Beyond Label Semantics: Concept-Sensitive Classification of National R&D Projects

Anonymous ACL submission

Abstract

We propose a concept-sensitive classification framework for large-scale national R&D projects. Unlike prior approaches that rely on implicit label semantics or surface-level similarity, our framework uses explicit conceptual definitions as operational semantic constraints for LLM-based classification of conceptually defined categories. We apply this framework to a two-stage pipeline: service R&D filtering and AI transformation (AX) stage classification, evaluated on about 58,000 national R&D projects from the Korean National Science and Technology Information Service (NTIS). Through a 2×2 ablation study varying inference paradigm (generative LLM vs. embedding-based similarity) and prompt content (concept definitions vs. label names only), we demonstrate that explicit concept definitions yield consistent performance gains across both classification stages, and that LLM-based reasoning substantially outperforms embedding-based semantic similarity. Our results suggest that concept-operationalization improves semantically ambiguous category discrimination and uncertainty calibration and that it is an effective and scalable approach to policy-oriented R&D classification tasks.

1 Introduction

Classification of national R&D projects is a critical task in science and technology (S&T) policy, enabling production of R&D statistics that support resource allocation, portfolio analysis, and strategic planning. Large-scale R&D databases, such as the Korean National Science and Technology Information Service (NTIS), accumulate tens of thousands of project records every year that contain unstructured text data. To extract in-depth information from such data, we need conceptual frameworks that define categories not by topic or domain, but by conceptual properties, such as whether a project constitutes a service R&D effort or the degree of AI autonomy it embodies.

Such classification tasks pose unique challenges. First, category boundaries are defined by latent conceptual properties rather than surface-level lexical similarity. For example, categories such as AI tool, AI agent, and agentic AI share substantial terminological overlap, yet differ critically in the level of autonomy, decision-making capability, and execution structure. Second, the scale of national R&D databases makes manual annotation infeasible, requiring automated approaches that maintain semantic consistency across diverse project descriptions.

Existing automated classification approaches fall into three broad paradigms: embedding-based similarity methods, supervised encoder fine-tuning, and prompting-based LLM approaches (Li et al., 2022; Cunha et al., 2025). Particularly, LLM-based prompting has recently enabled zero-shot classification through generative semantic reasoning. However, existing LLM-based approaches rely primarily on the model's prior knowledge to infer category meanings, leaving the classification process largely implicit and difficult to control.

We address these limitations by proposing a concept-guided classification framework in which LLM interpretation is explicitly steered through conceptual definitions embedded in the prompt. Rather than relying on implicit label semantics, our framework encodes operational criteria for each category as a modular concept layer that is maintained independently of the underlying inference model. This design enables interpretable, controllable, and reproducible classification of conceptually defined R&D categories.

We apply the framework to two sequential classification tasks: (1) service R&D filtering, which distinguishes service-oriented projects from non-service R&D, and (2) AX (AI transformation) stage classification in which the service R&D projects are categorized according to the level of AI autonomy and decision-making capability manifested in

the project.

Our contributions are as follows:

- We propose a concept-sensitive classification framework in which LLM inference is guided by explicit conceptual definitions, enabling controllable classification of conceptually defined R&D categories.
- We introduce a modular concept-layer architecture that separates conceptual definitions from inference mechanisms, supporting transparent, extensible, and interpretable classification pipelines.
- Through a 2×2 ablation study, we show that explicit concept definitions consistently outperform label-only prompting, and that LLM-based inference substantially outperforms embedding-based similarity for concept-sensitive classification.
- We conduct a large-scale empirical evaluation on approximately 58,000 national R&D project records, demonstrating the scalability and practical applicability of the proposed framework.

