes, supporting triage-oriented analytics with minimal retraining. Ruan et al. [40] apply a "Twitter-to-Reasoner" framework to 250k New York City transport tweets, showing that off-the-shelf LLMs can infer mode, sentiment, and root cause via a two-stage prompt. Despite some flagged edge cases, a single frozen LLM matches specialized classifiers while preserving interpretability.

2.3. Embedding-Based Approaches to Fine-Grained Sentiment Analysis

A parallel thread explores off-the-shelf sentence embeddings as lightweight alternatives to full fine-tuning [41–45]. In public sector informatics, multilingual BERT embeddings combined with K-means and a random forest head clustered Portuguese complaints [41]. A chatbot platform using frozen MiniLM with a logistic regression head routed 152 Indian grievance intents at macro-F1 0.92, showing that shallow classifiers over universal embeddings meet real-time e-government needs [42]. Hybrid models go further, e.g., a BERT–BiLSTM–CNN for Chinese public service requests that achieved weighted F1 0.95, surpassing Word2Vec baselines [43]. Similar patterns emerge elsewhere: LaBSE supports procurement anomaly detection [44], and IndoBERT powers multilingual petition routing in Indonesia [45]. These cases confirm that universal or lightly tuned embeddings, paired with simple or hybrid heads, offer strong performance with lower cost and annotation demands than full fine-tuning.

Building on this embedding-centric foundation, several studies attempt to predict multiple complaint facets with a single model by fine-tuning a shared transformer encoder in a multi-task configuration [46–50]. Singh and Saha's BERT-based framework [48] jointly learns complaint identification, Ekman-style emotion, and overall sentiment on English Twitter data, gaining 5 to 11 macro-F1 points over single-task counterparts. Bhatia et al. [49] extend the design to financial-services tweets, adding four-level severity prediction and reaching emotion F1 0.88 and severity F1 0.82 after end-to-end weight updates. Joshi et al. [50] push the idea to real-time inference, reporting complaint F1 97.9% with sub-30 ms latency using a multi-task BERT fine-tuned on streaming financial tweets. These architectures pursue a goal similar to ours, leveraging shared representations for correlated facets, yet they update the encoder weights rather than keeping them frozen, and they operate in English commercial domains rather than multilingual public service settings.

Our contribution is to assess how a single frozen encoder performs multi-faceted complaint classification in two complementary regimes: fully zero-shot and lightweight supervised probes. The prior literature documents lightweight multi-faceted fine-tuning; yet, to our knowledge, no study has explored a purely zero-shot embedding-based pipeline or directly compared zero-shot and lightweight strategies on the same public service corpus. We close this gap and offer the first systematic evaluation of both approaches.

3. Materials and Methods

3.1. Complaint Dataset

This study uses a corpus of public transportation complaints submitted by residents of Astana, Kazakhstan throughout 2023. The complaints are collected via official municipal platforms, where users report local service issues in free-text form. The full corpus includes approximately 58,000 entries. From this, a representative dataset of 2400 complaints is manually reviewed and annotated. Each record contains the complaint text and labels for four attributes: the targeted service aspect, a binary flag for implicit frustration, a binary flag for an explicit request, and the complaint's priority (see Table 1).

Table 1. Facet definitions and label descriptions for the complaint dataset.

No	Facet, Its Meaning	Label	Description	Direct or Indirect Indicators in the Text
1	Aspect, i.e., the service area addressed by the complaint	Condition	Issues related to the physical state and comfort of the bus	Indicated by phrases such as broken seats, malfunctioning doors, poor ventilation, dirt, noise, cold, etc.
		Information	Issues related to passenger information	Indicated by phrases such as missing route announcements, faulty display screens, non-working information display, etc.
		Operations	Issues related to operations	Indicated by phrases such as long waiting, late arrivals or departures, skipping stops, route deviations, overcrowding, etc.
		Payment	Issues related to fare collection and validation	Indicated by phrases such as non-working card readers, rejected QR, cash-only, double charging, difficulties topping up transport cards, etc.
		Personnel	Issues related to the behavior of drivers or conductors	Indicated by phrases such as rudeness, unsafe driving, ignoring passenger requests, improper dress, insult, boorish behavior, etc.
		Roadside	Issues related to deficiencies in stop and curb infrastructure	Indicated by phrases such as damaged shelters, poor lighting, lack of seating, slippery platforms, inaccessible ramps, etc.
		Bike	Issues related to the bike sharing system	Indicated by phrases such as bicycle, station, parked
		Lost	Lost items requests	Indicated by phrases such as lost, forgot, left
2	Frustration, i.e., an implicit emotional state caused by an obstacle to need satisfaction	0	Frustration is absent	Indicated by neutral, matter-of-fact tone and by the absence of implicit emotional tension or tangible loss or harm
		1	Frustration is present	Indicated by phrases such as loss, harm, hurt, wasted time, wasted money, disrupted plans, deterioration of health, spoiled mood, etc.
3	Request, i.e., an explicit call for action	0	Request is absent	Indicated by the absence of explicit call for action, demand, request
		1	Request is present	Indicated by phrases such as extend the route, take action, punish, fix, etc.

Table 1. *Cont.*

No	Facet, Its Meaning	Label	Description	Direct or Indirect Indicators in the Text
4	Priority level, as defined by the quadrant corresponding to the complaint's position on the Request–Frustration plane	Incident (I)	Both frustration and a request are present	Computed directly from the underlying Frustration and Request values
		Escalation (II)	Frustration is present, but a request is absent	
		Routine (IV)	Frustration is absent and a request is present	
		Record (III)	Both frustration and a request are absent	

Among the 2400 complaints, the distribution of aspect categories is highly uneven (Figure 1). More than half of the complaints concern operational issues such as delays, missed stops, and other day-to-day service failures, making this the dominant focus of passenger feedback. The second largest aspect category concerns personnel-related issues, while the third largest category pertains to payment problems. The distribution of complaint aspects follows Zipf's law, exhibiting a pronounced imbalance with a long tail.

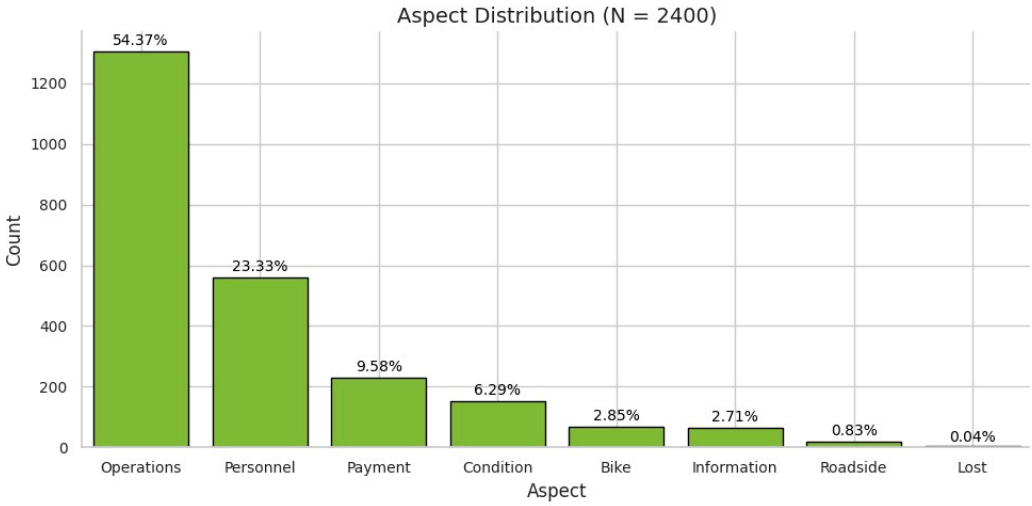


Figure 1. Distribution of Aspect categories in the dataset.

By contrast, the Frustration and Request facets are nearly balanced (see Figure 2), yet they differ in annotation difficulty. Labeling explicit frustration is particularly challenging, as short texts rarely express this feeling directly. In the TOD-Frustration guidelines [51], frustration is defined not as "any negative emotion" but as the tension arising when a user's goal is blocked in a task-oriented setting. It can occur without strong language. Following Berkowitz's definition [52], frustration is an emotional state triggered by obstacles to need satisfaction. Accordingly, remarks like "The place was dirty" or unrelated insults do not qualify. Traditional sentiment models often detect anger but miss this goal-related tension, especially when expressed politely [51]. Based on these insights, we label a complaint as frustration only when two signals align: emotional tension and a clearly blocked user goal, typically indicated by harm or disruption (e.g., a torn coat, verbal abuse, long wait in freezing weather, etc.).

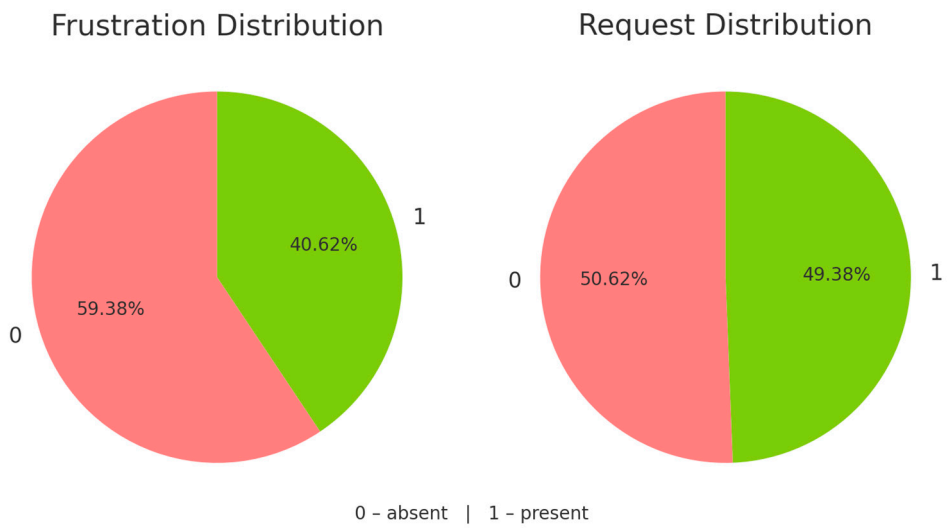
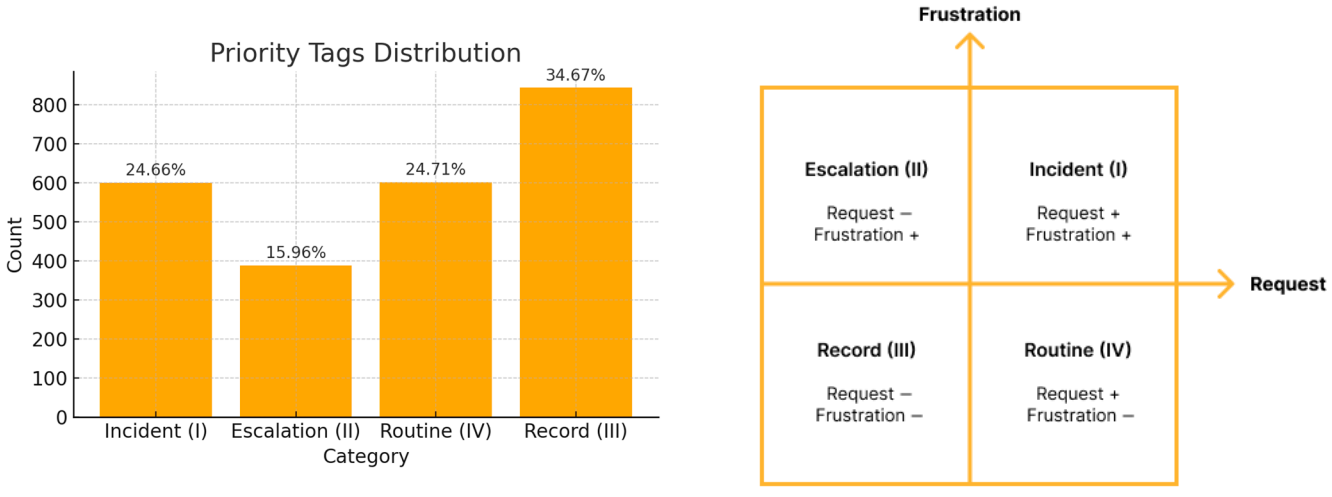


Figure 2. Distribution of Frustration and Request in the dataset.

The distribution of priority levels is nearly balanced (Figure 3a). These levels are derived directly from the four quadrants of the Frustration–Request plane (Figure 3b), with the rationale for their introduction as follows: Complaints that express both an explicit request and signs of frustration are classified as Incidents and routed immediately to the urgent escalation queue (Quadrant I). Complaints that convey frustration without a request become Escalations handled by the quality assurance team (Quadrant II). Complaints that contain an explicit request without frustration are considered Routine and are forwarded to the operational unit responsible for the service (Quadrant III). Finally, complaints showing neither frustration nor request are logged as Records for ongoing monitoring (Quadrant IV).



(**a**) (**b**)

Figure 3. (**a**) Distribution of Priority tags (levels) in the dataset. (**b**) Request–Frustration plane.

Although the Frustration and Request dimensions are categorical, this quadrant-based scheme introduces a fuzzy notion of urgency. Rather than imposing a strict ordinal ranking, we use co-occurrence patterns to approximate how the presence or absence of goal-related tension and explicit demands modulates perceived urgency. For instance, Incidents reflect both emotional tension and an operational ask, thereby justifying immediate routing. In contrast, Records indicate neither friction nor intent to escalate, warranting minimal attention. For example, “Please take an action! The driver braked sharply, and I fell, injuring

my arm" begins with an imperative directive in Searle's taxonomy [53] and implicitly signals frustration, thereby placing it in the Incident class.

Linguistically, the dataset is predominantly composed of messages written in Russian and Kazakh using the Cyrillic script, which accounts for approximately 84% of all complaints. The remaining 16% of messages exhibit a mix of Cyrillic and Latin characters. This mixing typically occurs in the form of English or Russian abbreviations (e.g., "QR", "м/a"—microbus, "a/м"—bus route), brand or company names ("Kaspi", "Expo"), or transliterated terms, reflecting the multilingual and multimodal nature of urban communication. Such code-mixing, although limited in scope, introduces additional complexity for tokenization and downstream processing. In terms of structure, the complaints are generally brief and concise. The average message length is approximately 110 characters. Only about 5% of messages exceed 250 words, making long-form complaints the exception rather than the norm. This brevity, combined with informal or fragmentary syntax, presents challenges for fine-grained semantic classification and emotion detection.

