A Comparative Study of BERT, LLM-Based, and LLM2Vec Embeddings for Text Classification

Matheus Yasuo Ribeiro Utino[1], **Ricardo Marcondes Marcacini**[1]

[1]Institute of Mathematics and Computer Sciences, University of São Paulo (USP)

matheusutino@usp.br, ricardo.marcacini@icmc.usp.br

Abstract. *This work presents a systematic comparison of BERT, raw hidden-layer embeddings from decoder-only Large Language Models (LLMs), and LLM2Vec for text classification across 11 datasets. The evaluation uses K-Nearest Neighbors as the downstream classifier, with hyperparameters tuned through Bayesian optimization. The results show that LLM2Vec achieves the best overall performance, while LLM2Vec and BERT form a statistically comparable performance tier. In contrast, raw LLM embeddings present lower and more variable performance, with only LLM2Vec showing a statistically significant advantage over them. Within LLM2Vec, supervised variants consistently outperform unsupervised ones, highlighting the importance of supervised contrastive learning. Regarding prompt engineering, raw LLM embeddings benefit from task-aligned instructions, whereas LLM2Vec representations remain robust to prompt variation. These findings indicate that encoder-oriented architectures, including BERT and adapted LLM2Vec models, provide more reliable representations for text classification than raw hidden-layer embeddings from decoder-only LLMs.*

1. Introduction

Text classification is one of the most fundamental tasks in Natural Language Processing (NLP), supporting applications such as fake news detection [Roumeliotis et al. 2025], sentiment analysis [Sharma et al. 2025], spam detection [Shen et al. 2025], and mental health [Utino et al. 2026]. Its effectiveness depends not only on the choice of classifier, but also on how textual instances are represented in a computational space. In this context, text embeddings play a central role by mapping documents into dense vector representations that preserve semantic information and allow classification models to exploit similarities among texts [da Costa et al. 2023].

The landscape of text representation methods has evolved substantially over time, from static dense vector approaches such as Word2Vec and GloVe to contextual representations introduced by Transformer-based models such as BERT. While static embeddings assign fixed representations to words independently of context, contextual models generate representations conditioned on the surrounding text, enabling richer semantic and syntactic modeling [da Costa et al. 2023]. This evolution has made the systematic evaluation of embedding approaches increasingly relevant for text classification, particularly because different representation strategies may lead to substantially different downstream performance.

More recently, Large Language Models (LLMs) have emerged as the state of the art across a broad range of NLP tasks and other complex tasks, including mathematics,

physics, and code [Wang et al. 2026, Barman et al. 2025, Jiang et al. 2026]. Their extensive pre-training enables them to produce rich, contextually nuanced representations [Tennenholtz et al. 2023], and their rapid advancement has drawn considerable attention from both academia and industry [Zhao et al. 2025]. Methods such as LLM2Vec further push this frontier by combining architectural elements of BERT and LLMs to yield higher-quality embeddings [Henry et al. 2024]. Despite this progress, selecting the most appropriate embedding strategy remains non-trivial [Muennighoff et al. 2023]. Each approach involves trade-offs among computational cost, semantic richness, and generalization capacity, and these trade-offs can directly impact performance in applications such as text classification, semantic search, and sentiment analysis.

This study addresses this gap through a systematic comparison of BERT, embeddings extracted from hidden layers of LLMs, and the LLM2Vec method for text classification. The main contributions are:

- A comprehensive comparison of BERT, LLM hidden-layer embeddings, and LLM2Vec across datasets from multiple domains (news, social media, and technical documents), evaluating their effectiveness in capturing semantic information for text classification.
- An analysis of the impact of prompt engineering on embedding quality, examining whether statistically significant differences arise from varying the prompt used during embedding generation.
- The experimental code is publicly available on GitHub[1], facilitating reproducibility, extension, and application of the proposed methods to new datasets.

2. Related Work

Embedding models for text classification have been extensively studied, with early works comparing static representations such as Word2Vec, GloVe, and FastText against contextual models like ELMo and BERT across standard NLP benchmarks [Wang et al. 2020, da Costa et al. 2023]. [Nair et al. 2023] extended this line of work to large-scale news classification using Spark NLP, combining TF-IDF, GloVe, BERT, and ELMo with logistic regression and neural classifiers, but remained limited to a single domain and traditional embedding techniques. More comprehensive evaluations have since been conducted across broader task and dataset collections. [Reusens et al. 2024] evaluated 11 model architectures on 20 datasets spanning fake news detection, topic classification, emotion and polarity detection, and sarcasm detection, comparing statistical methods with deep learning approaches including RoBERTa. These studies consistently demonstrate the advantages of contextual representations, though they do not consider more recent approaches based on LLMs.