2 Related Work

2.1 Text Classification Paradigms

Embedding-based methods classify texts through semantic proximity in latent representation spaces, while supervised encoder fine-tuning learns task-specific discriminative boundaries from labeled examples (Devlin et al., 2019). More recently, prompting-based LLM methods have enabled classification through generative semantic reasoning without task-specific parameter updates (Yin et al., 2019). Recent benchmark studies suggest that the effectiveness of these paradigms depends on task characteristics and application constraints (Cunha et al., 2025; Simões et al., 2026). In particular, concept-sensitive classification tasks require not only predictive performance but also controllable semantic interpretation of conceptually defined categories.

2.2 Scientific and R&D Text Classification

In scientific and R&D domains, classification tasks present additional challenges due to domain-specific language and specialized terminology. Beltagy et al. (2019) highlight the importance of domain adaptation for handling technical texts such as research abstracts and project descriptions, while Simões et al. (2026) show that task-specific encoder fine-tuning remains effective for localized scientific classification tasks. However, similarity-based methods primarily rely on latent semantic proximity, limiting their ability to support concept-sensitive tasks. Prior studies have largely treated category structures as fixed labels rather than semantically operationalized concepts, a limitation that becomes particularly significant when categories share overlapping terminology while differing in functional characteristics such as autonomy or execution structure.

2.3 LLM-based Classification

Recent work has explored LLM-based prompt engineering for text classification, demonstrating its applicability across specialized domain tasks (Clavié et al., 2023; Gunes and Florczak, 2025). Jang et al. (2025) propose a framework for evaluating public value in R&D outputs using GPT-based classification of patent texts into six categories. However, this approach relies primarily on the model's latent prior knowledge of the category labels, leaving category interpretation implicit and model-dependent. Our work addresses this gap through a concept-guided framework applied to large-scale R&D texts classification, where explicit concept definitions are maintained as an independent module.

3 Framework

3.1 Task Definition

We formulate concept-sensitive classification as a *constrained semantic mapping* problem. Given a project description x, the goal is to assign a label $y \in \mathcal{Y}$ via:

$$y^* = \arg\max_{y \in \mathcal{Y}} S(x, c_y; \theta) \tag{1}$$

where $c_y \in C$ is the concept definition associated with label y, and $S(\cdot)$ is a semantic compatibility function parameterized by inference mechanism θ. This formulation contrasts with standard zero-shot classification, where $S(x, y; \theta)$ operates directly on label names rather than structured concept definitions.

3.2 Concept Layer

A key feature of the proposed framework is the explicit encoding of conceptual definitions as a modular component of the classification pipeline. Each

category is associated with a structured set of operational criteria capturing the semantic properties that distinguish it from adjacent categories.

For Stage 1, the concept layer defines three categories: SERVICE, NON_SERVICE, and UNCERTAIN. Service R&D includes intangible service development, service process innovation, service-oriented systems, and related service-oriented social science research. Non-service R&D includes physical product development such as materials, parts, equipment, manufacturing processes, and pharmaceutical or biotechnology research.

For Stage 2, the concept layer defines five AX categories: NON_AX, LEVEL_1 (AI-assisted systems), LEVEL_2 (AI agents), LEVEL_3 (agentic AI), and UNCERTAIN. These categories represent increasing levels of AI autonomy, ranging from AI systems that support human decision-making (Level 1), to systems that execute predefined tasks autonomously (Level 2), to systems capable of autonomous planning and end-to-end goal-directed execution (Level 3).

3.3 Pipeline Engineering

Beyond the classification framework itself, we implement a production-grade pipeline for large-scale LLM inference that addresses practical challenges in applying LLMs to real-world R&D databases at scale.

Asynchronous batch orchestration. Rather than synchronous API calls, we leverage the OpenAI Batch API for asynchronous inference, reducing cost by 50% and enabling large-scale processing without blocking execution. The pipeline automatically partitions the full dataset into fixed-size batches, submits them asynchronously, and polls for completion before proceeding to the next stage.

Resumable execution. All batch submissions and their completion status are logged to a persistent record, enabling the pipeline to resume from any intermediate stage in the event of interruption. Each batch part is independently recoverable, supporting robust execution over multi-hour inference jobs.