3.2. Universal Embeddings

Five universal embedding models are selected to support the study and address the stated research questions (Table 2). Among these five models, E5-large-instruct, BGE-M3, and LaBSE use bidirectional BERT backbones (560, 560, and 471 million parameters, respectively), while E5-mistral-7B-instruct and GTE-Qwen2-1.5B-instruct repurpose decoder-only LLMs (7.1 and 1.5 billion parameters, respectively). All of them, with the exception of E5-mistral-7B-instruct, are inherently multilingual. Because the base Mistral-7B-v0.1 was trained primarily on English data, E5-mistral-7B-instruct initialized from Mistral-7B-v0.1 is officially recommended for English use only. Nevertheless, we include E5-mistral-7B-instruct in our evaluation since it is fine-tuned on a mixture of multilingual datasets, which confers some cross-lingual capability.

Table 2. Universal Text-Embedding models selected for evaluation.

Model	Size	Language Coverage	Tuning Type	Embedding Dimension	Architecture
multilingual-E5-large-instruct	~560 M	100+	Instruction-tuned	1024	XLM-Roberta-large (BERT)
E5-mistral-7B-instruct	~7.1 B	English with residual multilingual capability	Instruction-tuned	4096	Mistral-7B (LLM)
GTE-Qwen2-1.5B-instruct	1.5 B	29	Instruction-tuned	1536	Qwen2-1.5B (LLM)
BGE-M3	~560 M	100+	Task-tuned	1024	XLM-Roberta-large (BERT)
LaBSE	471 M	109	Language agnostic	768	BERT-base (BERT)

Among the selected models, three are instruction-tuned: E5-Large-Instruct, E5-Mistral-7B-Instruct, and GTE-Qwen2-1.5B-Instruct. Instruction-tuned models are trained to incorporate natural language directives such as "Given a web search query, retrieve relevant passages" or "Classify a customer review as counterfactual or not" into their embedding process. At inference time, they expect queries to be prefixed with a specific instruction, while texts are typically encoded with service prefix "passage" or without any prefix. This

setup allows for the encoder to differentiate between query intent and document content, producing task-aware embeddings.

In contrast, task-tuned models specialize via supervised contrastive learning using dataset-specific labels or symbolic task tags, without ever seeing natural-language instructions. BGE-M3, following this paradigm, is fine-tuned on a multilingual mix of dense, sparse, and multi-vector retrieval objectives. During training, the query mode is internally marked, not via text; at inference, the model uses separate heads: the dense head for general semantic similarity, the multi-vector head for long documents, and the sparse head for lexical overlap.

Finally, LaBSE is neither instruction-tuned nor task-tuned. It is trained only with multilingual translation objectives and general masked-language modeling, on parallel-sentence pairs across 109 languages. Its dual BERT encoders are optimized to produce nearly identical vectors for meaning-equivalent translations. This yields content-driven, language-agnostic embeddings that generalize well to multilingual retrieval and clustering.

3.3. Zero-Shot Classification

3.3.1. Zero-Shot Classification Workflow

The zero-shot classification workflow is composed of three simple steps. Firstly, a semantic anchor is created for each class as a textual representation that encapsulates its most representative and distinctive features. Additionally, input complaint texts are preprocessed by attaching specific instruction prefixes, if needed. Secondly, embeddings are generated for both complaint texts and anchors using a universal embedding model. Thirdly, cosine similarity is computed between complaint embeddings and anchor embeddings. Finally, each complaint is assigned to the class whose anchor is nearest in the embedding space, as determined by cosine similarity (see Figure 4).

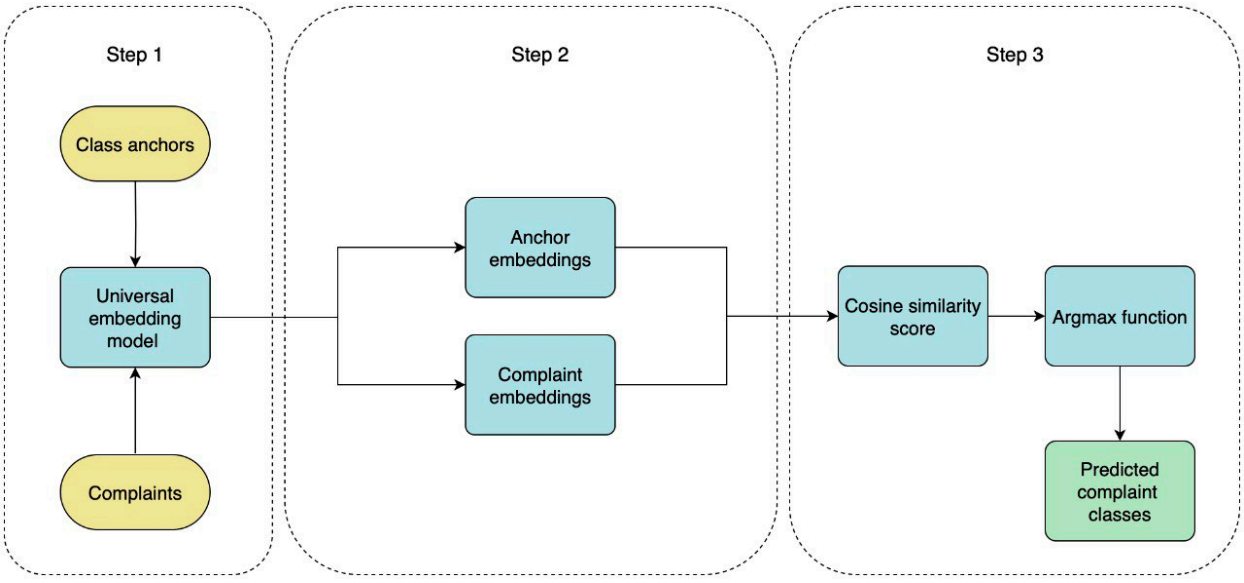


Figure 4. The zero-shot classification workflow for the Aspect facet.

If the facet is binary, as in the case of Frustration and Request, the final step of the workflow is slightly modified. In this case, only a single anchor is constructed for each facet, reflecting the presence of the corresponding state. The embedding of each complaint is compared to that of the anchor, and a value of 1 (indicating the state is present) is assigned if the similarity exceeds a predefined threshold. As the decision threshold, we adopt the mean cosine similarity calculated over the entire complaint dataset. A fixed, model-agnostic

cut-off is impractical because embedding models differ in their similarity scales: some tend to generate higher cosine scores, whereas others produce lower ones.

3.3.2. Anchor Construction

Anchors play a pivotal role in the zero-shot classification workflow because they serve as fixed reference points in the embedding space against which the position of each complaint is measured, as metaphorically illustrated in Figure 5. Following [54], we define an anchor as an extended textual description of a facet label. In this study, we construct each anchor as a list of facet-related words (see Table 3). For the Aspect facet, we construct eight semantic anchors, corresponding to the number of categories within this facet, and populate them with the most representative words. Therefore, when new long-tail categories emerge, there is no need to retrain the model; instead, it suffices to define a new anchor.

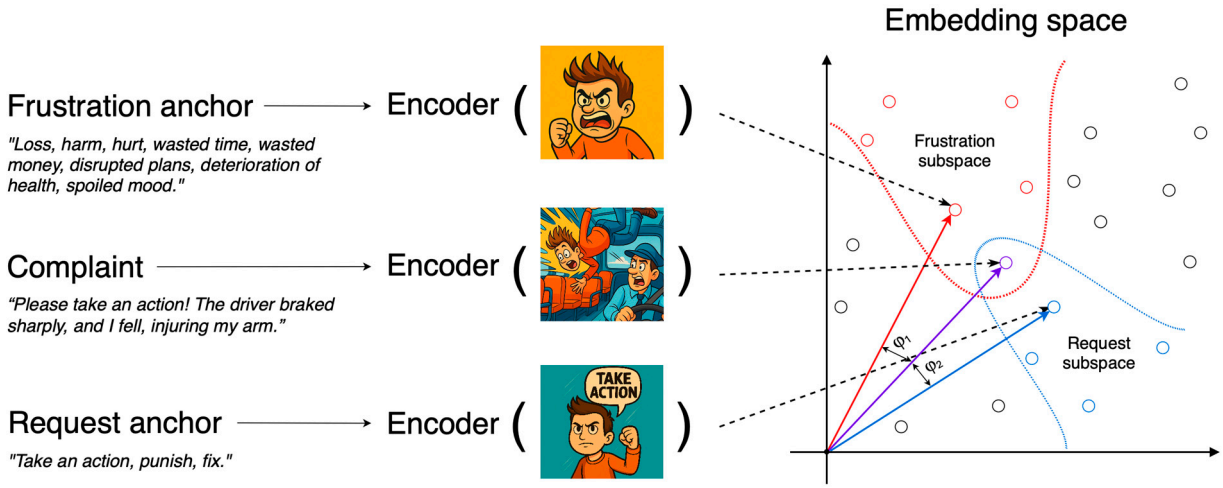


Figure 5. Metaphorical anchor positioning in embedding space.

Table 3. Full anchor texts for Aspect, Frustration, and Request facets.

No	Facet	Class Label	Anchor Text
1	Aspect	Condition	bus, interior, doors, windows, dirty, noisy, crowded, temperature
		Information	information board, electronic display, at the stop
		Operations	long, wait, route, late, skipped, dropped off
		Payment	double charge, payment, QR, terminal, amount
		Personnel	rude, driver, conductor, inspector, said, surname
		Roadside	manhole, parking, gate, snow, sidewalk
		Bike	bicycle, station, parked
		Lost	left, forgot, phone
2	Frustration	1	frustration, dissatisfaction, discontent, lateness
3	Request	1	request, demand, take, action, improvement, suggestion

A short list of "anchor words" looks like a bag of words, but once the list is passed through a modern encoder it becomes a single dense vector that captures contextual and distributional information the individual tokens lack. Because large-scale transformer encoders were pre-trained on billions of co-occurrence events, the resulting anchor vector embeds a whole semantic neighborhood: synonyms, paraphrases, topical cues, even pragmatic connotations. This makes the anchor behave like a class prototype rather than a literal

keyword filter. Prototype-style reasoning is well studied in few-/zero-shot learning [55]. Anchors follow the same principle, but the prototype is distilled from label semantics instead of support examples. The study in [56] automatically generates category anchors with an LLM and reports consistent gains on six benchmarks over sparse-keyword and entailment baselines. Similarly, the authors in [57] enrich hierarchical taxonomies by synthesizing new label descriptions, and the resulting prototype vectors achieve state-of-the-art performance in strict zero-shot settings.

Anchors, therefore, are not mere keyword lists but learned semantic beacons that exploit the inductive biases of pre-trained encoders. By operating in the same high-dimensional manifold as the model's internal representations, they deliver a principled, extensible, and empirically validated foundation for zero- and few-shot text classification. Herewith, anchors must strike a balance between precision and recall, they should include enough representative words to capture the full semantic range of a label without introducing extraneous words that add noise.

3.3.3. Input Text Formatting

Depending on the model, the input texts may be expanded by adding natural-language instruction and a role-specific prefix. For instance, for instruction-tuned encoders such as multilingual-E5-large-instruct and E5-mistral-7B-instruct, this formatting is essential. The developers of these models explicitly state that "queries must include instructions; otherwise, the model's performance degrades" [58]. One of the prefix configurations presented in Table 4 reproduces the exact format used during the pre-training and instruction tuning of the E5-large-Instruct model for the retrieval task. The other prefix configuration in Table 4 corresponds to symmetric query–query comparisons such as semantic similarity and no instruction line is required, mirroring the format used to train E5 models on this task. Applying these configurations is crucial for ensuring consistency between training and inference. To the best of our knowledge, no study has systematically compared which prefix configuration performs better for tasks like ours, i.e., tasks that can, broadly speaking, be mapped either to the asymmetric retrieval pattern or to the symmetric query–query pattern.

Table 4. Input text formats for the E5-instruct Model Family.

Side/Component	Recommended Prefix	Semantic Role in Training	Effect on Resulting Embedding
The retrieval task (asymmetric)			
Instruction	**Instruct:** + *one-sentence task description* (e.g., *"Given a web search query, retrieve relevant passages that answer the query\n"*)	Conveys task context	Shifts the query embedding toward the intended retrieval objective
Query	**Query:** + *anchor text*	Poses an information query	Embedding becomes concise and abstract, optimized to match relevant passages
Passage	**Passage:** + *complaint text*	Provides a comprehensive answer fragment	Embedding retains richer factual/contextual detail to satisfy queries

Table 4. *Cont.*

Side/Component	Recommended Prefix	Semantic Role in Training	Effect on Resulting Embedding
The semantic similarity task (symmetric)			
Instruction	**-**	Not used for symmetric tasks	-
Query 1	**query:** + *anchor text*	Serves as the first sentence in a similarity pair	Embedding encodes full semantics and is optimized to lie close to its counterpart when the pair is meaning-equivalent
Query 2	**query:** + *complaint text*	Serves as the second sentence in a similarity pair	Embedding likewise captures full semantics, aligning with Query 1 when the two sentences are paraphrases and separating when they are not

In contrast, models like BGE-M3 and LaBSE were not trained with instructions and therefore operate on raw, un-prefixed text. As is shown in their documentation, any added prefixes and prompts are ignored by these models and may degrade their performance. Nevertheless, our preliminary experiments reveal that, although BGE-M3 and LaBSE are officially instruction-free, prefixing inputs with the literal token query yields a small but consistent performance gain. We attribute this to the model's contrastive fine-tuning on query → passage pairs, so inserting a stable token at the beginning of the input restores a weak role signal that the encoder has implicitly learned to exploit. Similar prefix effects have been reported in the literature [59,60].

3.4. Lightweight Supervised Classification

3.4.1. Lightweight Classification Workflow

Zero-shot prototypes offer strong initial cues, yet they cannot resolve borderline cases. A lightweight supervised layer (head) on top of frozen embeddings provides two advantages: selective adjustment and computational economy. Lightweight classification keeps the encoder entirely frozen—the model continues to map every text to exactly the same vector it produced during pre-training—so its generalization capacity is left intact. On top of this static map, we train a single soft-max (or sigmoid, for binary facets) layer that learns new decision boundaries without altering the underlying geometry. Because the head contains roughly two-tenths of a percent of the backbone's parameters, optimization converges very fast on a modern CPU or GPU, and retraining is inexpensive whenever new annotated data emerge.

In this study, we use a single-layer logistic regression and fit the classification head with standard cross-entropy loss. This procedure is enough to nudge borderline cases across the correct side of the boundary and to compensate for label imbalance through learned bias terms. In practice, the light head raises macro-F1 by about 5–10 pp compared with the pure anchor-based baseline, while adding less than two megabytes of memory and virtually no inference latency [17]. The workflow of lightweight classification is depicted in Figure 6.