[Tao et al. 2024] provided an overview of LLM-generated embeddings benchmarked on MTEB, highlighting their effectiveness across multiple domains, though the LLM2Vec adaptation was not addressed. [Buckmann and Hill 2024] proposed combining embeddings from small local LLMs such as LLaMA2-7B with penalized logistic regression, achieving results competitive with GPT-4 on 10 datasets using as few as 60 to 75 labeled samples per class, but neither of these works incorporates LLM2Vec. [Henry et al. 2024] assessed BERT and LLM2Vec for sentiment classification on a single

[1] https://github.com/Matheusutino/bert-llm-llm2vec-text-classification

dataset using SVM and neural classifiers, but did not include raw LLM embeddings as a baseline. [Utino and Marcacini 2025] investigated LLM2Vec as an embedding strategy for nearest-neighbor-based semantic retrieval, showing competitive performance against BERT-based representations and stable behavior under prompt variation. However, the role of raw hidden-layer embeddings from decoder-only LLMs remains less explored in controlled classification settings.

Thus, the comparative behavior of encoder-based embedding models, raw LLM representations, and LLM2Vec remains insufficiently explored under a common text-classification protocol. This work addresses this gap by systematically evaluating these three representation strategies across 11 datasets from diverse domains.

3. Methodology

This section describes the problem formulation, datasets, embedding models, prompts, and experimental setup used in our study.

3.1. Problem Formulation

Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ denote a labeled text classification dataset, where $x_i \in \mathcal{X}$ is a textual document and $y_i \in \mathcal{Y} = \{c_1, c_2, \ldots, c_L\}$ is its class label among L categories.

Given an embedding model $m \in \mathcal{M}$, where $\mathcal{M}$ comprises BERT-based, raw LLM-based, and LLM2Vec-based models, each instance x_i is mapped to a dense vector:

$$\mathbf{z}_i^{(m,p)} = \phi_{m,p}(x_i), \tag{1}$$

where $\phi_{m,p} : \mathcal{X} \rightarrow \mathbb{R}^{d_m}$ is the embedding function for model m, d_m is the embedding dimensionality, and p denotes the prompt configuration. BERT-based models require no prompt, while LLM and LLM2Vec models are evaluated under different prompt strategies to assess their influence on representation quality.

A classifier h_θ is then trained on the embedded representations $\{(\mathbf{z}_i^{(m,p)}, y_i)\}_{i=1}^{N}$ to predict the label of unseen instances. The objective is to identify the embedding configuration (m, p) that yields the most discriminative representation for classification.

3.2. Datasets

For a comprehensive and robust analysis, we utilized 11 datasets to compare the embedding methods. These datasets vary significantly in terms of their characteristics, such as the number of instances, the length of the text (measured in characters), the number of classes, and the complexity of their respective applications. This variety allows for a thorough examination of the embedding techniques across different domains and contexts, providing valuable insights into their effectiveness and scalability. Table 1 summarizes the key features of each dataset, including the number of instances, the minimum, maximum, and median text lengths (in characters), and the number of labels present in each dataset. These datasets span a range of applications, from sentiment analysis to topic categorization, offering a diverse set of challenges for evaluating the embedding methods.

Table 1. Summary of the datasets used in the experiments, including the number of instances, text length statistics (in characters), and the number of class labels.

Dataset	Instances	Min Length	Max Length	Median Length	Labels
CSTR	299	150	2807	1078	4
SyskillWebert	333	20	55595	1372	4
Review Polarity	2000	90	14898	3592	2
Dmoz Science	6000	20	506	145	12
Dmoz Health	6500	23	489	149	13
Classic4	7095	4	4294	646	4
Re8	7674	29	5294	362	8
Industry Sector	8817	1	159727	1196	11
Dmoz Computers	9500	15	692	123	18
NSF	10524	18	159	70	16
Dmoz Sports	13500	23	410	121	27

3.3. Embedding Models

BERT is an encoder-based language model built upon the Transformer architecture and represents a major milestone in Natural Language Processing (NLP). Its main characteristic is the use of bidirectional contextualization, allowing each token representation to incorporate information from both its left and right contexts. This design enables the model to capture semantic and syntactic dependencies more effectively than strictly unidirectional architectures. During pre-training, BERT combines Masked Language Modeling (MLM), in which randomly masked tokens must be predicted from their surrounding context, with Next Sentence Prediction (NSP), which aims to determine whether two input sentences appear consecutively in the original text. These objectives allow BERT to learn contextualized representations that are highly effective for a wide range of downstream NLP tasks [Devlin et al. 2019].

Large Language Models (LLMs) are neural language models characterized by large parameter counts, broad training corpora, and strong generalization capabilities across multiple domains and tasks [Zhao et al. 2026]. Although they have gained particular prominence in text generation, their internal hidden states encode rich semantic information about the input text. Many contemporary LLMs adopt decoder-only architectures, which are primarily optimized for autoregressive generation rather than direct embedding extraction. Nevertheless, the representations produced by their intermediate or final hidden layers can be adapted and leveraged as dense text embeddings, making LLMs increasingly relevant for representation learning.

