CARMA 2026

8th Int. Conf. on Advanced Research Methods and Analytics
1-3 July 2026, Valencia, Spain
DOI: https://doi.org/10.4995/CARMA2026.2026.21939

Between the Embedding and the Prompt: Systematic Design Effects in LLM-based Occupation Coding

Olga Kononykhina[1,2], Anna-Carolina Haensch[1,2,3], Frauke Kreuter[1,2,3]
[1]Ludwig-Maximilians-Universität Munich, [2]Munich Center for Machine Learning (MCML), [3]University of Maryland, College Park, USA

How to cite: Kononykhina, O.; Haensch, A.-C.; Kreuter, F. (2026). Between the Embedding and the Prompt: Systematic Design Effects in LLM-based Occupation Coding In: 8th International Conference on Advanced Research Methods and Analytics (CARMA 2026). Valencia, 1-3 July 2026. https://doi.org/10.4995/CARMA2026.2026.21939

Abstract

Assigning free-text job descriptions to standardised taxonomies is a persistent bottleneck in survey research and official statistics. Large language models (LLMs) offer a promising path toward automation, but each step in the pipeline involves both model architecture and measurement choices about how an occupation should be represented. Through 119 experiments on German survey data, we systematically vary the textual representation of occupational categories, embedding models, LLMs, and prompt design. Category representation changes retrieval accuracy by 8–23 percentage points and classification by 11–21. Prompt role and abstention behaviour are model-specific and must be validated before deployment. The dominant source of variance, however, sits outside model measurement choices. How respondents describe their work matters more than any model or design choice (ICC = 0.76).

Keywords: *occupational coding, LLM, embeddings, text classification, survey*

1. Introduction

Occupational coding serves as a foundation for obtaining standardised information that supports employment analytics, migration policies or epidemiological research. Large hierarchical taxonomies such as the German "Klassifikation der Berufe" (KldB, ~1300 categories) offer an analytical destination for the task, but the coding itself remains a persistent bottleneck. If implemented manually it requires support from highly trained coders, and for large surveys the coding takes months.

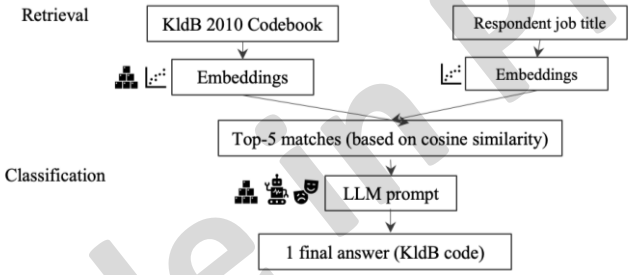
Several automatic solutions have been developed to ease the task, including rule-based systems such as CASCOT (Jones & Elias, 2004), ensemble classifiers such as SOCcer (Russ et al., 2023) and machine learning models such as OccuCoDe (Simson et al., 2023). They show mixed results limited by contextual ambiguity, overly generic job titles, scarce training data, and the evolving

Article in Press

This work is licensed under a Creative Commons License CC BY-NC-SA 4.0
Editorial Universitat Politècnica de València

nature of both jobs and coding systems. Large language models (LLMs) offer new potential: they can capture semantic context, and adapt flexibly to new inputs. Early studies using embeddings and LLMs achieve up to 80% accuracy (Achananuparp et al., 2025; Bach et al., 2025; Sturgis et al., 2025).

However, current approaches to the task tell an engineering story, focused on benchmarking specific models and pipelines while treating the textual inputs to these systems as given. Occupational classification is also a measurement problem that spans both respondent input and model architecture (Jacobs & Wallach, 2021; Radford & Joseph, 2020). Assigning an answer to a standardized taxonomic category requires operationalization at several points, including how the respondent describes their work, how each taxonomic category is textually represented, and how the prompt frames the classification task. Each of these decisions is a measurement choice with consequences for the validity of the resulting code. Two of these operationalizations, both on the model side, remain understudied. It is unclear (1) how different occupational components available in taxonomies (descriptions, tasks, example titles) affect retrieval and classification performance, and (2) how prompt design choices, including the role assigned to the LLM, interact with model selection.