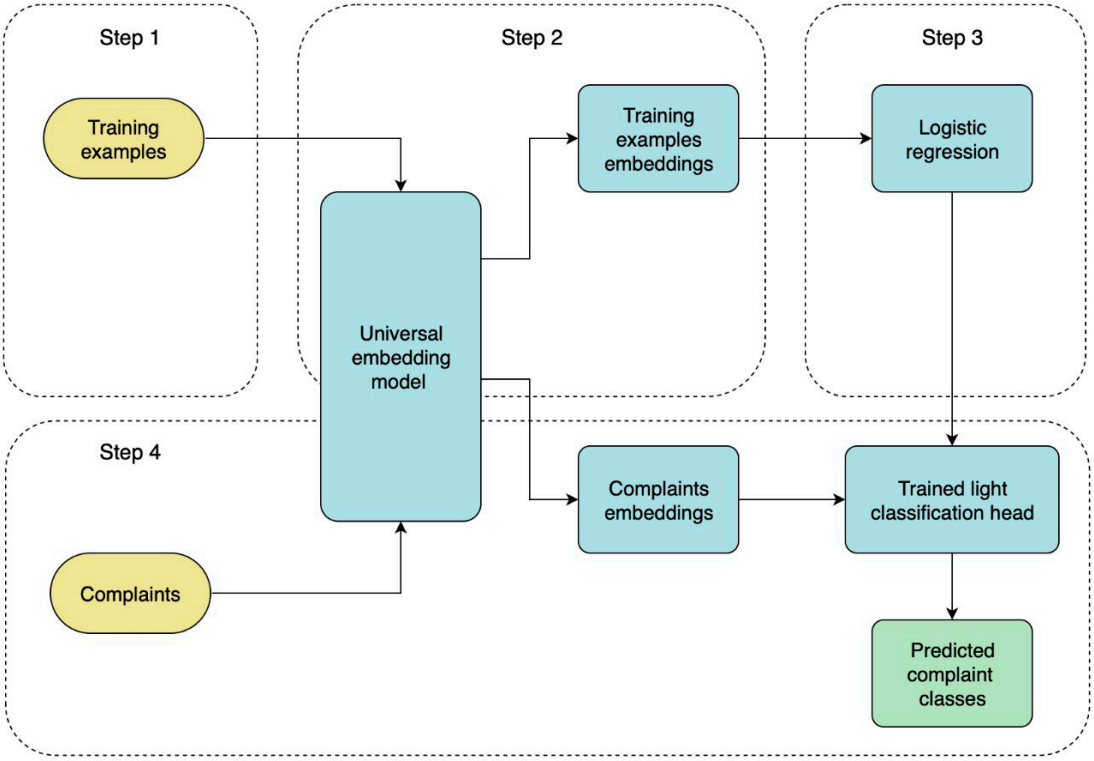


Figure 6. The lightweight supervised classification workflow.

3.4.2. Input Text Formatting in Lightweight Classification

Each input sequence retains the constant prefix "query:" to preserve the distributional assumptions of the embedding space, while the natural-language instruction line used in zero-shot experiments is omitted. This fixed prefix prevents representation drift between training and inference, allowing for the lightweight head to learn a stable linear decision boundary.

4. Results and Analysis

4.1. Service Aspect Classification

The first facet in our multi-faceted complaint analysis is the service aspect, i.e., the operational domain targeted by the passengers' grievance, such as bus condition, payment issues, personnel behavior, etc. Because aspect categories are distributed very unevenly, this becomes a highly imbalanced multi-class classification problem. An extreme case is the Lost category, which appears only once in the entire dataset. That lone instance fell into the training portion of the 60/40 split, leaving the test set without any Lost examples; as a result, precision, recall, and F1 cannot be reported for this class in the lightweight-supervised evaluation. To counteract imbalance in the lightweight supervision, we employ inverse-class-frequency weighting in the cross-entropy loss. By scaling each example's loss inversely to its class prevalence, this widely adopted technique equalizes the expected gradient contribution of all classes and reduces bias toward dominant ones [61]. The zero-shot configuration, by contrast, involves no gradient-based learning and is therefore unaffected by skews in the training data, although rare classes can still perform worse if their semantic anchors are ambiguous or overlap with more frequent categories. Finally, all evaluation metrics are macro-averaged, ensuring that minority classes remain proportionally represented in the aggregate scores.

Tables 5–9 report the classification performance of the evaluated models under two settings: a zero-shot configuration and a lightweight supervised configuration, where a logistic

regression head is trained on the frozen embeddings using a 60%/40% train–test split. Performance is reported in terms of precision (P), recall (R), and F1 score for each class, along with overall accuracy. We also ran a stratified five-fold cross-validation; performance differed by less than 0.5 pp. from the 60/40 split (Appendix A.1).

Table 5. The performance of multilingual-E5-large-instruct ("Query:"–"Query:" configuration).

Aspect	Zero-Shot Classification				Lightweight Classification			
	Count	P	R	F1	Count	P	R	F1
Bike	68	0.980	0.735	0.840	29	0.933	0.966	0.949
Condition	151	0.417	0.669	0.514	60	0.697	0.767	0.730
Information	65	0.516	0.754	0.613	20	0.680	0.850	0.756
Lost	1	0.143	1.000	0.250	-	-	-	-
Operations	1305	0.807	0.918	0.859	524	0.922	0.865	0.892
Payment	230	0.916	0.761	0.831	96	0.922	0.865	0.892
Personnel	560	0.880	0.473	0.616	223	0.841	0.874	0.857
Roadside	20	0.536	0.750	0.625	8	0.714	0.625	0.667
Total	2400	Accuracy: 0.772			960	Accuracy: 0.895		

Table 6. The performance of E5-mistral-7b-instruct ("Query:"–"Query:" configuration).

Aspect	Zero-Shot Classification				Lightweight Classification			
	Count	P	R	F1	Count	P	R	F1
Bike	68	0.938	0.882	0.909	29	0.966	0.966	0.966
Condition	151	0.361	0.656	0.466	60	0.746	0.733	0.739
Information	65	0.481	0.800	0.601	20	0.652	0.750	0.698
Lost	1	0.000	0.000	0.000	-	-	-	-
Operations	1305	0.831	0.857	0.843	524	0.938	0.931	0.935
Payment	230	0.934	0.796	0.859	96	0.946	0.906	0.926
Personnel	560	0.832	0.568	0.675	223	0.838	0.857	0.847
Roadside	20	0.556	0.750	0.638	8	0.667	0.750	0.706
Total	2400	Accuracy: 0.769			960	Accuracy: 0.895		

Table 7. The performance of gte-Qwen2-1.5B-instruct ("Query:"–"Query:" configuration).

Aspect	Zero-Shot Classification				Lightweight Classification			
	Count	P	R	F1	Count	P	R	F1
Bike	68	0.882	0.882	0.882	29	0.964	0.931	0.947
Condition	151	0.269	0.450	0.337	60	0.667	0.700	0.683
Information	65	0.333	0.908	0.488	20	0.727	0.800	0.762
Lost	1	0.043	1.000	0.083	-	-	-	-
Operations	1305	0.766	0.904	0.829	524	0.943	0.908	0.925
Payment	230	0.967	0.635	0.766	96	0.936	0.917	0.926

Table 7. *Cont.*

Aspect	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
Personnel	560	0.863	0.248	0.386	223	0.786	0.857	0.820
Roadside	20	0.577	0.750	0.652	8	0.800	0.500	0.615
Total	2400	Accuracy: 0.695			960	Accuracy: 0.879		

Table 8. The performance of BGE-M3 ("Query:" before anchors configuration).

Aspect	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
Bike	68	0.937	0.868	0.901	29	0.963	0.897	0.929
Condition	151	0.688	0.291	0.409	60	0.708	0.767	0.736
Information	65	0.355	0.769	0.485	20	0.727	0.800	0.762
Lost	1	0.077	1.000	0.143	-	-	-	-
Operations	1305	0.669	0.969	0.792	524	0.955	0.924	0.939
Payment	230	0.970	0.700	0.813	96	0.917	0.917	0.917
Personnel	560	0.942	0.087	0.160	223	0.843	0.892	0.867
Roadside	20	0.923	0.600	0.727	8	0.714	0.625	0.667
Total	2400	Accuracy: 0.683			960	Accuracy: 0.9		

Table 9. The performance of LaBSE ("Query:"–"Query:" configuration).

Aspect	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
Bike	68	0.312	0.794	0.448	29	0.900	0.931	0.915
Condition	151	0.223	0.656	0.333	60	0.535	0.767	0.630
Information	65	0.479	0.523	0.500	20	0.600	0.900	0.720
Lost	1	0.000	0.000	0.000	-	-	-	-
Operations	1305	0.774	0.788	0.781	524	0.947	0.912	0.929
Payment	230	0.867	0.396	0.543	96	0.943	0.865	0.902
Personnel	560	0.778	0.087	0.157	223	0.833	0.803	0.817
Roadside	20	0.084	0.750	0.151	8	0.500	0.375	0.429
Total	2400	Accuracy: 0.571			960	Accuracy: 0.869		

The best-performing strategy for every encoder we tested (even those that are not instruction-tuned) is to prepend the prefix "query:" to both the anchor text and the complaint text. All other prefix or instruction variants lowered accuracy by 2–5 pp. The only exception is BGE-M3: it still benefits from the "query:" prefix in front of anchors, but accuracy drops if the same prefix is added to the complaint text. For instance, for the E5-large-instruct encoder, accuracy is 0.693 when neither instructions nor role prefixes are added (Figure 7b). If we prepend the developer-recommended instruction line together with the prefixes "Query:" before the anchor and "Passage:" before the complaint, accuracy rises to 0.740. The highest score, 0.772, is obtained when we keep only the role prefixes

and omit the instruction line (Figure 7a). The complete 16-configuration prefix/instruction grid is provided in Appendix A.3. This design space captures all schemes reported in the literature for E5-style and BGE/LaBSE encoders.

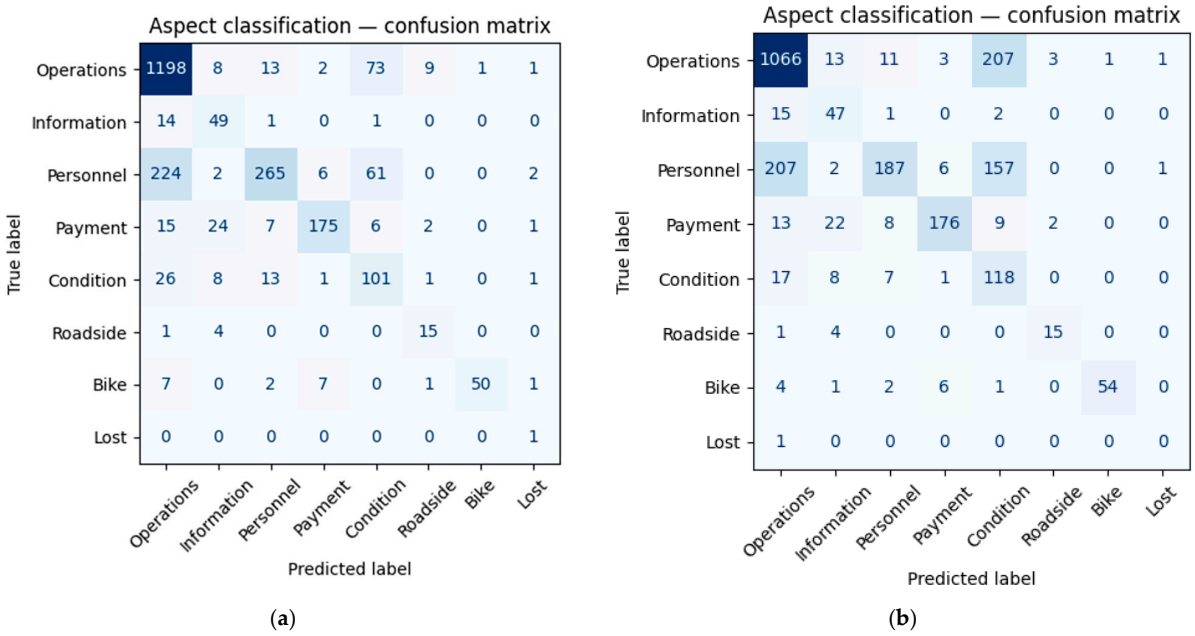


Figure 7. The confusion matrix for multilingual-E5-large-instruct: (**a**) "Query:"–"Query:" configuration; (**b**) "No prefixes" configuration.

As Figure 8 shows, instruction-tuned E5 models start from the strongest zero-shot baseline and gain a moderate boost (12–13 pp) once the lightweight head is added. BGE-M3 and especially LaBSE begin much lower, but both benefit disproportionately from the logistic head, recovering 18–30 pp and closing most of the gap to E5. Instruction-tuned models already position the anchor prompts in task-relevant regions of the embedding space, so a linear probe can add only marginal refinements. LaBSE and BGE-M3 leave these task boundaries latent; a lightweight logistic head uncovers them, producing the much larger performance jump.

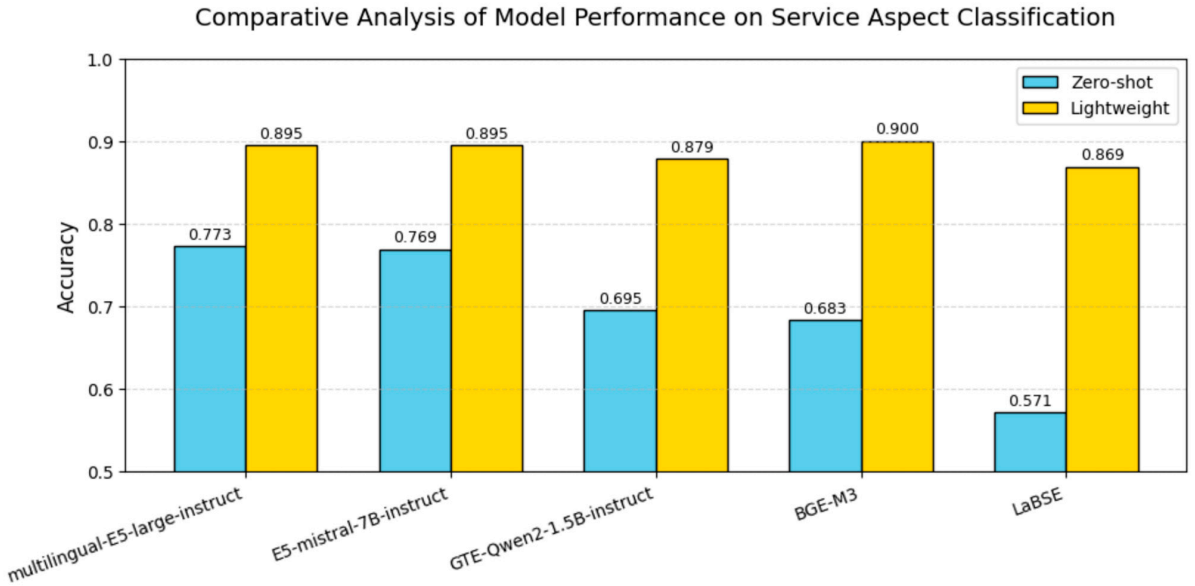


Figure 8. Model performance comparison on Service Aspect classification.