Staged semantic classification. The two-stage pipeline (SVS → AX) enforces sequential dependency: Stage 2 is executed only after Stage 1 results are fully committed to the database, ensuring that AX classification operates on a well-defined SERVICE subset.

Structured output enforcement and parser recovery. All LLM outputs are required to conform to a fixed JSON schema. A robust parser handles common failure modes including markdown-wrapped JSON and partial outputs, with failed cases logged separately for downstream inspection.

Database persistence and idempotency. Results are stored in a relational database using upsert semantics, ensuring that re-execution of any batch part produces consistent results without duplication. The schema separates Stage 1 and Stage 2 results within a single table, supporting efficient cross-stage queries and multi-condition ablation management across five separate databases.

Modular prompt architecture. Concept definitions are maintained as independent plain-text files, fully decoupled from inference logic. This separation enables prompt updates, ablation variants, and cross-domain adaptation without modifying pipeline code, and ensures that all four ablation conditions share identical infrastructure while varying only in model and prompt content. Figure 1 illustrates the overall framework.

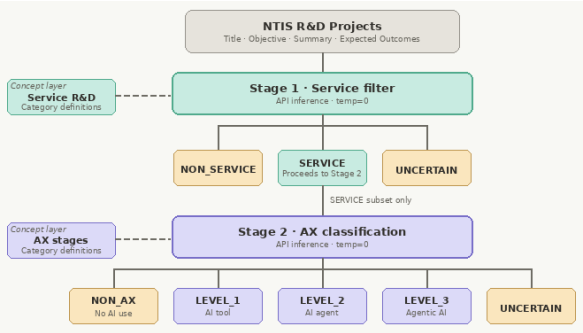

Figure 1: Two-stage classification pipeline for R&D project categorization. Stage 1 filters projects into SERVICE, NON_SERVICE, and UNCERTAIN categories. Stage 2 classifies SERVICE projects by AX level. The concept layer provides category definitions independently for each stage.

3.4 Baselines

We design a 2×2 ablation varying two factors: inference paradigm (generative LLM vs. embedding-based similarity) and prompt content (concept definitions vs. label names only).

LLM-label Same API inference as our framework but with category label names only, removing concept definitions. This baseline isolates the contribution of explicit concept definitions by holding the inference model constant.

Table 1: Baseline composition

Baseline	Inference paradigm	Prompt
Emb-label	Embedding (mpnet)	Label only
Emb-concept	Embedding (mpnet)	Concept
LLM-label	Generative LLM (GPT)	Label only
LLM-concept	Generative LLM (GPT)	Concept

Emb-concept paraphrase-multilingual-mpnet-base-v2 (Reimers and Gurevych, 2019) with per-label concept descriptions extracted from the full prompt, classified via argmax cosine similarity. This baseline isolates the contribution of LLM-based generative reasoning over embedding-based similarity, holding concept definitions constant. The concept-definition prompts are applied as-is to maintain fair comparison; any performance gap thus reflects differences in inference paradigm rather than prompt design.

Emb-label Embedding-based classification with label name strings only, corresponding to classic zero-shot embedding similarity. This is the weakest baseline, with neither generative LLM reasoning nor explicit concept definitions.

Together, these four conditions isolate the independent contributions of concept-guided prompting and generative LLM-based inference. While supervised encoder fine-tuning has shown effectiveness in resource-rich settings, our experimental setup is characterized by limited labeled data and a zero-shot evaluation setting, which precludes its direct use as a baseline. We therefore restrict our comparison to embedding-based similarity and prompting-based approaches, which more naturally reflect the realistic deployment conditions of large-scale R&D classification where labeled data is scarce.

4 Dataset and Experimental Setup

4.1 Dataset

We use project records from the Korean National Science and Technology Information Service (NTIS) for the year of 2025. Each record contains a project title, research objective, abstract, and expected outcomes in Korean. The dataset comprises 57,477 projects with non-empty classification text, with 59 records excluded during preprocessing (11 test records and 48 records with missing content, $<0.1\%$).