7 occupational components (OC): Descriptions (D), Tasks (T), Examples (E), D + T, D + E, T + E, D + T + E

5 embedding models (EM): Alibaba-NLP/gte-multilingual-base (Zhang et al., 2024); Paraphrase-multilingual-MiniLM-L12-v2 (Reimers & Gurevych, 2019); Gemini:text-embedding-004 (Google, 2026), IBM:granite-embedding-278m-multilingual (ibm-granite, 2025); GPT:text-embedding-3-large (OpenAI, 2025b)

6 models (LLM): GPT-4o-mini (OpenAI, 2025a); Llama-3.1-70b-instruct (meta-llama, 2025); DeepSeek-R1 (deepseek-ai, 2025); Mistral-small (Mistral AI, 2025); Gemini-1.5-flash (Gemini Team et al., 2024); Qwen2.5-72b-instruct (Qwen, 2025)

2 Role types (Role): LLM as a coding expert ("You act as an experienced expert in occupational classification according to the German KldB 2010 system"); LLM as a respondent ("Imagine you are participating in a survey about your professional activity. You are [gender] and [age] years old. Your job title is: [job title].")

Figure 1. Scheme of the experiments

This paper addresses these gaps through 119 retrieval and classification experiments on German survey data coded into the KldB 2010. We systematically vary seven occupational component

Article in Press

combinations, five embedding models, six LLMs, and two prompt roles, and analyze their effects on successful job classification. Our results show that measurement choices, namely how respondents describe their work, how occupational categories are textually represented and how prompts are structured, explain more variance in accuracy than the choice of model itself. These findings offer methodological guidance for researchers who implement LLM-based occupational coding. In principle, the same measurement considerations apply wherever free-text responses must be mapped onto standardised taxonomies, including the International Classification of Diseases (ICD) in medicine or product classifications in trade.

2. Background

Recent research on LLMs for occupational coding has clustered around three technical approaches: direct prompting, embedding-based retrieval/RAG, and fine-tuning. Studies show that LLMs struggle when asked directly to classify occupations (Achananuparp et al., 2025; Li et al., 2025). An alternative approach relies on embeddings either as a standalone method or as a first-stage filter that reduces the set of candidate categories prior to LLM-based classification. Embedding-only or RAG-based accuracies range from 0.32 to 0.81 depending on the taxonomy, model, and language (Achananuparp et al., 2025; Bach et al., 2025; Li et al., 2025). A third line of work fine-tunes models on occupational data. Safikhani et al. (2023) fine-tuned BERT and GPT-3 on German KldB, achieving 64% and 30% accuracy, respectively. Fine-tuning offers competitive accuracy but is expensive and struggles with hierarchical taxonomies, improving broad category identification while worsening fine-grained distinction (Li et al., 2025).

Across all three approaches, prior work treats pipeline inputs as fixed; this paper treats them as design variables with consequences for validity (Jacobs & Wallach, 2021). No study has systematically compared how different occupational components shape retrieval and classification accuracy, nor has the role assigned to the LLM been varied beyond the default "expert" coder.

3. Methodology

3.1. Data

Two data sources provide the empirical foundation for this paper. The reference set is the official codebook of 1,286 German occupations (KldB 2010). Each occupation is specified by its 5-digit code, occupation title, description, associated tasks, and examples (Bundesagentur für Arbeit, 2010). The query set comes from a probability-based nationwide telephone survey conducted by the Institute for Applied Social Science (infas) in Germany between 1 April and 30 June 2019. Of 1,415 adults aged 18+, 1,379 had work experience and were asked: "What is

Article in Press

the professional activity that you mainly perform?" infas in-house expert coders double-coded all responses, which we use as the gold standard.

3.2. Experiments

During retrieval, we embed the respondent's answer (typically a job title) and each KldB category's textual representation, then retrieve the five categories with the highest cosine similarity to each respondent's answer. We vary two dimensions: (a) the occupational component (OC) used to construct each KldB category's textual representation, and (b) the embedding model (EM). This yields 35 retrieval conditions. During classification, the top-5 candidates from the best-performing retrieval condition (GPT embeddings with D+E, see Section 4.1) are passed to an LLM via zero-shot prompting. We vary three dimensions: (1) the LLM role (Role), either occupational coding expert or survey respondent; (2) the occupational component (OC) used to present the top-5 candidates to the LLM (same seven variants as retrieval); and (3) the LLM model (LLM), resulting in 84 experiments. All models run at temperature 0, with the top-5 order randomized to avoid positional bias. To keep the experiments tractable in time and cost, we draw a random sample of 400 responses from the query dataset.