Figures 9–11 show learning curves of the lightweight classifiers built on E5-large-instruct, BGE-M3 and LaBSE, respectively. Each plot shows macro-F1 for both the training and validation sets as the amount of labelled data grows. In every case, the training score remains near 0.90–0.95 from the start, whereas the validation score begins around 0.52–0.55 and climbs steadily. A pronounced rise appears once the sample reaches roughly 800 complaints. At that point, even the rarest labels are present in every cross-validation fold, so macro-averaging no longer penalizes missing classes and the curves continue to improve together. After this threshold, the distance between the two lines stabilizes, indicating that the small logistic head is reducing bias rather than memorizing noise. The average generalization gaps (0.148 ± 0.102 for E5, 0.166 ± 0.118 for BGE-M3, and 0.207 ± 0.108 for LaBSE) remain well below the levels that would signal overfitting. A minimal amount of supervised training therefore suffices to align the decision boundary with the actual class structure while preserving sound generalization.

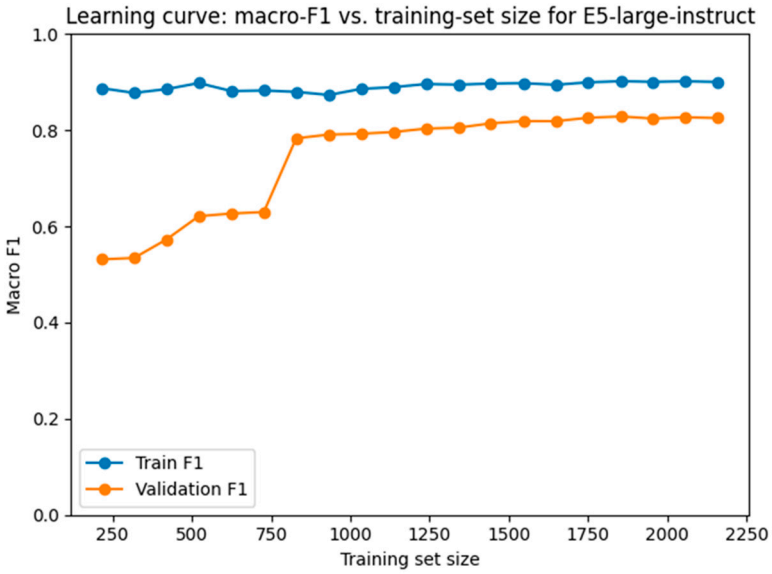


Figure 9. Learning curve for the lightweight logistic head on top of E5-large-instruct.

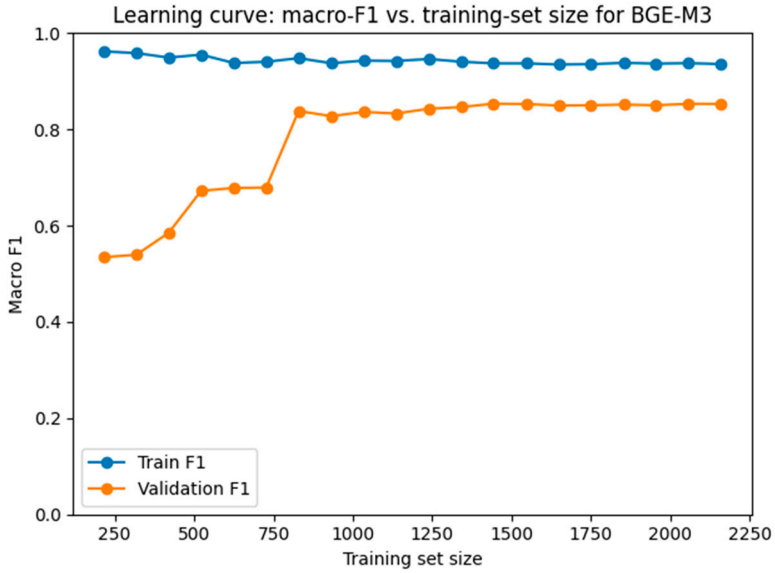


Figure 10. Learning curve for the lightweight logistic head on top of BGE-M3.

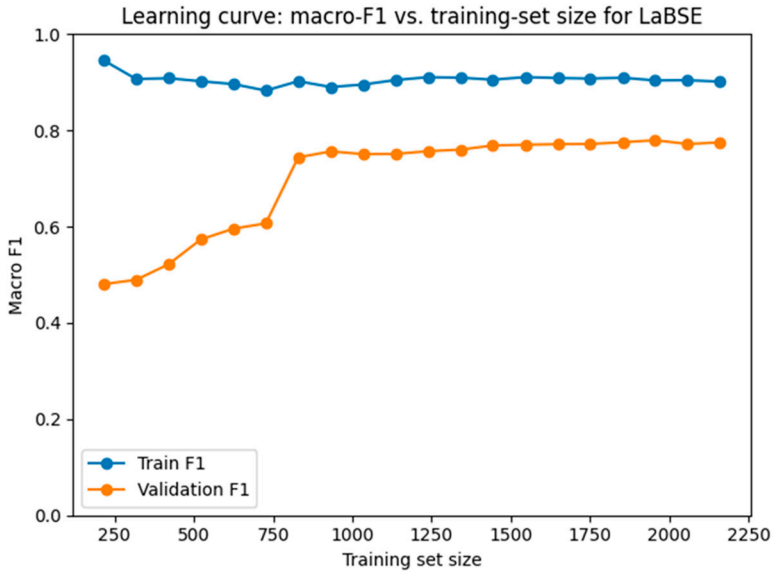


Figure 11. Learning curve for the lightweight logistic head on top of LaBSE.

LaBSE and BGE-M3 improve more sharply than E5. This is because zero-shot E5 anchors are already well aligned with its instruction-tuned geometry, so its error is dominated by residual variance, not bias. BGE-M3 and LaBSE, in contrast, encode the same semantic distinctions but place anchor vectors in systematically displaced regions (BGE because of task-tuning; LaBSE because translation training compresses cosine ranges). These are classic underfitting artefacts, the representational space is informative, yet the hyperplane that separates classes is mispositioned. A small supervised adjustment therefore yields a larger marginal benefit.

Empirical evidence supports this view. After training the head, the mean generalization gap remains below 0.20 macro-F1 for all three models, and cross-validated variance shrinks rather than grows, indicating that the new parameters are not memorizing idiosyncrasies but correcting a systematic offset. In short, the post-training surge of BGE-M3 and LaBSE is best explained by bias correction of an initially underfit linear decision rule, not by overfitting. E5 shows a smaller gain precisely because its zero-shot bias was already low. We did not generate learning curve figures for GTE-Qwen2-1.5B-instruct and E5-mistral-7B-instruct because their training-and-validation trajectories mirrored the intermediate pattern already illustrated for the other encoders.

Therefore, E5 remains the most effective model in the pure zero-shot setting because its instruction prompts and internal prefixes are already tailored to the facet extraction task, placing the initial decision boundary close to the optimum. BGE-M3 surpasses E5 once the lightweight classifier is trained, as its embeddings are slightly misaligned. The logistic layer compensates for this bias by recalibrating the decision thresholds, thereby unveiling the latent discriminatory signal. LaBSE approaches but does not exceed E5, since its translation-oriented training compresses cosine distances and obscures several class margins. Although the lightweight head expands this scale and introduces class-specific biases, some rare categories (most notably Roadside) still underperform relative to E5. Class imbalance leaves only a handful of Roadside examples, so the head's weights are estimated with high variance. LaBSE's post-training rescaling amplifies that variance and depresses F1, whereas E5's better-separated geometry still isolates each minority instance, giving a cleaner hyperplane and higher recall.

4.2. Frustration and Request Detection

For aspect classification, every model performed just as well with purely English anchors; adding Kazakh or Russian words to the anchor texts produced no measurable gain, despite the fact that all complaints are written in Russian or Kazakh. By contrast, for the binary facets Frustration and Request, enriching the anchors with Kazakh and Russian terms led to a clear improvement in accuracy. Therefore, we use the anchor "Frustration, фрустрация, Расстройство, Қорлау, Причинение вреда, Damnification, Опоздание" for frustration detection, and the anchor "Request, Просьба, Заявка, Примите, Меры, Шара, Improvement, Suggestion" for request detection.

Moreover, for aspect classification, every model performed just as well with the symmetric Query–Query format. By contrast, for the binary facets Frustration and Request the symmetric Query–Query format proved ineffective. We therefore switched to an asymmetric "Instruction → Query → Passage" configuration for the instruction-tuned models, which yielded substantially better results. During preliminary experiments we tested several alternative instructions, but the developer-recommended instruction "Instruct: Given a web search query, retrieve relevant passages that answer the query" added before anchors consistently delivered the best performance. For the remaining two models BGE-M3 and LaBSE the most effective setup is to prepend "Query:" to the anchor texts while leaving the complaint texts un-prefixed.

Tables 10–14 report the classification performance of the evaluated models under two settings: a zero-shot configuration and a lightweight supervised configuration, where a logistic regression head is trained on the frozen embeddings using a 60%/40% train–test split. Performance is reported in terms of precision (P), recall (R), and F1 score for each class, along with overall accuracy. All models achieve higher scores on the Request facet and somewhat lower scores on Frustration, an expected pattern given that requests are expressed explicitly in the text whereas frustration is usually conveyed implicitly.

Table 10. The performance of multilingual-E5-large-instruct for Frustration and Request detection.

Facet	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
Frustration								
0	1428	0.818	0.709	0.759	575	0.828	0.906	0.865
1	972	0.642	0.769	0.700	385	0.837	0.719	0.774
Total	2400		Accuracy: 0.733		960		Accuracy: 0.831	
Request								
0	1215	0.828	0.770	0.798	497	0.887	0.821	0.853
1	1185	0.780	0.836	0.807	463	0.822	0.888	0.854
Total	2400		Accuracy: 0.803		960		Accuracy: 0.853	

Table 11. The performance of E5-mistral-7b-instruct for Frustration and Request detection.

Facet	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
Frustration								
0	1428	0.784	0.662	0.718	575	0.825	0.845	0.835
1	972	0.595	0.731	0.657	385	0.760	0.732	0.746

Table 11. *Cont.*

Facet	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
Total	2400		Accuracy: 0.690		960		Accuracy: 0.800	
				Request				
0	1215	0.761	0.740	0.750	497	0.870	0.805	0.836
1	1185	0.741	0.761	0.751	463	0.806	0.870	0.837
Total	2400		Accuracy: 0.750		960		Accuracy: 0.836	

Table 12. The performance of gte-Qwen2-1.5B-instruct for Frustration and Request detection.

Facet	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
				Frustration				
0	1428	0.767	0.704	0.734	575	0.804	0.857	0.830
1	972	0.612	0.685	0.647	385	0.764	0.688	0.724
Total	2400		Accuracy: 0.697		960		Accuracy: 0.790	
				Request				
0	1215	0.712	0.726	0.719	497	0.823	0.813	0.818
1	1185	0.713	0.699	0.706	463	0.802	0.812	0.807
Total	2400		Accuracy: 0.713		960		Accuracy: 0.812	

Table 13. The performance of BGE-M3 for Frustration and Request detection.

Facet	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
				Frustration				
0	1428	0.727	0.629	0.674	575	0.844	0.877	0.860
1	972	0.545	0.652	0.594	385	0.804	0.758	0.781
Total	2400		Accuracy: 0.638		960		Accuracy: 0.829	
				Request				
0	1215	0.771	0.763	0.767	497	0.871	0.869	0.870
1	1185	0.759	0.767	0.763	463	0.860	0.862	0.861
Total	2400		Accuracy: 0.675		960		Accuracy: 0.866	

Table 14. The performance of LaBSE for Frustration and Request detection.

Facet	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
				Frustration				
0	1428	0.658	0.532	0.588	575	0.833	0.868	0.850
1	972	0.463	0.594	0.521	385	0.789	0.740	0.764
Total	2400		Accuracy: 0.557		960		Accuracy: 0.817	

Table 14. *Cont.*

Facet	Zero-Shot Classification				Lightweight Classification			
	Count	P	R	F1	Count	P	R	F1
				Request				
0	1215	0.583	0.597	0.590	497	0.909	0.859	0.883
1	1185	0.576	0.563	0.570	463	0.857	0.907	0.881
Total	2400		Accuracy: 0.580		960		Accuracy: 0.882	

As Figure 12 shows, the "bigger boost for lower baseline" trend persists, yet absolute accuracies diverge: detecting explicit requests remains consistently easier and still leads by roughly 3–5 pp, even after the lightweight head is applied. The most striking feature of Figure 12 is the zero-shot performance of multilingual-E5-large-instruct. Even without a single task-specific parameter update, E5-large-instruct approaches the mid 80% range for accuracy, underscoring how thoroughly instruction tuning has aligned its semantic space with the cues that signal frustration and explicit requests. The other encoders, in contrast, require a lightweight logistic head to catch up, and although the resulting gains are impressive (BGE-M3 and LaBSE surge by nearly 30 percentage points), the sheer magnitude and speed of that improvement invite a skeptical glance. Such jumps raise the possibility that the linear probe is exploiting idiosyncratic patterns in the limited training split rather than discovering genuinely task-general structure, an issue that warrants closer scrutiny through cross-validation before those numbers can be considered as robust as E5's out-of-the-box performance.