4.2 Implementation

All LLM-based experiments use the OpenAI Batch API with `gpt-4o-mini` at temperature 0 for reproducibility. Embedding-based experiments use `paraphrase-multilingual-mpnet-basev2` (Reimers and Gurevych, 2019) with maximum sequence length 128. Per-label concept prompts for the embedding baseline are designed to fit within the model's maximum sequence length to ensure full encoding of conceptual criteria.

4.3 Evaluation

Validation set. To evaluate classification performance, we construct a manually annotated validation set of 742 projects through stratified sampling. Stratified sampling is preferred over random sampling because the label distribution is heavily skewed toward lower AX levels, while higher autonomy categories such as LEVEL_3 are relatively rare.

For Stage 1, 300 projects are sampled uniformly across the three predicted categories, yielding 100 projects per label. For Stage 2, 500 projects are initially sampled from projects predicted as SERVICE. During annotation, 58 projects are re-labeled as NON_SERVICE, reflecting Stage 1 false positives. The remaining 442 projects constitute the Stage 2 validation set.

Because Stage 2 classification is applied only to projects predicted as SERVICE in Stage 1, the effective evaluation size for AX classification differs across ablation conditions.

Annotation. Annotation was performed by a single domain expert in Korean S&T policy, without access to model predictions. The annotator was provided with explicit labeling guidelines derived from the concept definitions in Section 3.2, specifying that classification should be based on the direct output of a project rather than its anticipated downstream application. The validation set reflects the judgment of a single domain expert, and inter-annotator agreement was not measured due to resource constraints at the time of submission. Future work should incorporate a second annotator with measured Cohen's κ to provide more reliable gold-standard labels and validate annotation consistency.

Metrics. We report accuracy and macro F1-score across all conditions. Macro F1 is the primary metric, as it treats all classes equally regardless of frequency, which is appropriate given the skewed

label distribution. In addition to overall metrics, we report per-class F1 scores for each category in both stages (Table 3 and Table 4), enabling fine-grained analysis of classification performance across majority and minority classes. Statistical significance is assessed using McNemar's test for all four pairwise comparisons corresponding to the 2×2 ablation design, with significance threshold $\alpha = 0.05$. 95% confidence intervals for macro F1 are estimated via bootstrap resampling ($n = 1000$).

5 Results

Table 2 presents classification performance across all four conditions of the 2×2 ablation study. Across both stages and evaluation settings, LLM-concept consistently achieves the highest Macro F1, demonstrating the effectiveness of explicit conceptual operationalization for concept-sensitive classification tasks. Macro F1 is the primary metric in this study because the label distributions are highly imbalanced, particularly in Stage 2 where higher autonomy categories such as LEVEL_3 are relatively rare. Unlike accuracy, Macro F1 penalizes systems that ignore minority categories or overpredict dominant labels.

McNemar's test confirms that explicit concept definitions significantly improve LLM-based classification performance in both stages (SVS: $p = 0.025$; AX: $p < 0.001$). The effect of inference paradigm is also significant in Stage 2 when concept definitions are held constant ($p = 0.007$), indicating that generative LLM-based reasoning substantially outperforms embedding-based similarity for operationally defined AX categories. In contrast, no significant paradigm difference is observed in Stage 1 under concept-guided conditions ($p = 0.708$), suggesting that broad SERVICE vs. NON_SERVICE discrimination is less dependent on generative reasoning than fine-grained autonomy classification.

A notable result is that LLM-label achieves the highest raw accuracy in Stage 1 (0.637) despite lower Macro F1 (0.462). This behavior reflects systematic overclassification toward majority categories rather than balanced semantic discrimination. As shown in Table 3, LLM-label predicts SERVICE for 53.8% of all projects while producing almost no UNCERTAIN predictions (0.2%). Correspondingly, its UNCERTAIN F1 is near zero (0.056). Because label-only prompting provides no explicit operational criteria for ambiguity handling, the model exhibits overconfident classification behavior driven primarily by latent label semantics.