3.3. Analysis

Responses in the query dataset were spellchecked and gendered. For retrieval, we compute cosine similarity between each job title embedding and all reference embeddings, retaining the top-5 most similar categories. Retrieval accuracy equals 1 when the gold code appears in the top-5, and 0 otherwise. To identify the optimal retrieval configuration, we fit a generalized linear mixed-effects model:

$$\text{logit}\, P(Y_{ij} = 1) = \beta_0 + \beta_1 \,\text{EM}_{ij} + \beta_2 \,\text{OC}_{ij} + \beta_3 \,(\text{EM} \times \text{OC})_{ij} + u_j. \quad (1)$$

For classification, accuracy equals 1 when the LLM selects the gold code or correctly abstains ("None of the above") when the gold code is absent from the top-5. A second mixed-effects model examines the effects of role, occupational component, and LLM model on classification accuracy:

$$\text{logit}\, P(Y_{ij} = 1) = \beta_0 + \beta_1 Role_{ij} + \beta_2 LLM_{ij} + \beta_3 OC_{ij} + \beta_4 (Role \times LLM)_{ij} + \beta_5 (Role \times OC)_{ij} + \beta_6 (LLM \times OC)_{ij} + u_j. \quad (2)$$

In both models, the random intercept u_j captures respondent-level variation, OC denotes the occupational component, EM the embedding model, LLM the language model used for classification, and Role the prompt type (expert or respondent).

Article in Press

4. Results

After excluding non-codable responses (e.g., "student," "housewife"), the cleaned dataset comprised 1,336 observations for retrieval and a random subsample of 400 for classification.

4.1. Does occupational component design affect retrieval accuracy?

Table 1 shows that component design affects the accuracy and the effect is substantial. Depending on which component is used, retrieval accuracy varies by 8 percentage points for GPT, and by 18–23 percentage points for gte, Gemini, and ibm embeddings. For practitioners choosing smaller, locally deployable models, component design becomes a primary lever for performance. Across models, combining components significantly improves retrieval over single components. The mixed-effects regression confirms that D+E, T+E, and D+T+E all outperform individual components (all $p < 0.001$). Descriptions+Examples (D+E) emerged as the most robust combination, improving accuracy for GPT, Gemini, and MiniLM without reducing performance for any model. GPT embeddings significantly outperformed all alternatives (all $p < 0.001$), achieving 62–69% accuracy across components. The random intercept revealed an ICC of 0.77, indicating that the quality and specificity of the job title provided explain more variance in retrieval accuracy than either model or component choice. Based on these results, we selected GPT embeddings with D+E as the input for all classification experiments.

Table 1: Retrieval accuracy rate (%) based on the top-5 most similar occupational components.
Note: D, T, and E stand for Descriptions, Task, and Examples, respectively. Bold font highlights the highest accuracy for each model

Component	GPT	MiniLM	gte	IBM:granite	Gemini
D	62	34	49	48	47
T	63	30	39	32	39
E	63	31	52	52	47
D+T	65	38	51	49	51
D+E	67	**40**	**57**	**55**	**58**
T+E	**69**	30	53	49	54
D+T+E	68	39	56	**55**	57

4.2. Does prompt design affect classification accuracy?

Table 2 reports classification accuracy across six LLMs, seven occupational components, and two prompt roles. As in retrieval, component design has a substantial effect, but the pattern reverses. While retrieval benefited from compound components (D+E), classification accuracy is often highest when LLMs choose from Examples alone, and lowest when Tasks are the prime source of reference. The gap is large: Gemini's accuracy dropped by 18 percentage points in expert mode and 22 points in respondent mode when Tasks replaced Examples.