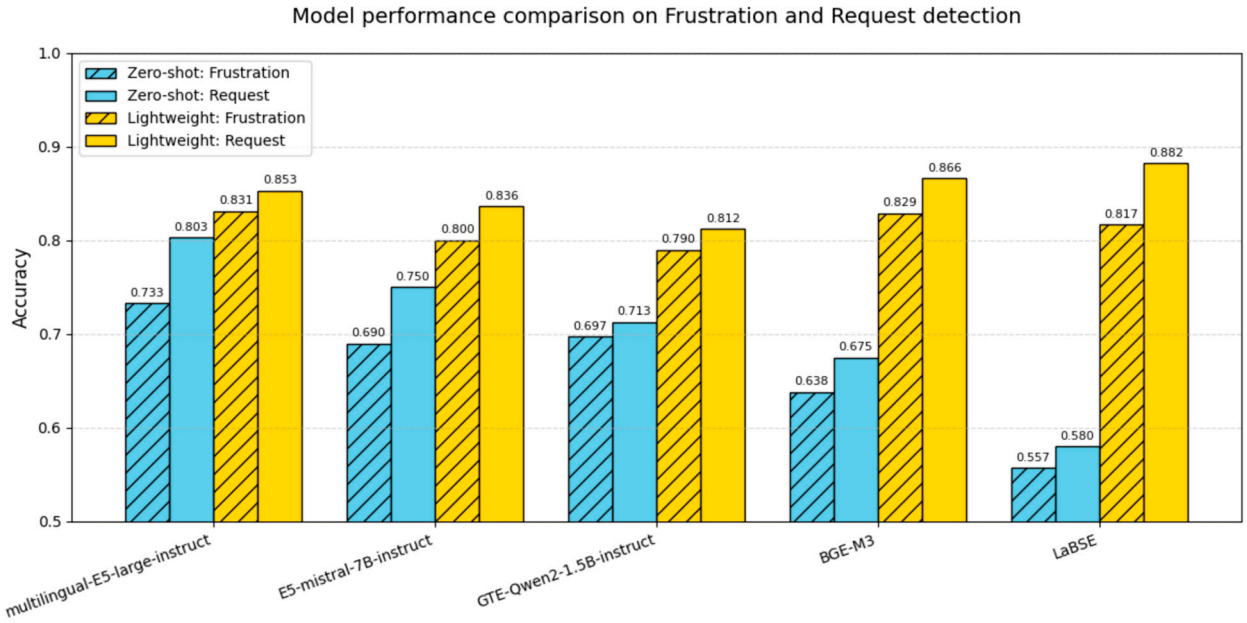


Figure 12. Model performance comparison on Frustration and Request detection.

In zero-shot classification, E5-large-instruct remains the top performer, and, somewhat unexpectedly, its much larger sibling E5-mistral-7B-instruct ($\approx 14 \times$ more parameters) fails to surpass it in any of the three facets. The key difference is architectural: E5-mistral is built on a causal decoder backbone, so its sentence vector is extracted from an autoregressive representation that entangles semantic content with next-token-prediction cues, producing noisier cosine geometry. Recent comparative work on LLM-based embeddings shows that, even under the same contrastive objectives, decoder-only models lag behind encoder-

only models on short-text retrieval and classification because bidirectional context and purpose-built pooling yield cleaner class boundaries [62,63]. E5-large, being an encoder-only transformer trained end-to-end for instruction-driven passage ranking, thus retains a more task-aligned embedding space and outperforms its parameter-rich but architecturally mismatched counterpart.

Compared with the imbalanced Aspect facet, the more balanced Request facet allows for LaBSE and BGE-M3, after lightweight-classifier training, to outperform E5-large-instruct. Because no class suffers from severe data scarcity, the logistic head can exploit each encoder's full cosine geometry. Therefore, LaBSE gains from scaling its translation-compressed space, while BGE-M3 benefits from shifting its anchors toward a query-centered region. This redistribution of bias and scale yields a sharper decision boundary for both models than for E5, whose zero-firing alignment has already reached its performance ceiling.

However, within the Frustration facet, E5-large-instruct still holds the lead even though the class distribution here is balanced. This advantage stems from the intrinsically implicit and stylistically diffuse nature of frustration cues. Because E5 was originally instruction-tuned on a diverse set of tasks that explicitly feature complaints, indirect expressions of dissatisfaction, and evaluative modal constructions, its embedding space already contains a latent "dissatisfaction axis." The lightweight logistic layer therefore needs only minimal threshold adjustments to achieve effective separation. For example, BGE-M3's retrieval-focused training arranges sentences primarily by topical similarity rather than by emotional tone.

4.3. Priority Level Identification

The priority grid is derived from the logical combination of two binary facets: Request (mainly explicit) and Frustration (mainly implicit). In other words, we do not employ evaluated models to predict the Priority class directly. Nevertheless, we report priority level identification performance within the framework of each evaluated model, because it is directly shaped by the accuracy of the preceding facet-detection stage. Tables 15–19 report the Priority level identification performance within the context of evaluated models under two settings: a zero-shot configuration and a lightweight supervised configuration which uses a 60%/40% train–test split. Performance is reported in terms of precision (P), recall (R), and F1 score for each Priority level, along with overall accuracy.

Table 15. The performance of multilingual-E5-large-instruct for Priority identification.

Priority Category	After Zero-Shot Classification				After Lightweight Classification			
	Count	P	R	F1	Count	P	R	F1
Incident (I)	591	0.477	0.717	0.573	228	0.617	0.693	0.653
Escalation (II)	381	0.493	0.354	0.412	157	0.733	0.350	0.474
Routine (IV)	594	0.622	0.399	0.486	235	0.668	0.694	0.681
Record (III)	834	0.743	0.763	0.753	340	0.787	0.891	0.836
Total	2400	Accuracy: 0.597			960	Accuracy: 0.707		

Table 16. The performance of E5-mistral-7b-instruct for Priority level identification.

Priority Category	After Zero-Shot Classification				After Lightweight Classification			
	Count	P	R	F1	Count	P	R	F1
Incident (I)	591	0.434	0.660	0.523	228	0.555	0.711	0.623
Escalation (II)	381	0.332	0.257	0.290	157	0.646	0.325	0.432

Table 16. *Cont.*

Priority Category	After Zero-Shot Classification				After Lightweight Classification			
	Count	P	R	F1	Count	P	R	F1
Routine (IV)	594	0.561	0.301	0.392	235	0.688	0.609	0.646
Record (III)	834	0.662	0.704	0.682	340	0.766	0.859	0.810
Total	2400	Accuracy: 0.522			960	Accuracy: 0.675		

Table 17. The performance of gte-Qwen2-1.5B-instruct for Priority level identification.

Priority Category	After Zero-Shot Classification				After Lightweight Classification			
	Count	P	R	F1	Count	P	R	F1
Incident (I)	591	0.478	0.613	0.537	228	0.562	0.636	0.597
Escalation (II)	381	0.263	0.228	0.244	157	0.607	0.344	0.439
Routine (IV)	594	0.530	0.360	0.429	235	0.645	0.579	0.610
Record (III)	834	0.618	0.673	0.644	340	0.721	0.853	0.782
Total	2400	Accuracy: 0.510			960	Accuracy: 0.651		

Table 18. The performance of BGE-M3 for Priority level identification.

Priority Category	After Zero-Shot Classification				After Lightweight Classification			
	Count	P	R	F1	Count	P	R	F1
Incident (I)	591	0.411	0.575	0.479	228	0.651	0.671	0.661
Escalation (II)	381	0.324	0.286	0.304	157	0.617	0.503	0.554
Routine (IV)	594	0.550	0.342	0.422	235	0.712	0.694	0.703
Record (III)	834	0.648	0.674	0.661	340	0.796	0.862	0.828
Total	2400	Accuracy: 0.506			960	Accuracy: 0.717		

Table 19. The performance of LaBSE for Priority level identification.

Priority Category	After Zero-Shot Classification				After Lightweight Classification			
	Count	P	R	F1	Count	P	R	F1
Incident (I)	591	0.309	0.443	0.364	228	0.623	0.702	0.660
Escalation (II)	381	0.246	0.257	0.252	157	0.654	0.433	0.521
Routine (IV)	594	0.468	0.244	0.321	235	0.712	0.706	0.709
Record (III)	834	0.394	0.399	0.397	340	0.817	0.879	0.847
Total	2400	Accuracy: 0.349			960	Accuracy: 0.722		

Accuracy rises for every model once the lightweight head is added, but the size of the gain again follows the "bigger boost for lower baseline" rule (see Figure 13). LaBSE, whose zero-shot score is the lowest, jumps the most—37 pp. The residual spread between models therefore compresses from 25 pp in the zero-shot setting to just 1 pp after lightweight adaptation, showing that miscalibration in the facet detectors, rather than the quadrants themselves, is what limits priority-level accuracy.

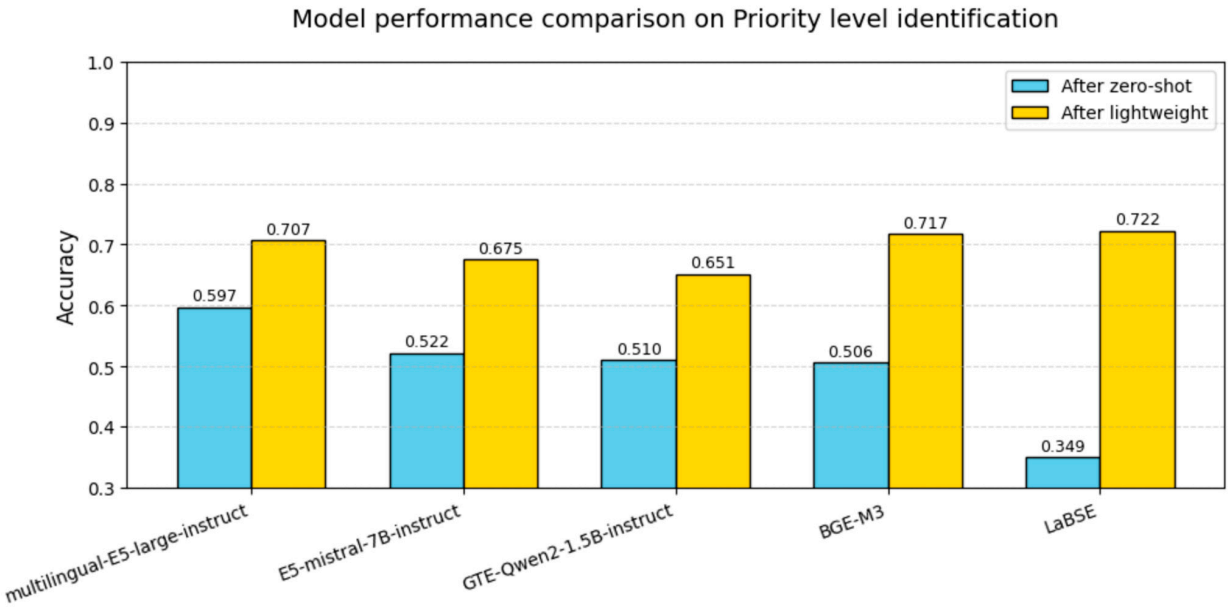


Figure 13. Model performance comparison on Priority identification.

Priority level identification depends entirely on the upstream decisions for Request and Frustration, so its ceiling is fixed by the weaker of those two signals. Most misclassifications occur between Escalation vs. Incident and Routine vs. Incident because each of those pairs shares one overt signal with Incident (either a request or frustration) and differs only by the subtler, harder-to-detect second cue (Figure 14a). After we train the lightweight logistic head, predictions for Routine improve substantially: the head learns a sharper threshold for the explicit Request flag and corrects many Routine → Incident slips. Escalation, however, remains the main source of confusion, since it is distinguished from Incident solely by latent frustration cues that are context-dependent and much noisier than explicit requests; the linear head cannot fully resolve that ambiguity (Figure 14b). Further progress therefore hinges on sharpening the Frustration detector (e.g., through targeted annotation or contrastive augmentation) rather than on redesigning the priority grid or adding a deeper head.

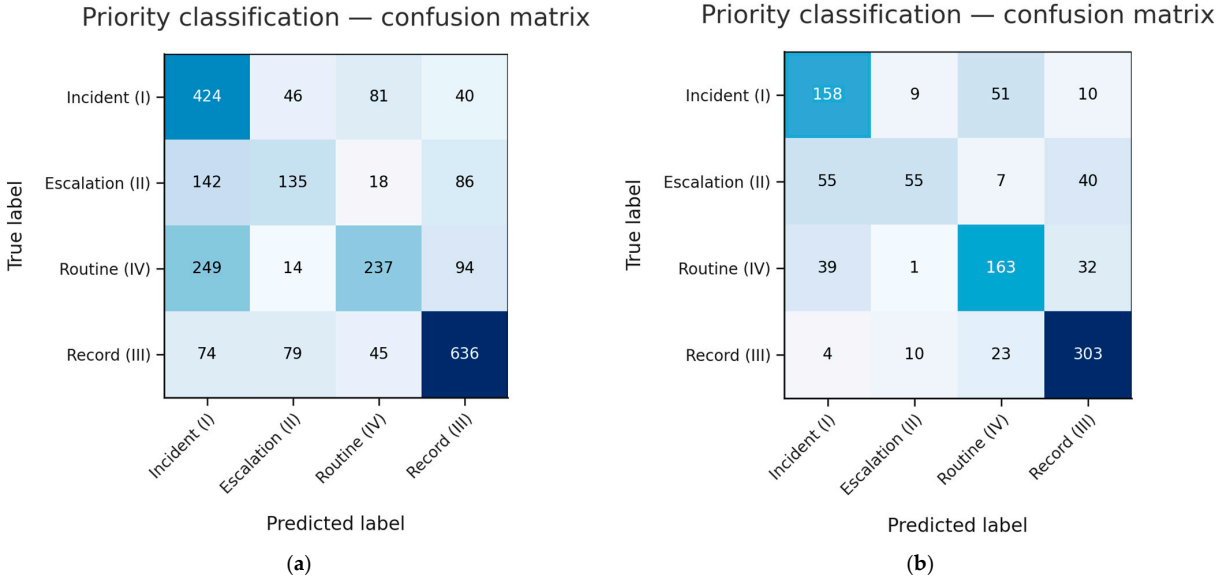


Figure 14. The confusion matrix for Priority level identification after applying the multilingual-E5-large-instruct model for Frustration and Request detection: (**a**) the zero-shot configuration; (**b**) the lightweight classification.

Although each encoder performs well on the separate binary tasks of Frustration and Request detection, their combined outputs yield markedly lower accuracy in the four-level priority scheme. The decline reflects classic error propagation. Even a single misclassification can push a message into the wrong quadrant, especially among the semantically adjacent categories Incident, Escalation, and Routine, while fixed thresholds and the lack of adaptation to dataset statistics make the mapping brittle and systematically biased. Error accumulation in multi-stage NLP pipelines has been examined extensively [64–66]. Whenever a downstream component relies on upstream predictions, early mistakes are magnified, degrading overall performance. Joint or multi-task learning mitigates this by aligning decision boundaries through shared parameters, whereas lightweight meta-classifiers trained on raw logits reconcile conflicting signals without sacrificing modularity.