In contrast, LLM-concept produces substantially more balanced label distributions. SERVICE predictions decrease from 53.8% to 20.8%, while UNCERTAIN predictions increase from 0.2% to 9.6%, closely aligning with the gold annotation distribution. This shift results in the highest Stage 1 Macro F1 (0.506) and the strongest UNCERTAIN performance (F1 = 0.237). These findings suggest that explicit concept definitions function not merely as descriptive prompts, but as operational semantic constraints that regulate inference behavior and improve calibration under ambiguity.

The effect of concept-guided inference is even more pronounced in Stage 2 AX classification. LLM-concept achieves the highest overall Macro F1 (0.537), substantially outperforming Emb-concept (0.322) and LLM-label (0.311). The largest gains appear in semantically nuanced categories requiring operational reasoning rather than topical similarity. In particular, LEVEL_3 (agentic AI) achieves an F1 score of 0.684 under LLM-concept, compared to 0.417 for LLM-label and 0.078 for Emb-label. Because LEVEL_3 is defined primarily by autonomous planning and end-to-end execution capability rather than surface lexical features, this result suggests that concept-guided generative inference better captures operational semantic distinctions that are difficult to represent through embedding similarity alone.

Presented in Table 4 and Table 5, per-class analysis further reveals that LLM-concept is the only condition to achieve meaningful UNCERTAIN classification performance in both stages (SVS: 0.237; AX: 0.355), while all other conditions score near zero. This pattern indicates that explicit conceptual operationalization enables calibrated abstention behavior by providing explicit semantic criteria for ambiguity handling. In contrast, label-only prompting suppresses uncertainty reasoning and encourages forced categorical assignment even for borderline cases.

Table 6 and Table 7 present confusion matrices for LLM-concept. In Stage 1, the most common error pattern is the mapping of NON_SERVICE projects to UNCERTAIN, suggesting that the abstention criterion is conservatively applied to borderline non-service cases. In Stage 2, the largest confusion occurs between LEVEL_1, LEVEL_2, and UNCERTAIN categories, reflecting genuine semantic overlap between AI-assisted decision support, partially

Table 2: Classification performance across all four conditions. A. full set: SVS 3-class, AX 5-class. B. excluding UNCERTAIN: UNCERTAIN class excluded. Macro F1 with 95% bootstrap CI in parentheses. Best results in **bold**.

		A. full set		B. excluding UNCERTAIN	
		SVS	**AX**	**SVS**	**AX**
Acc.	Emb-label	0.478	0.348	0.700	0.411
	Emb-concept	0.555	0.428	0.704	0.504
	LLM-label	0.637	0.303	0.717	0.349
	LLM-concept	**0.543**	**0.545**	**0.812**	**0.675**
Macro F1	Emb-label	0.397 (0.349, 0.443)	0.174 (0.138, 0.211)	0.693 (0.625, 0.755)	0.240 (0.190, 0.296)
	Emb-concept	0.422 (0.374, 0.473)	0.322 (0.273, 0.364)	0.684 (0.618, 0.743)	0.441 (0.377, 0.502)
	LLM-label	0.462 (0.417, 0.510)	0.311 (0.270, 0.347)	0.709 (0.651, 0.763)	0.411 (0.363, 0.460)
	LLM-concept	**0.506** (0.450, 0.558)	**0.537** (0.491, 0.581)	**0.808** (0.749, 0.865)	**0.681** (0.633, 0.729)

Table 3: Label distribution (%) across all four conditions for Stage 1 (SVS) and Stage 2 (AX). Values may not sum to 100 due to rounding.