Article in Press

The role assigned to the LLM had a model-dependent effect. While the expert role has been the universal default in prior research, our results show that prompting a model as a survey respondent can sometimes improve accuracy. For example, GPT benefited modestly from the switch ($\beta = 0.19$, $p = .019$; OR = 1.21). However, the same switch reduced accuracy for Gemini ($\beta = -0.28$, $p = .014$; OR = 0.75) and Llama ($\beta = -0.34$, $p = .003$; OR = 0.71). This means that the choice of prompt role should be treated and tested as a model-specific design decision rather than a universal default. As in retrieval, respondent-level variation dominated: the ICC of 0.76 again indicates that the quality of the original job title explains more variance than any pipeline design choice.

Models also differed in abstention behaviour. When the gold code was absent from the top-5 (33% of cases), each prompt included an explicit "None of the above" option. With Examples in expert mode, GPT correctly abstained 45% of the time and DeepSeek 44%, while Mistral managed only 5%. Prompt wording around abstention should therefore be tested per model before deployment, since identical instructions do not produce comparable behaviour across them.

Table 2: Top-1 Classification Accuracy Rate (%) across Roles, LLMs and Occupational components. Note: D, T, and E stand for Descriptions, Task, and Examples, respectively. Bold font highlights the highest accuracy for each model and each role

	Prompt: Expert						**Prompt: Respondent**					
OC	Deep-Seek	Mistral	Gemini	GPT	Llama	Qwen	Deep-Seek	Mistral	Gemini	GPT	Llama	Qwen
D	54	42	56	55	51	50	54	41	55	57	48	53
T	49	36	47	46	44	43	50	30	44	46	42	44
E	**64**	**50**	**64**	**64**	56	56	**64**	**50**	**65**	**63**	54	**61**
D+T	54	38	56	53	49	51	52	33	52	55	45	48
D+E	58	**50**	**64**	60	**58**	**57**	58	48	63	62	**56**	58
T+E	58	44	58	52	54	56	58	39	59	59	55	56
D+T+E	59	42	60	59	55	54	60	38	61	60	**56**	56

5. Discussion

Our experiments show systematic effects of pipeline design choices on LLM-based occupational coding accuracy. The textual representation of occupational categories matters at every stage, but in opposite directions. Retrieval improved with compound representations (e.g., Descriptions + Examples), while classification accuracy was highest when LLMs chose from Examples alone, dropping by up to 12 percentage points when compound variants were used as a reference. More broadly, the representation best suited to retrieving plausible candidates is not the one best suited to discriminating among them, even when both stages operate on the

Article in Press

same construct. Prompt role and abstention behaviour also varied substantially across models, and both must be calibrated and validated for the specific deployment context before use.

Beyond the 119 tested conditions, exploratory analyses suggest that weighted component combinations, enriched respondent input, and abstention-prompt wording each merit dedicated investigation. All of these design choices, however, operate within a ceiling set by respondent input. The Intraclass Correlation Coefficient (ICC) of 0.76 indicates that how respondents describe their occupations explains far more variance than model, component, or role selection. This suggests that beyond optimising the pipeline, researchers should invest in eliciting better occupational responses at the point of data collection, for instance through LLM-assisted follow-up probing when an initial answer is too vague (Sturgis et al., 2025).

Our findings carry several limitations: we used only zero-shot prompting; inputs were short survey job titles; evaluation targeted the full five-digit KldB code without examining where errors concentrate in the hierarchy; results are based on a German classification (and English language taxonomies and data might have more stable performance across tested conditions); and reliance on proprietary models limits reproducibility. Nonetheless, the core message we aim to share is that adopting LLMs for occupational coding is as much a measurement problem as a technology choice. The same model can perform well or poorly depending on how occupational categories are operationalised and how the prompt is framed. Wherever free-text responses must be mapped onto standardised taxonomies, whether in occupational, medical, or product classification, how the mapping is operationalised matters as much as the model that performs it.