5. Discussion

5.1. Key Findings and Implications

This study demonstrates that modern universal embedding models can underpin a practical, fully frozen pipeline for fine-grained complaint analytics. Without any task-specific fine-tuning, these models generate dense representations that cleanly separate complaint facets. This enables multilingual scalability and reproducibility across domains, even when labeled data is limited or unavailable. Operationally, such a frozen pipeline offers a pragmatic alternative to heavyweight task-specific models or costly LLM calls, achieving competitive accuracy while preserving interpretability and cost efficiency.

The potential of universal embeddings is also substantiated by the earlier study [67], which benchmarked BGE-M3 and multilingual-E5-large-instruct against few-shot LLMs on the same complaint corpus. Unlike our work, that study addressed only a single-facet, seven-class taxonomy and did not employ anchor-based prompting. Nevertheless, modest supervised fine-tuning lifted BGE-M3 and E5 to about 93% accuracy, matching the best LLM while operating at a fraction of the computational cost and latency.

Building on these comparative insights, we now turn to the two research questions that frame this study:

- RQ 1: Zero-shot capability of universal embeddings. With correct input formatting, the best encoder achieves approximately 77% accuracy for Aspect classification, 74% for Frustration detection, and 80% for Request detection in a pure zero-shot setting. Thus, universal embeddings alone can reach ~75–80% accuracy on multilingual complaints.
- RQ 2: Benefit of a lightweight head. Adding a single logistic-regression head raises performance by 12–30 pp, bringing all encoders to 89% accuracy for Aspect, 83–87 for the binary facets, and 72% accuracy on priority levels, while adding negligible computational cost.

Moreover, three key observations stand out. First, zero-shot quality depends more on format alignment than on raw model size. All encoders (whether instruction-tuned or not) benefit when the input texts reproduce the prefix scheme seen in training. E5-based models reach a strong baseline once the "query:" role token is applied symmetrically; adding the long natural-language Instruct line is unnecessary and even harms aspect accuracy by 2–3 pp. Conversely, BGE-M3 and LaBSE, which were released as "instruction-free," still gain 3–5 pp from a single query: token in front of anchors but lose accuracy when the same token precedes complaints. These effects confirm that even minor mismatches in prompt geometry can tilt the cosine scale and misplace complaints relative to their anchors.

Second, a classification head closes most of the performance gap, especially for models whose embeddings are less aligned with the domain or language mix. The logistic head contributes only 0.1% of the parameters yet lifts macro-F1 by 12 pp for the E5 family, 18–22 pp for BGE-M3 and GTE-Qwen2, and 30 pp for LaBSE. Because the encoder remains

frozen, these gains must come from better calibration of the decision surface: the head re-scales the dense similarity values, shifts biases for rare classes, and combines information from multiple anchors rather than relying on a hard nearest prototype rule. The fact that LaBSE, initially the weakest, catches up to E5 after the probe underlines the value of this lightweight adjustment.

Third, explicit cues are easier to capture than implicit ones. Across all models, Request detection outperforms Frustration by 3–5 pp even after adaptation. This outcome reflects the linguistic asymmetry of the task: requests are usually marked by imperatives or modal verbs, whereas frustration is transmitted indirectly and diverse through narrative context. The residual confusion between priority Levels I and IV (both contain a request) and between Levels I and II (both contain frustration) stems from this same imbalance of cue salience.

These experiments highlight a clear research gap: no systematic framework exists for selecting the optimal prefix–instruction anchor mix for a given encoder and task. Current practice relies on model card anecdotes or ad hoc trials. The sharp accuracy swings we observe show that this choice is non-trivial yet understudied. Overall, this study shows that a single universal encoder, augmented only by handcrafted anchors and a tiny linear probe, can cover all three complaint facets and their derived priority grid with competitive accuracy.

This "one encoder—many facets" paradigm offers a low-latency, low-maintenance alternative to multiple task-specific fine-tuned models or API-based LLM calls. Updates to new domains or languages involve simply adjusting anchor phrases or retraining a few kilobytes of probe weights, rather than fine-tuning and validating multiple large-scale models. By eliminating the need for separate task-specific pipelines or repeated API calls to language models, this approach not only lowers operational costs but also ensures consistent, reproducible behavior across environments.

5.2. Rationale for Avoiding Full Supervised Fine-Tuning

To establish a baseline for our multi-faceted sentiment-classification experiments, we fine-tuned the multilingual BERT-base-cased (mBERT) model, which has 110 million parameters and supports 102 languages. Devlin et al. [68] reported that fine-tuning BERT-large on limited data is noticeably less stable and therefore requires multiple restarts, whereas BERT-base exhibits fewer fluctuations under the same conditions. Follow-up work [69] focuses on BERT-base for the same reason, concluding that its variance across random seeds is substantially lower in few-shot settings. However, employing BERT-base does not eliminate the risk of overfitting. Several studies [68–71] demonstrate that fine-tuning large pre-trained transformer models on small datasets remains unstable, with performance varying widely across random initializations.

Although the dataset size required for stable, full-parameter fine-tuning varies with the task, corpus characteristics, label granularity, class imbalance, and linguistic diversity, recent studies indicate that it must be considerably larger than our annotated dataset. For example, Manias et al. [72] successfully fine-tuned mBERT on the Multilingual Amazon Reviews Corpus, which provides 200,000 labeled reviews for training and 5000 each for development and testing for every one of six languages. Under these conditions the authors reported macro-F1 scores of about 0.72 to 0.74 on thirty-one-way product categorization and five-level sentiment analysis. A more recent study [73] on the lower-resourced Lithuanian language fine-tuned an entire BERT model on a corpus of 123,604 reviews and observed that updating all layers led to rapid overfitting. We therefore include the fully fine-tuned mBERT baseline not to endorse large-scale fine-tuning for low-resource cases but to serve as an experimental reference that illustrates its practical limitations.

The fine-tuning results for BERT are presented in Appendix A.2. Figure 15 compares the performance of the fine-tuned BERT model with lightweight adaptation of embedding models in aspect classification. It is clearly evident that BERT lags behind the encoder-based models and, moreover, exhibits strong overfitting. Figure 16 illustrates that the training loss decreases rapidly and approaches zero by the third epoch, while the test loss initially improves but then rises steadily across subsequent epochs.

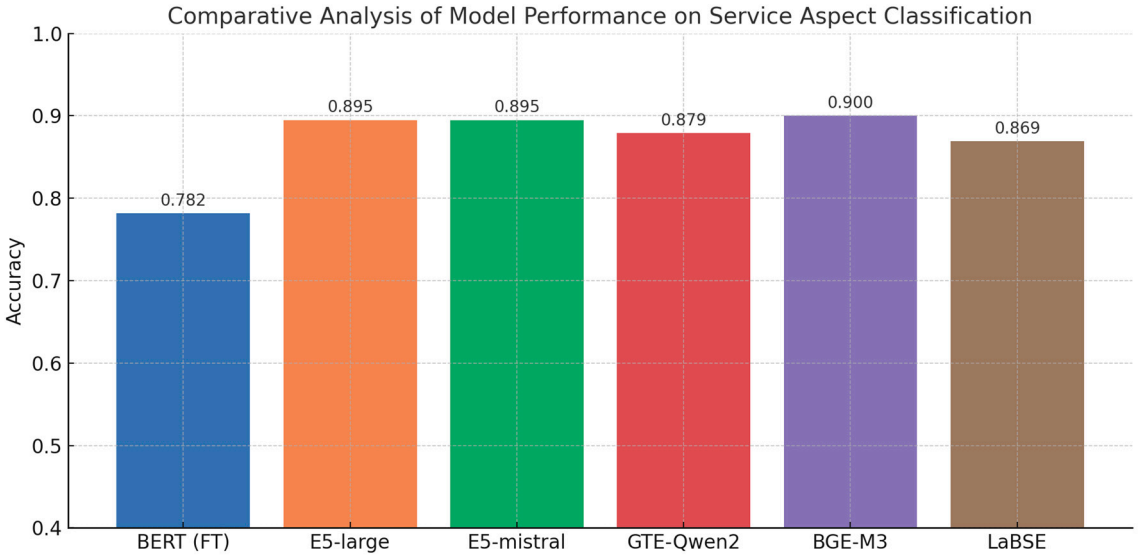


Figure 15. Model performance comparison on Aspect classification.

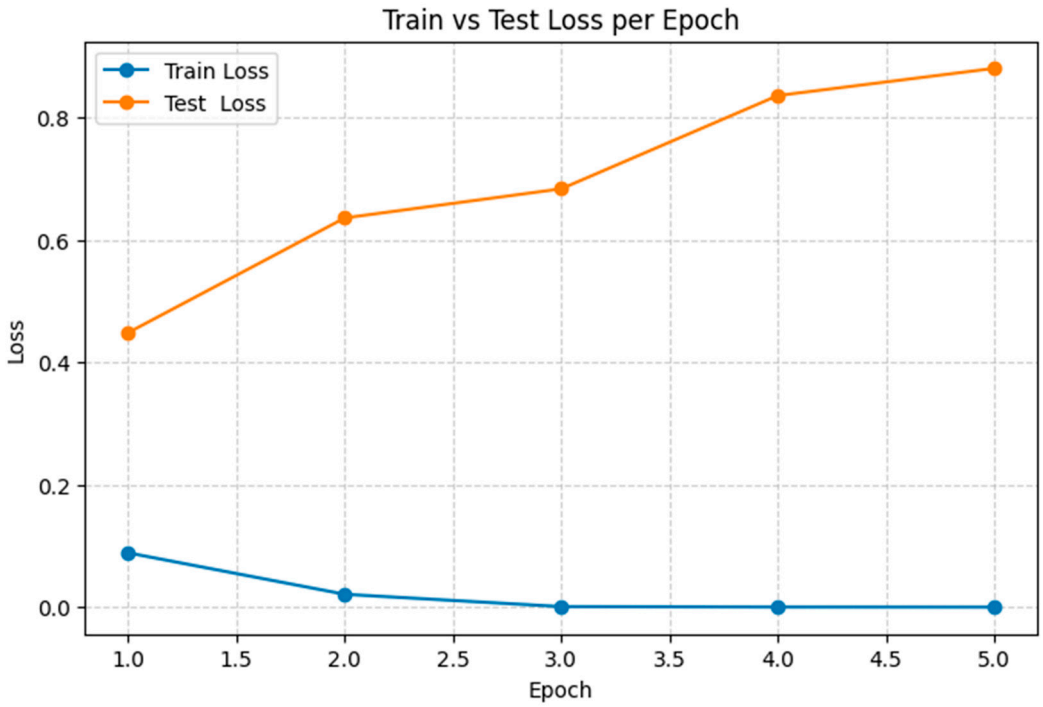


Figure 16. Train vs. test loss across epochs for the fine-tuned BERT model (Aspect classification).

Figures 17 and 18 clearly illustrate the overall lag of the fine-tuned BERT model: it trails every universal encoder on Frustration detection and surpasses only one encoder (GTE-Qwen2-1.5B-instruct) on Request detection. Even that isolated advantage is attributable not to better generalization but to the overfitting pattern already observed in its train-versus-test loss curves (see Figure 19).

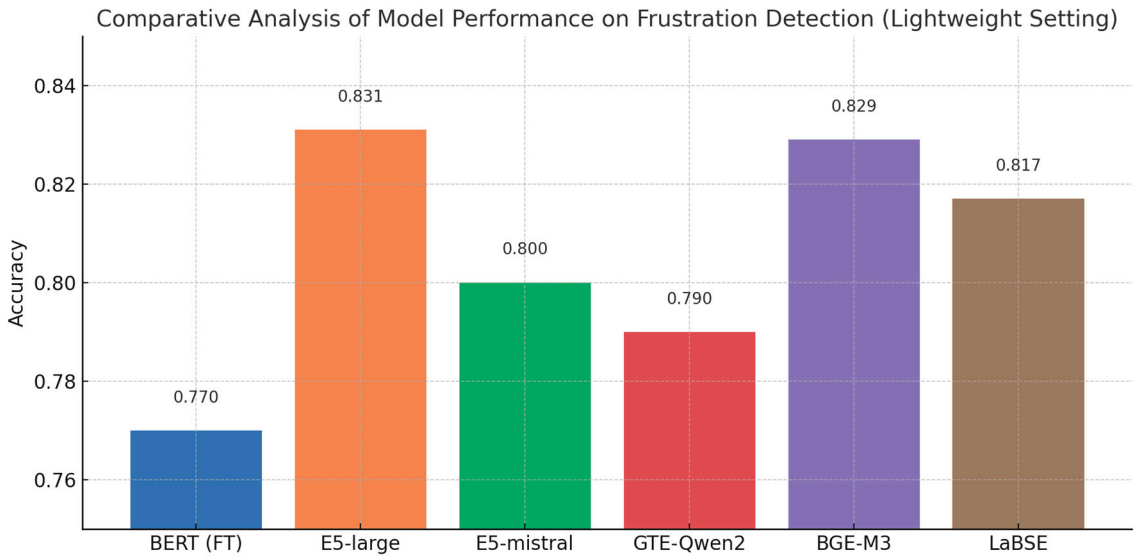


Figure 17. Model performance comparison on Frustration detection.

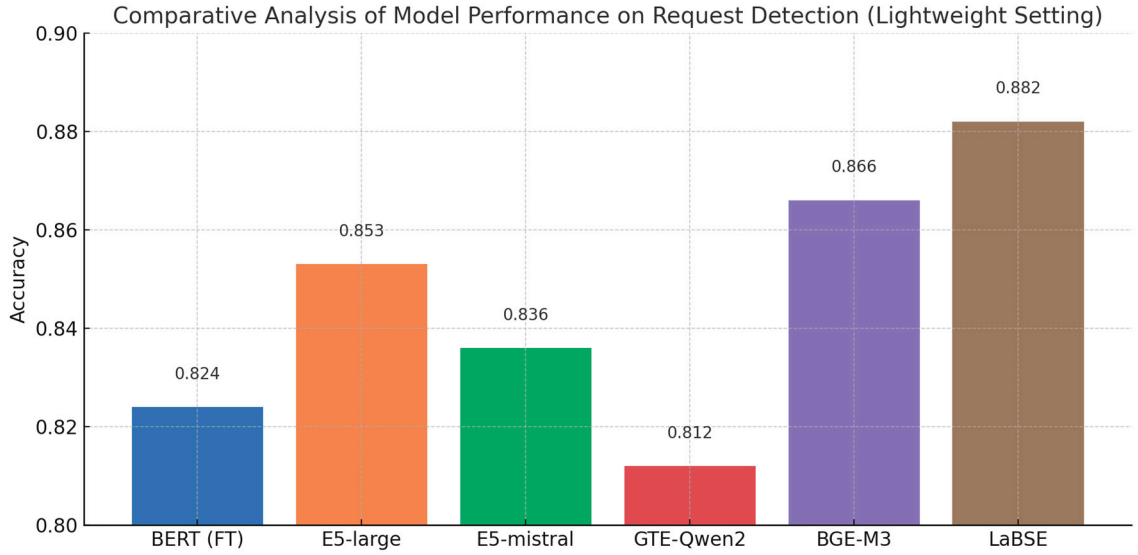


Figure 18. Model performance comparison on Request detection.