	Emb-label	**Emb-concept**	**LLM-label**	**LLM-concept**
	Stage 1 (SVS)			
SERVICE	32.8	23.2	53.8	20.8
NON_SERVICE	56.0	69.7	46.0	69.7
UNCERTAIN	11.2	7.1	0.2	9.6
	Stage 2 (AX)			
NON_AX	14.6	34.3	53.5	42.4
LEVEL_1	71.7	42.0	12.8	19.2
LEVEL_2	5.3	2.8	32.6	17.4
LEVEL_3	3.7	17.6	1.0	5.4
UNCERTAIN	4.8	3.4	0.0	15.6

autonomous execution, and ambiguous autonomy descriptions within project documents. Importantly, LEVEL_3 maintains relatively strong separation from lower autonomy categories, with 65 correct predictions and limited confusion with LEVEL_1, indicating that the concept-guided framework successfully captures higher-order autonomy distinctions.

Taken together, the results suggest that explicit conceptual operationalization fundamentally alters semantic inference behavior in large language model classification systems. Rather than relying solely on latent semantic associations between labels and text, concept-guided inference enables operational reasoning over category-defining properties, resulting in improved minority-class discrimination, calibrated uncertainty handling, and more balanced semantic classification behavior.

6 Discussion

6.1 Overall Performance

The experimental results suggest that concept-sensitive classification cannot be fully addressed through latent semantic similarity alone, particularly when category boundaries depend on operational rather than topical criteria. This distinction is especially evident in the AX task, where LEVEL_1, LEVEL_2, and LEVEL_3 are differentiated primarily by autonomy, planning, and execution capability rather than domain vocabulary. Under these conditions, embedding-based methods show limited discrimination ability even with concept descriptions, whereas concept-guided LLM inference achieves substantially stronger performance.

The findings further indicate that explicit conceptual operationalization systematically alters LLM inference behavior. Without concept definitions, label-only prompting encourages overconfident predictions driven by latent associations between labels and project text, resulting in highly skewed label distributions and near-zero UNCERTAIN performance. In contrast, concept-guided inference produces more balanced predictions and substantially improved uncertainty handling, suggesting that explicit operational criteria function as semantic constraints that regulate inference behavior.

The results also highlight uncertainty handling as an important but often overlooked aspect of LLM-based classification. Existing pipelines typically assume forced categorical assignment even

Table 4: Per-class F1 scores for Stage 1 (SVS). Best results in **bold**.

Condition	SERVICE	NON_SERVICE	UNCERTAIN	Macro F1
Emb-label	0.573	0.559	0.059	0.397
Emb-concept	0.546	0.661	0.059	0.422
LLM-label	0.596	**0.734**	0.056	0.462
LLM-concept	**0.678**	0.604	**0.237**	**0.506**

Table 5: Per-class F1 scores for Stage 2 (AX). Best results in **bold**.

Condition	NON_AX	LEVEL_1	LEVEL_2	LEVEL_3	UNCERTAIN	Macro F1
Emb-label	0.148	0.555	0.091	0.078	0.000	0.174
Emb-concept	0.468	0.531	0.154	0.456	0.000	0.322
LLM-label	**0.660**	0.203	0.275	0.417	0.000	0.311
LLM-concept	0.667	**0.590**	**0.390**	**0.684**	**0.355**	**0.537**

for ambiguous or incomplete inputs. However, real-world R&D data frequently contains borderline cases that do not clearly satisfy operational criteria. The strong improvement of UNCERTAIN classification under LLM-concept conditions suggests that explicit concept definitions can support calibrated abstention behavior through interpretable ambiguity thresholds, which may be particularly important in policy and governance contexts where overclassification can distort statistical interpretation.

The proposed framework also has broader implications beyond national R&D classification. Many real-world categorization tasks—including regulatory compliance assessment, policy analysis, scientific taxonomy construction, and emerging technology governance—involve conceptually defined categories that cannot be reliably separated through surface-level similarity alone. By explicitly separating conceptual definitions from inference mechanisms, the framework enables category semantics to be modularly updated, transparently specified, and independently evaluated, which may be particularly valuable in domains where classification criteria evolve over time or require policy-sensitive interpretation.

6.2 Error Analysis

Analysis of misclassified cases reveals three major error patterns.