References

Achananuparp, P., Lim, E.-P., & Lu, Y. (2025). A Multi-Stage Framework with Taxonomy-Guided Reasoning for Occupation Classification Using Large Language Models (arXiv:2503.12989). arXiv. https://doi.org/q6rg

Bach, R., Klamm, C., Heyne, S., Kogan, I., Kononykhina, O., & Jarck, J. (2025). From Job Titles to ISCO Codes: Enhancing Occupational Classification With RAG-based LLMs. OSF. https://doi.org/q6rh

Bundesagentur für Arbeit. (2010). KldB 2010 Online Ausgabe. Statistik der Bundesagentur für Arbeit. https://shorturl.at/N0n84

deepseek-ai. (2025). Deepseek-ai/DeepSeek-R1 [Computer software]. DeepSeek. https://github.com/deepseek-ai/DeepSeek-R1 (Original work published 2025)

Gemini Team, Georgiev, P., Lei, V. I., Burnell, R., Bai, L., Gulati, A., Tanzer, G., Vincent, D., Pan, Z., Wang, S., Mariooryad, S., Ding, Y., Geng, X., Alcober, F., Frostig, R., Omernick, M., Walker, L., Paduraru, C., Sorokin, C., … Vinyals, O. (2024). Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context (arXiv:2403.05530). arXiv. https://doi.org/q6rk

Google. (2026). Models | Gemini API. Google AI for Developers. https://ai.google.dev/gemini-api/docs/models

ibm-granite. (2025). Ibm-granite/granite-embedding-278m-multilingual · Hugging Face. https://huggingface.co/ibm-granite/granite-embedding-278m-multilingual

Article in Press

Jacobs, A. Z., & Wallach, H. (2021, March 1). Measurement and Fairness. https://doi.org/gjh2p4

Jones, R., & Elias, P. (2004). Cascot: Computer Assisted Structured Coding Tool. https://warwick.ac.uk/fac/soc/ier/data_group/cascot/choose_classificatio/

Li, N., Kang, B., & De Bie, T. (2025). LLM4Jobs: Unsupervised occupation extraction and standardization leveraging Large Language Models. Knowledge-Based Systems, 316, 113302. https://doi.org/q6rr

meta-llama. (2025). Llama-models/models/llama3_1/MODEL_CARD.md at main · meta-llama/llama-models. GitHub. https://github.com/meta-llama/llama-models/blob/main/models/llama3_1/MODEL_CARD.md

Mistral AI. (2025). Bienvenue to Mistral AI Documentation | Mistral AI. https://docs.mistral.ai/

OpenAI. (2025a). GPT-4o mini: Advancing cost-efficient intelligence. https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/

OpenAI. (2025b). Vector embeddings | OpenAI API. https://developers.openai.com/api/docs/guides/embeddings/

Qwen. (2025). Qwen2.5: A Party of Foundation Models! | Qwen. https://qwenlm.github.io/blog/qwen2.5/

Radford, J., & Joseph, K. (2020). Theory In, Theory Out: The Uses of Social Theory in Machine Learning for Social Science. Frontiers in Big Data, 3, 18. https://doi.org/gnn2wx

Reimers, N., & Gurevych, I. (2019, November). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. http://arxiv.org/abs/1908.10084

Russ, D. E., Josse, P., Remen, T., Hofmann, J. N., Purdue, M. P., Siemiatycki, J., Silverman, D. T., Zhang, Y., Lavoué, J., & Friesen, M. C. (2023). Evaluation of the updated SOCcer v2 algorithm for coding free-text job descriptions in three epidemiologic studies. Annals of Work Exposures and Health, 67(6), 772–783. https://doi.org/q6rq

Safikhani, P., Avetisyan, H., Föste-Eggers, D., & Broneske, D. (2023). Automated occupation coding with hierarchical features: A data-centric approach to classification with pre-trained language models. Discover Artificial Intelligence, 3(1), 6. https://doi.org/q6rn

Simson, J., Kononykhina, O., & Schierholz, M. (2023). occupationMeasurement: A Comprehensive Toolbox for Interactive Occupation Coding in Surveys. Journal of Open Source Software, 8(88), 5505. https://doi.org/q6rs

Sturgis, P., Robinson, T., Fung, L., & Roberts, C. (2025). SOCbot: Using Large Language Models to measure and classify occupations in surveys. OSF. https://doi.org/10.31235/osf.io/9ruwe_v1

Zhang, X., Zhang, Y., Long, D., Xie, W., Dai, Z., Tang, J., Lin, H., Yang, B., Xie, P., Huang, F., & others. (2024). mGTE: Generalized Long-Context Text Representation and Reranking Models for Multilingual Text Retrieval. Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track, 1393–1412.

Article in Press