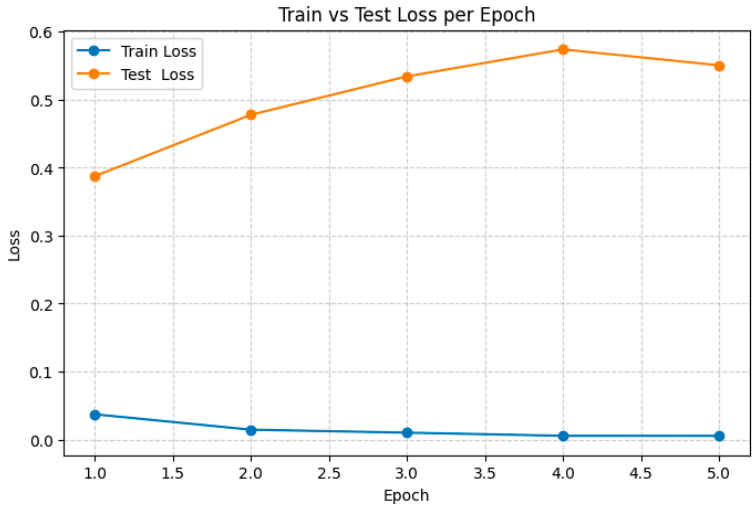


Figure 19. Train vs. test loss across epochs for the fine-tuned BERT model (Request detection).

5.3. Limitations

Our study has several limitations that may affect the reproducibility of the proposed approach:

- Narrow domain and language scope. The dataset is restricted to bus service complaints in Russian and Kazakh, so the results may not generalize to other domains or languages.
- Hand-crafted anchors. Anchor lists were created manually, which may introduce author bias; automatic anchor generation warrants further investigation.
- Global cosine threshold. A single mean similarity threshold is applied to the binary facets. This heuristic can drift under domain shift or when new encoders with different distance scales are used. In particular, if the dataset contains many poorly separable examples, the mean similarity rises, the global cut-off shifts upward, and the entire decision pipeline becomes unstable.

6. Conclusions

Our findings underscore the practical value of universal embedding pipelines for industry and public sector sentiment analysis/opinion mining tasks. Because a single frozen encoder, combined with lightweight logistic heads, already reaches 85–90% macro-F1 across multilingual facets, organizations can deploy real-time complaint triage without GPU-intensive fine-tuning or proprietary LLM APIs. The same anchor-based framework flexibly supports aspect-based sentiment analysis, emotion detection, and priority routing, all while running on-device and adapting to new categories with nothing more than an updated anchor list. In settings such as public transport feedback or social media monitoring, where data arrive continuously and budgets are limited, this low-cost, language-agnostic approach offers an immediately actionable alternative to full-scale model retraining, aligning well with this special issue's focus on practical, domain-specific sentiment applications.

Although our evaluation focuses on a single domain, the underlying architecture is inherently domain-invariant: no part of the encoder or classifier is tuned to transport-specific content, and adaptation to new domains requires only anchor reformulation, not retraining. Future work will include broader domain evaluations to further test the generalizability of the proposed approach.

We will also pursue efficient, fully automated anchor generation that avoids any renewed dependence on LLMs. Specifically, we will explore unsupervised keyword extraction algorithms (e.g., graph-based ranking and TF–IDF salience) in conjunction with semantic embedding clustering as a lightweight, interpretable alternative to LLM-driven prompting.

In future work, we will also investigate density-aware thresholding. Specifically, we intend to cluster complaint embeddings within an ε-radius of each anchor (e.g., via DBSCAN) and classify a complaint as positive only when it lies in a sufficiently dense neighborhood around that anchor. This adaptive rule tightens decision boundaries in noisy regions and relaxes them in sparse ones, while preserving the zero-shot nature of the pipeline and avoiding any encoder fine-tuning.

Author Contributions: Conceptualization, A.N.; Data curation, D.R. and A.M.; Methodology, A.N.; Software, A.N. and D.R.; Validation, A.N., D.R. and A.M.; Visualization, A.M.; Writing—original draft, A.N.; Writing—review and editing, A.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Committee of Science of the Ministry of Science and Higher Education of the Republic of Kazakhstan (Grant No.BR24992852 "Intelligent models and methods of Smart City digital ecosystem for sustainable development and the citizens' quality of life improvement").

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: Authors declare no conflicts of interest.

Appendix A

Appendix A.1. Cross-Validation Performance for Aspect Classification

Table A1. The performance of multilingual-E5-large-instruct ("Query:"–"Query:" configuration) with five-fold cross-validation.

Aspect	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
Bike	68	0.980	0.735	0.840	68	0.971	0.971	0.971
Condition	151	0.417	0.669	0.514	151	0.713	0.788	0.748
Information	65	0.516	0.754	0.613	65	0.747	0.862	0.800
Lost	1	0.143	1.000	0.250	1	0.000	0.000	0.000
Operations	1305	0.807	0.918	0.859	1305	0.950	0.923	0.937
Payment	230	0.916	0.761	0.831	230	0.926	0.874	0.899
Personnel	560	0.880	0.473	0.616	560	0.831	0.861	0.846
Roadside	20	0.536	0.750	0.625	20	0.680	0.850	0.756
Total	2400	Accuracy: 0.772			2400	Accuracy: 0.894		

Table A2. The performance of E5-mistral-7b-instruct ("Query:"–"Query:" configuration) with five-fold cross-validation.

Aspect	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
Bike	68	0.938	0.882	0.909	68	0.985	0.985	0.985
Condition	151	0.361	0.656	0.466	151	0.739	0.788	0.763
Information	65	0.481	0.800	0.601	65	0.833	0.846	0.840
Lost	1	0.000	0.000	0.000	1	0.000	0.000	0.000
Operations	1305	0.831	0.857	0.843	1305	0.954	0.921	0.937
Payment	230	0.934	0.796	0.859	230	0.942	0.922	0.932
Personnel	560	0.832	0.568	0.675	560	0.829	0.873	0.850
Roadside	20	0.556	0.750	0.638	20	0.600	0.900	0.720
Total	2400	Accuracy: 0.769			2400	Accuracy: 0.901		

Table A3. The performance of gte-Qwen2-1.5B-instruct ("Query:"–"Query:" configuration) with five-fold cross-validation.

Aspect	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
Bike	68	0.882	0.882	0.882	68	0.971	0.971	0.971
Condition	151	0.269	0.450	0.337	151	0.676	0.775	0.722

Table A3. *Cont.*

Aspect	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
Information	65	0.333	0.908	0.488	65	0.821	0.846	0.833
Lost	1	0.043	1.000	0.083	1	0.000	0.000	0.000
Operations	1305	0.766	0.904	0.829	1305	0.953	0.906	0.929
Payment	230	0.967	0.635	0.766	230	0.938	0.926	0.932
Personnel	560	0.863	0.248	0.386	560	0.789	0.848	0.818
Roadside	20	0.577	0.750	0.652	20	0.696	0.800	0.744
Total	2400	Accuracy: 0.695			2400	Accuracy: 0.885		

Table A4. The performance of BGE-M3 ("Query:" before anchor configuration) with five-fold cross-validation.

Aspect	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
Bike	68	0.937	0.868	0.901	68	0.985	0.971	0.963
Condition	151	0.688	0.291	0.409	151	0.702	0.781	0.754
Information	65	0.355	0.769	0.485	65	0.848	0.862	0.857
Lost	1	0.077	1.000	0.143	1	0.000	0.000	0.000
Operations	1305	0.669	0.969	0.792	1305	0.959	0.928	0.944
Payment	230	0.970	0.700	0.813	230	0.935	0.935	0.930
Personnel	560	0.942	0.087	0.160	560	0.841	0.879	0.863
Roadside	20	0.923	0.600	0.727	20	0.762	0.800	0.800
Total	2400	Accuracy: 0.683			2400	Accuracy: 0.906		

Table A5. The performance of LaBSE ("Query:"–"Query:" configuration) with five-fold cross-validation.

Aspect	Zero-Shot Classification				Lightweight Classification			
	Count	**P**	**R**	**F1**	**Count**	**P**	**R**	**F1**
Bike	68	0.312	0.794	0.448	68	0.928	0.941	0.934
Condition	151	0.223	0.656	0.333	151	0.631	0.781	0.698
Information	65	0.479	0.523	0.500	65	0.667	0.831	0.740
Lost	1	0.000	0.000	0.000	1	0.000	0.000	0.000
Operations	1305	0.774	0.788	0.781	1305	0.957	0.910	0.933
Payment	230	0.867	0.396	0.543	230	0.905	0.874	0.889
Personnel	560	0.778	0.087	0.157	560	0.819	0.839	0.829
Roadside	20	0.084	0.750	0.151	20	0.538	0.700	0.609
Total	2400	Accuracy: 0.571			2400	Accuracy: 0.879		

Appendix A.2. Fine-Tuning BERT

Table A6. Performance of the finetuned BERTBase (uncased) model on "Aspect" classification.

Aspect	Count	P	R	F1
Bike	27	0.923	0.444	0.600
Condition	61	0.000	0.000	0.000
Information	26	0.421	0.308	0.356
Lost	0	0.000	0.000	0.000
Operations	522	0.852	0.914	0.882
Payment	92	0.833	0.761	0.795
Personnel	224	0.648	0.821	0.724
Roadside	8	0.000	0.000	0.000
Total	960		Accuracy: 0.782	

Table A7. Performance of the fine-tuned BERT-Base (uncased) model on Frustration and Request detection.

Facet	Count	P	R	F1
Frustration				
0	571	0.806	0.807	0.807
1	389	0.717	0.715	0.716
Total	960		Accuracy: 0.770	
Request				
0	486	0.868	0.770	0.816
1	474	0.788	0.880	0.832
Total	960		Accuracy: 0.824	

Appendix A.3. Instruction–Prefix Sensitivity

Table A8. The best "Instruction-Prefix" configurations for the universal embedding models.

ID	Instruction Line	Anchor Prefix	Complaint Prefix	Multilingual-E5-Large-Instruct	E5-Mistral-7B-Instruct	GTE-Qwen2-1.5B-Instruct	BGE-M3	LaBSE
1	No	-	-					
2	No	query:	query:	✓	✓	✓		✓
3	No	query:	passage:	-	-	-	-	-
4	No	query:	-					
5	No	-	query:				✓	
6	No	-	passage:					
7	No	passage:	passage:					
8	No	passage:	-					
9	Yes	-	-					
10	Yes	query:	query:					
11	Yes	query:	passage:					
12	Yes	query:	-					

Table A8. *Cont.*

ID	Instruction Line	Anchor Prefix	Complaint Prefix	Multilingual-E5-Large-Instruct	E5-Mistral-7B-Instruct	GTE-Qwen2-1.5B-Instruct	BGE-M3	LaBSE
13	Yes	-	query:					
14	Yes	-	passage:					
15	Yes	passage:	passage:					
16	Yes	passage:	-					

References

1. Sigurdsson, V.; Larsen, N.M.; Gudmundsdottir, H.K.; Alemu, M.H.; Menon, R.V.; Fagerstrøm, A. Social media: Where customers air their troubles—How to respond to them? *J. Innov. Knowl.* **2021**, *4*, 257–267. [CrossRef]
2. Moreno, A.; Iglesias, C.A. Understanding customers' transport services with topic clustering and sentiment analysis. *Appl. Sci.* **2021**, *21*, 10169. [CrossRef]
3. Osorio-Arjona, J.; Horak, J.; Svoboda, R.; García-Ruíz, Y. Social media semantic perceptions on Madrid Metro system: Using Twitter data to link complaints to space. *Sustain. Cities Soc.* **2021**, *64*, 102530. [CrossRef]
4. Gong, S.H.; Teng, J.; Duan, C.Y.; Liu, S.J. Framework for evaluating online public opinions on urban rail transit services through social media data classification and mining. *Res. Transp. Bus. Manag.* **2024**, *56*, 101197. [CrossRef]
5. Das, R.D. Understanding users' satisfaction towards public transit system in India: A case-study of Mumbai. *ISPRS Int. J. Geo-Inf.* **2021**, *3*, 155. [CrossRef]
6. Sahil, P.S.; Jamatia, A. Team A at SemEval-2025 Task 11: Breaking Language Barriers in Emotion Detection with Multilingual Models. *arXiv* **2025**, arXiv:2502.19856.
7. Zhou, Y.; Muresanu, A.I.; Han, Z.; Paster, K.; Pitis, S.; Chan, H.; Ba, J. Large language models are human-level prompt engineers. In Proceedings of the Eleventh International Conference on Learning Representations, Kigali, Rwanda, 25 April 2022.
8. Lan, Y.; Wu, Y.; Xu, W.; Feng, W.; Zhang, Y. Chinese fine-grained financial sentiment analysis with large language models. *Neural Comput. Appl.* **2024**, 1–10. [CrossRef]
9. Shah, F.A.; Sabir, A.; Sharma, R. A Fine-grained Sentiment Analysis of App Reviews using Large Language Models: An Evaluation Study. *arXiv* **2024**, arXiv:2409.07162. [CrossRef]
10. Teng, J.; He, H.; Hu, G. A fine-grained sentiment recognition method for online Government-Public interaction texts based on large language models. In Proceedings of the International Conference on Artificial Intelligence and Machine Learning Research (CAIMLR 2024), Novena, Singapore, 28–29 September 2024; SPIE: Bellingham, WA, USA, 2025; Volume 13635, pp. 11–16.
11. Wang, L.; Yang, N.; Huang, X.; Yang, L.; Majumder, R.; Wei, F. Multilingual e5 text embeddings: A technical report. *arXiv* **2024**, arXiv:2402.05672. [CrossRef]
12. Chen, J.; Xiao, S.; Zhang, P.; Luo, K.; Lian, D.; Liu, Z. BGE M3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv* **2024**, arXiv:2402.03216.
13. Li, Z.; Zhang, X.; Zhang, Y.; Long, D.; Xie, P.; Zhang, M. Towards general text embeddings with multi-stage contrastive learning. *arXiv* **2023**, arXiv:2308.03281. [CrossRef]
14. Solatorio, A.V. Gistembed: Guided in-sample selection of training negatives for text embedding fine-tuning. *arXiv* **2024**, arXiv:2402.16829. [CrossRef]
15. Cao, H. Recent advances in text embedding: A Comprehensive Review of Top-Performing Methods on the MTEB Benchmark. *arXiv* **2024**, arXiv:2406.01607.
16. Gan, D.; Li, J. Small, Open-Source Text-Embedding Models as Substitutes to OpenAI Models for Gene Analysis. *bioRxiv* **2025**, 638462. [CrossRef]
17. Muennighoff, N.; Tazi, N.; Magne, L.; Reimers, N. MTEB: Massive text embedding benchmark. *arXiv* **2022**, arXiv:2210.07316.
18. *BGE-M3 Model Card and Documentation*, Version 1.0; Hugging Face; Beijing Academy of Artificial Intelligence: Beijing, China, 2023. Available online: https://huggingface.co/BAAI/bge-m3 (accessed on 26 June 2025).
19. Goffman, E. *Forms of Talk*; University of Pennsylvania Press: Philadelphia, PA, USA, 1981.
20. Su, H.; Shi, W.; Kasai, J.; Wang, Y.; Hu, Y.; Ostendorf, M.; Yih, W.-t.; Smith, N.A.; Zettlemoyer, L.; Yu, T. One embedder, any task: Instruction-finetuned text embeddings. *arXiv* **2022**, arXiv:2212.09741.
21. Wu, C.; Wu, F.; Liu, J.; Yuan, Z.; Wu, S.; Huang, Y. Thu_ngn at semeval-2018 task 1: Fine-grained tweet sentiment intensity analysis with attention Cnn-LSTM. In Proceedings of the 12th International Workshop on Semantic Evaluation, New Orleans, LA, USA, 5–6 June 2018; pp. 186–192.
22. What is sentiment analysis? In Proceedings of the IBM Think, Boston, MA, USA, 24 August 2023; IBM: Armonk, NY, USA, 2023. Available online: https://www.ibm.com/think/topics/sentiment-analysis (accessed on 22 July 2025).