False positives: Over-generalization of humanities research. The most common false positives involved pure humanities and social science projects being classified as SERVICE. For example, a corpus linguistics study on immigrant discourse and a longitudinal study on elderly housing environments were classified as SERVICE based on their social science framing despite lacking direct relevance to service delivery or innovation. This suggests that the criterion for service-related humanities research—*"research on customer psychology, consumer behavior, and other service-related humanities and social sciences"*—was interpreted too broadly, extending to research with only indirect social relevance.

False negatives: Boundary cases between basic and applied research. The most common false negatives involved basic research projects whose expected outcomes included potential service applications. Quantum communication theory, electromagnetic wave research, and statistical modeling studies were annotated as SERVICE by human annotators based on anticipated downstream applications, but classified as NON_SERVICE by the model based on the immediate research objective. This reflects an inherent ambiguity in the task definition: whether service R&D classification should depend on direct research outputs or intended downstream applications. We adopt the former criterion, consistent with standard R&D policy frameworks, though this may under-classify basic research with strong service orientation.

Product-service fusion cases. A third pattern involved projects combining hardware development with service delivery, such as AI-based fire detection systems and augmented analytics platforms. These cases were inconsistently classified by both the model and human annotators, suggesting ambiguity in the classification framework itself rather than simple model error. Future work could introduce a dedicated PRODUCT_SERVICE_FUSION category for such hybrid cases.

Table 6: Confusion matrix for LLM-concept, Stage 1 (SVS). Rows: predicted labels. Columns: gold labels.

	SERVICE	NON_SERVICE	UNCERTAIN
SERVICE	60	9	8
NON_SERVICE	25	87	76
UNCERTAIN	15	4	16

Table 7: Confusion matrix for LLM-concept, Stage 2 (AX). Rows: predicted labels. Columns: gold labels.

	NON_AX	LEVEL_1	LEVEL_2	LEVEL_3	UNCERTAIN
NON_AX	40	2	3	0	16
LEVEL_1	8	79	36	8	42
LEVEL_2	1	8	30	13	5
LEVEL_3	0	2	22	65	5
UNCERTAIN	10	4	6	10	27

6.3 Implications for Concept Layer Design

The observed error patterns suggest several directions for refining concept definitions. First, the criterion for service-related humanities research could be narrowed to require direct relevance to service delivery or operations, excluding purely academic or policy-oriented studies. Second, introducing a clearer distinction between basic research and applied service development could reduce false negatives in projects where service applications are anticipated but not central to the research objective.

Importantly, such refinements would require corresponding updates across both embedding- and LLM-based conditions to preserve comparability under the shared concept layer. We leave systematic concept optimization to future work, as the primary contribution of this study is to establish the concept-governed classification framework and demonstrate its effectiveness for concept-sensitive semantic inference rather than to optimize performance for a specific annotation scheme.

7 Conclusion

We proposed a concept-sensitive classification framework for large-scale national R&D project texts. By encoding explicit conceptual definitions as a modular component of the classification pipeline, the proposed approach enables controllable and interpretable classification of conceptually defined categories using LLM-based inference.

Applied to a two-stage pipeline for service R&D filtering and AX stage classification over nearly 58,000 NTIS project records, the framework demonstrates consistent improvements over both embedding-based similarity and label-only LLM prompting, particularly for operationally defined and semantically ambiguous categories.

The results further suggest that explicit conceptual operationalization can improve semantic controllability, uncertainty handling, and discrimination of higher-order operational categories that are difficult to capture through latent semantic similarity alone.

Because the concept layer is modular and independently editable, the framework also supports transparent policy updates, cross-domain extensibility, and interpretable governance-oriented classification workflows at scale.

Limitations

The proposed framework relies on `gpt-4o-mini` for LLM-based inference, which introduces API costs and potential rate-limit constraints at large scale. Although the Batch API substantially reduces inference cost, deployment in resource-constrained environments may require adaptation to smaller open-source models whose concept-following and reasoning capabilities may be weaker.