23. Amazon Web Services. What Is Sentiment Analysis? Available online: https://aws.amazon.com/what-is/sentiment-analysis/ (accessed on 22 June 2025).
24. Kausar, S.; Huahu, X.U.; Ahmad, W.; Shabir, M.Y. A sentiment polarity categorization technique for online product reviews. *IEEE Access* **2019**, *8*, 3594–3605. [CrossRef]
25. Tirpude, S.; Thakre, Y.; Sudan, S.; Agrawal, S.; Ganorkar, A. Mining Comments and Sentiments in YouTube Live Chat Data. In Proceedings of the 2023 4th International Conference on Intelligent Technologies (CONIT), Hubballi, India, 23–25 June 2023; IEEE: Piscataway, NJ, USA, June 2024; pp. 1–6.
26. Dong, Y. Fine-grained sentiment analysis for social media: From multi-model collaboration to cross-language multimodal analysis. *Adv. Eng. Innov.* **2025**, *5*, 152–156. [CrossRef]
27. Zhao, Y.; Fang, J.; Jin, S. Fine-Grained Sentiment Analysis Based on SSFF-GCN Model. *Systems* **2025**, *2*, 111. [CrossRef]
28. Kaminska, O.; Cornelis, C.; Hoste, V. Fuzzy rough nearest neighbour methods for aspect-based sentiment analysis. *Electronics* **2023**, *5*, 1088. [CrossRef]
29. Birjali, M.; Kasri, M.; Beni-Hssane, A. A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowl.-Based Syst.* **2021**, *226*, 107134. [CrossRef]
30. Zhao, L.; Lee, S.W. Integrating Ontology-Based Approaches with Deep Learning Models for Fine-Grained Sentiment Analysis. *Comput. Mater. Contin.* **2024**, *81*, 1855–1877. [CrossRef]
31. Hu, H.; Wei, Y.; Zhou, Y. Product-harm crisis intelligent warning system design based on fine-grained sentiment analysis of automobile complaints. *Complex Intell. Syst.* **2023**, *3*, 2313–2320. [CrossRef]
32. Akila, R.; Revathi, S. Fine Grained Analysis of Intention for Social Media Reviews Using Distance Measure and Deep Learning Technique. *J. Internet Serv. Inf. Secur.* **2023**, *2*, 48–64. [CrossRef]
33. Torres, E.C.M.; de Picado-Santos, L.G. Sentiment Analysis and Topic Modeling in Transportation: A Literature Review. *Appl. Sci.* **2025**, *12*, 6576. [CrossRef]
34. Pullanikkat, R.; Poddar, S.; Das, A.; Jaiswal, T.; Singh, V.K.; Basu, M.; Ghosh, S. Utilizing the Twitter social media to identify transportation-related grievances in Indian cities. *Soc. Netw. Anal. Min.* **2024**, *1*, 118. [CrossRef]
35. Leong, M.; Abdelhalim, A.; Ha, J.; Patterson, D.; Pincus, G.L.; Harris, A.B.; Eichler, M.; Zhao, J. Metroberta: Leveraging traditional customer relationship management data to develop a transit-topic-aware language model. *Transp. Res. Rec.* **2024**, *9*, 215–229. [CrossRef]
36. Rønningstad, E.; Storset, L.C.; Mæhlum, P.; Øvrelid, L.; Velldal, E. Mixed Feelings: Cross-Domain Sentiment Classification of Patient Feedback. In Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025), Tallinn, Estonia, 3–5 March 2025; pp. 537–543.
37. Pullanikkat, R.; Basu, M.; Ghosh, S. Hear the Commute: A Generative AI-Based Framework to Summarize Transport Grievances from Social Media. *Int. J. Intell. Transp. Syst. Res.* **2025**, 1–17. [CrossRef]
38. Li, J.; Yang, Y.; Chen, R.; Zheng, D.; Pang, P.C.I.; Lam, C.K.; Wong, D.; Wang, Y. Identifying healthcare needs with patient experience reviews using ChatGPT. *PLoS ONE* **2025**, *3*, e0313442. [CrossRef]
39. Li, J.; Yang, Y.; Mao, C.; Pang, P.C.I.; Zhu, Q.; Xu, D.; Wang, Y. Revealing patient dissatisfaction with health care resource allocation in multiple dimensions using large language models and the international classification of diseases 11th revision: Aspect-based sentiment analysis. *J. Med. Internet Res.* **2025**, *27*, e66344. [CrossRef]
40. Ruan, K.; Wang, X.; Di, X. From Twitter to Reasoner: Understand mobility travel modes and sentiment using large language models. In Proceedings of the 2024 IEEE 27th International Conference on Intelligent Transportation Systems (ITSC), Edmonton, AB, Canada, 24–27 September 2024; IEEE: New York, NY, USA, 2024; pp. 454–459.
41. Esperança, M.; Freitas, D.; Paixão, P.V.; Marcos, T.A.; Martins, R.A.; Ferreira, J.C. Proactive Complaint Management in Public Sector Informatics Using AI: A Semantic Pattern Recognition Framework. *Appl. Sci.* **2025**, *12*, 6673. [CrossRef]
42. Mehta, P.; Bansal, S. A Unified Platform for Resolving Citizens' Queries on Beneficiary Services by Using AI-Powered Chatbots. *Int. J. Environ. Sci.* **2025**, *11*, 362–376. [CrossRef]
43. Xiong, Y.; Chen, G.; Cao, J. Research on Public Service Request Text Classification Based on BERT-BiLSTM-CNN Feature Fusion. *Appl. Sci.* **2024**, *14*, 6282. [CrossRef]
44. Nai, R.; Meo, R.; Morina, G.; Pasteris, P. Public tenders, complaints, machine learning and recommender systems: A case study in public administration. *Comput. Law Secur. Rev.* **2023**, *51*, 105887. [CrossRef]
45. Agustina, N.; Naseer, M.; Gusdevi, H.; Rismayadi, D.A. Development of a Public Complaint Classification Model to Support E-Government Using IndoBERT. In Proceedings of the 2024 6th International Conference on Cybernetics and Intelligent System (ICORIS), Central Java, Indonesia, 29–30 November, 2024; IEEE: New York, NY, USA, 2024; pp. 1–6.
46. Jin, M.; Aletras, N. Modeling the Severity of Complaints in Social Media. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Mexico City, Mexico, 6–11 June 2021; pp. 2264–2274.

47. Liu, J.; Long, R.; Chen, H.; Wu, M.; Ma, W.; Li, Q. Topic-sentiment analysis of citizen environmental complaints in China: Using a Stacking-BERT model. *J. Environ. Manag.* **2024**, *371*, 123112. [CrossRef]
48. Singh, A.; Saha, S. Are you really complaining? A multi-task framework for complaint identification, emotion, and sentiment classification. In Proceedings of the International Conference on Document Analysis and Recognition, Lausanne, Switzerland, 5–10 September 2021; Springer International Publishing: Cham, Switzerland, 2021; pp. 715–731.
49. Singh, A.; Bhatia, R.; Saha, S. Complaint and severity identification from online financial content. *IEEE Trans. Comput. Soc. Syst.* **2023**, *1*, 660–670. [CrossRef]
50. Joshi, A.; Kumar, A.; Patel, N. Leveraging Natural Language Processing for Real-Time Financial Complaint Analysis on Social Media: A Multitasking Approach. *SSRN* **2024**, SSRN 5059673. [CrossRef]
51. Caralt, M.H.; Sekulić, I.; Carević, F.; Khau, N.; Popa, D.N.; Guedes, B.; GuimarãesMathis, V.; Yang, Z.; Manso, A.; Reddy, M.; et al. "Stupid robot, I want to speak to a human!" User Frustration Detection in Task-Oriented Dialog Systems. *arXiv* **2024**, arXiv:2411.17437. [CrossRef]
52. Berkowitz, L. Frustration-aggression hypothesis: Examination and reformulation. *Psychol. Bull.* **1989**, *106*, 59. [CrossRef] [PubMed]
53. Searle, J.R. *Expression and Meaning: Studies in the Theory of Speech Acts*; Cambridge University Press: Cambridge, UK, 1979.
54. Wang, L.; Li, L.; Dai, D.; Chen, D.; Zhou, H.; Meng, F.; Zhou, J.; Sun, X. Label Words are Anchors: An Information Flow Perspective for Understanding In-Context Learning. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Singapore, 6–10 December 2023.
55. Snell, J.; Swersky, K.; Zemel, R. Prototypical networks for few-shot learning. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 4077–4087.
56. Liu, H.; Zhao, S.; Zhang, X.; Zhang, F.; Wang, W.; Ma, F.; Chen, H.; Yu, H.; Zhang, X. Liberating seen classes: Boosting few-shot and zero-shot text classification via anchor generation and classification reframing. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024; Volume 38, pp. 18644–18652.
57. Paletto, L.; Basile, V.; Esposito, R. Label Augmentation for Zero-Shot Hierarchical Text Classification. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics, Bangkok, Thailand, 11–16 August 2024; Volume 1, pp. 7697–7706.
58. Hugging Face. Intfloat/Multilingual-e5-large-instruct. Available online: https://huggingface.co/intfloat/multilingual-e5-large-instruct (accessed on 1 July 2025).
59. Asai, A.; Schick, T.; Lewis, P.; Chen, X.; Izacard, G.; Riedel, S.; Yih, W.T. Task-aware Retrieval with Instructions. In *Findings of the Association for Computational Linguistics: ACL*; ACL: Stroudsburg, PA, USA, 2023; pp. 3650–3675.
60. Zhang, W.; Xiong, C.; Stratos, K.; Overwijk, A. Improving multitask retrieval by promoting task specialization. *Trans. Assoc. Comput. Linguist.* **2023**, *11*, 1201–1212. [CrossRef]
61. He, H.; Garcia, E.A. Learning from imbalanced data. *IEEE Trans. Knowl. Data Eng.* **2009**, *9*, 1263–1284. [CrossRef]
62. Tang, Y.; Yang, Y. Pooling and attention: What are effective designs for llm-based embedding models? *arXiv* **2024**, arXiv:2409.02727. [CrossRef]
63. Wang, L.; Yang, N.; Huang, X.; Jiao, B.; Yang, L.; Jiang, D.; Majumder, R.; Wei, F. Text embeddings by weakly-supervised contrastive pre-training. *arXiv* **2022**, arXiv:2212.03533.
64. Huang, L.; Liang, S.; Ye, F.; Gao, N. A fast attention network for joint intent detection and slot filling on edge devices. *IEEE Trans. Artif. Intell.* **2023**, *2*, 530–540. [CrossRef]
65. Senge, R.; Del Coz, J.J.; Hüllermeier, E. On the problem of error propagation in classifier chains for multi-label classification. In *Data Analysis, Machine Learning and Knowledge Discovery*; Springer International Publishing: Cham, Switzerland, 2013; pp. 163–170.
66. Bell, S.J.; Meglioli, M.C.; Richards, M.; Sánchez, E.; Ropers, C.; Wang, S.; Williams, A.; Sagun, L.; Costa-jussà, M.R. On the role of speech data in reducing toxicity detection bias. *arXiv* **2024**, arXiv:2411.08135. [CrossRef]
67. Rakhimzhanov, D.; Belginova, S.; Yedilkhan, D. Automated Classification of Public Transport Complaints via Text Mining Using LLMs and Embeddings. *Information* **2025**, *8*, 644. [CrossRef]
68. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, Minnesota, 2–7 June 2019; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; Volume 1 (Long and Short Papers), pp. 4171–4186.
69. Mosbach, M.; Andriushchenko, M.; Klakow, D. On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines. *arXiv* **2020**, arXiv:2006.04884.
70. Du, Y.; Nguyen, D. Measuring the Instability of Fine-Tuning. In Proceedings of the 61st Annual Meeting Of The Association For Computational Linguistics, Toronto, Canada, 9–14 July 2023.
71. Hua, H.; Li, X.; Dou, D.; Xu, C.; Luo, J. Noise Stability Regularization for Improving BERT Fine-tuning. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Mexico City, Mexico, 6–11 June 2021; pp. 3229–3241.

72. Manias, G.; Mavrogiorgou, A.; Kiourtis, A.; Symvoulidis, C.; Kyriazis, D. Multilingual text categorization and sentiment analysis: A comparative analysis of the utilization of multilingual approaches for classifying twitter data. *Neural Comput. Appl.* **2023**, *29*, 21415–21431. [CrossRef] [PubMed]
73. Vileikytė, B.; Lukoševičius, M.; Stankevičius, L. Sentiment analysis of Lithuanian online reviews using large language models. In Proceedings of the CEUR Workshop Proceedings: IVUS 2024: Information Society and University Studies 2024, Proceedings of the 29th International Conference on Information Society and University Studies (IVUS 2023), Kaunas, Lithuania, 17 May 2024; CEUR-WS: Aachen, Germany, 2024; Volume 3885, pp. 249–258.