The study focuses on Korean national R&D project data and two specific classification stages, which may limit direct generalizability to other domains, languages, or policy frameworks. In addition, the concept definitions were manually designed based on policy-oriented operational criteria, meaning that classification quality may depend on the completeness and consistency of the conceptual framework itself.

The manually validated dataset remains relatively limited in size despite stratified sampling. Larger annotated datasets would provide more reliable performance estimates, particularly for low-frequency categories such as LEVEL_3 (agentic AI), where class imbalance remains substantial. In

addition, the study did not include formal inter-annotator agreement analysis. Because several categories involve operational and policy-dependent distinctions, annotation consistency may be sensitive to subjective interpretation in semantically ambiguous cases such as basic research with downstream service applicability or product–service fusion projects.

The 2×2 ablation design also carries methodological limitations. First, the concept definitions used in the **Emb-concept** condition were originally designed for generative LLM inference and may not be optimal for dense vector encoding. Given the 128-token sequence limit of `paraphrase-multilingual-mpnet-base-v2`, portions of the concept definitions may be truncated or encoded suboptimally, meaning that part of the observed performance gap between **Emb-concept** and **LLM-concept** may reflect prompt–encoder mismatch rather than inference paradigm alone. Second, the embedding condition was evaluated using a limited set of multilingual encoders and LLM configurations. Stronger embedding models such as `multilingual-e5-large`, alternative reasoning architectures, retrieval augmentation, or domain-specialized language models may reduce the observed gap. Accordingly, the comparison should be interpreted as reflecting the specific pairing of `paraphrase-multilingual-mpnet-base-v2` and `gpt-4o-mini`, rather than a universal comparison between embedding-based and generative approaches.

Future work may further explore automated concept refinement, hierarchical concept structures, and adaptive uncertainty calibration mechanisms. More broadly, the findings suggest that reliable LLM-based classification systems may depend not only on increasing model scale, but also on improving the explicit representation and governance of category semantics themselves.

Ethical Considerations

This work classifies publicly available national R&D project records from the NTIS database. No personally identifiable information is used or disclosed. Automated classification results should be reviewed by domain experts before informing policy decisions, as classification errors may affect resource allocation or policy analysis.

Acknowledgments

The authors used Claude (Anthropic) for coding assistance in implementing the classification pipeline and for language assistance in drafting this manuscript. All research ideas, experimental design, and interpretations are the authors' own.

References

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of EMNLP-IJCNLP*, pages 3615–3620.

Benjamin Clavié, Alexandru Ciceu, Frederick Naylor, Guillaume Soulié, and Thomas Brightwell. 2023. Large language models in the workplace: A case study on prompt engineering for job type classification. In *Natural Language Processing and Information Systems (NLDB 2023)*, volume 13913 of *Lecture Notes in Computer Science*, pages 3–18. Springer.

Washington Cunha, Leonardo Rocha, and Marcos André Gonçalves. 2025. A thorough benchmark of automatic text classification: From traditional approaches to large language models. *arXiv preprint arXiv:2504.01930*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Erkan Gunes and Christoffer Koch Florczak. 2025. Replacing or enhancing the human coder? multiclass classification of policy documents with large language models. *Journal of Computational Social Science*, 8(2):31.

Hyejin Jang, Taeyeon Roh, and Byungun Yoon. 2025. Public values in public R&D through natural language processing. *Scientific Reports*, 15:37397.

Qian Li, Hao Peng, Jianxin Li, Congying Xia, Renyu Yang, Lichao Sun, Philip S. Yu, and Lifang He. 2022. A survey on text classification: From traditional to deep learning. *ACM Transactions on Intelligent Systems and Technology*, 13(2).

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of EMNLP-IJCNLP*, pages 3982–3992.

Ivo Simões, Hugo Gonçalo Oliveira, and João Correia. 2026. Field of science and technology classification of academic documents in Portuguese. In *Proceedings of PROPOR*, pages 1021–1026.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. In *Proceedings of EMNLP-IJCNLP*, pages 3914–3923.