LLM2Vec is an approach specifically designed to adapt decoder-only LLMs for embedding generation. The method modifies the conventional causal attention mechanism to enable bidirectional contextualization, allowing the model to represent each token using information from the entire input sequence. In addition, LLM2Vec introduces Masked Next Token Prediction (MNTP), a training objective that adapts the generative capabilities of LLMs to representation learning. The method also employs unsupervised contrastive learning, in which the same sentence is processed under different dropout masks, and the resulting representations are encouraged to be similar while remaining

Table 2. Summary of the models, detailing the embedding method, context window size, and output dimensions.

Method	Model	Context Window	Output Size
BERT	All-distilroBERTa-v1	128	768
	All-MiniLM-L12-v2	256	384
	All-MiniLM-L6-v2	256	384
	All-mpnet-base-v2	384	768
LLM	Gemma2-9B	8192	3584
	Qwen2.5-7B	32768	3584
	Mistral-7B	32768	4096
	LLaMA-3.2-3B	131072	3072
LLM2Vec	Sheared-LLaMA (unsup-SimCSE)	4096	2048
	Sheared-LLaMA (supervised)	4096	2048
	Llama-3-8B (unsup-SimCSE)	8192	4096
	Llama-3-8B (supervised)	8192	4096
	Mistral-7B (unsup-SimCSE)	32768	4096
	Mistral-7B (supervised)	32768	4096

distinct from representations of other sentences in the batch. Beyond the unsupervised stage, LLM2Vec incorporates supervised contrastive learning, where semantically related sentence pairs are brought closer in the embedding space, whereas unrelated pairs, including hard negatives and in-batch negatives, are pushed apart. This combination allows LLM2Vec to exploit the expressive capacity of LLMs while tailoring them to embedding-oriented tasks, achieving state-of-the-art results on MTEB among models trained exclusively with publicly available data [BehnamGhader et al. 2024].

Encoder-based models such as BERT are highly effective at producing compact and semantically meaningful representations, often with favorable computational efficiency. However, their relatively limited context windows may constrain their applicability to longer documents [Gao et al. 2021, Ding et al. 2020]. In contrast, LLM-based representations benefit from broader pre-training, larger context windows, and higher-dimensional hidden states, which may provide richer semantic representations for complex or long-form texts [Zhao et al. 2026]. Thus, the models considered in this study cover different architectural paradigms and representation capacities, enabling a comparative analysis of their effectiveness for embedding generation. Table 2 summarizes the models employed, including their context windows and output dimensionalities.

3.4. Prompts

LLM-based models exhibit a significant dependence on the chosen prompt [Zhao et al. 2026], which influences the nuances and details of the generated text. To assess the sensitivity of the LLM and LLM2Vec methods to prompt variations, three types of prompts were selected: Base Prompt (BP), Instruction Summary Prompt (ISP), and Instruction Classification Prompt (ICP), as detailed in Figure 1. The BP consists solely of the system prompt, without providing explicit instructions on the required task. This represents the simplest version and serves as a baseline. The ISP instructs the model to summarize the text and extract its key points, potentially enhancing the efficiency of data

classification. In contrast, the ICP explicitly directs the LLM to classify the text within predefined categories, providing prior knowledge of the possible classes. In this case, the model is expected to adjust its embeddings to better align with the most relevant class.

System Prompt

"You are an intelligent assistant that follows the user's instructions closely. Respond to the user's queries in a clear, concise, and helpful manner. Ensure that all answers are accurate, relevant, and tailored to the user's needs. Stay focused on providing valuable information and completing tasks as requested by the user."

Base Prompt

" "

Instruction Summary Prompt

"Please summarize the given text by identifying and highlighting the main points, key arguments, and any significant details. Provide a concise summary that captures the essential information and overall message of the text."

Instruction Classification Prompt

"Please classify the given text into one of the following categories: {classes}. Analyze the content carefully and determine which category best represents the overall topic and message of the text."

Figure 1. Prompts used to evaluate the sensitivity of generative models.

3.5. Experimental Setup

The embeddings generated by each representation model are used as input features for a K-Nearest Neighbors (KNN) classifier, providing a direct evaluation of how well the embedding space organizes semantically related instances. Performance is estimated using stratified 5-fold cross-validation and macro-averaged F1-score (F1-macro). KNN hyperparameters are selected through 20 iterations of Bayesian optimization [Akiba et al. 2019], considering the number of neighbors ($k \in [3, 50]$), distance metric (Euclidean, Manhattan, or cosine), and weighting strategy (uniform or distance). Inputs exceeding the maximum context length supported by each model were truncated. Regarding embedding extraction, encoder-based models use the pooling and normalization strategies provided by their respective Sentence Transformer checkpoints, with mean pooling over the final hidden states and padding excluded through the attention mask. For raw LLMs, prompted inputs are converted into L2-normalized embeddings. For LLM2Vec, embeddings are extracted from the final hidden layer using mean pooling, excluding instruction and padding tokens.

4. Results

Figure 2 presents the distribution of F1-scores obtained by the KNN classifier across the 11 datasets, comparing the three prompt configurations (Base, Instruction Classification, and Instruction Summary) for both the LLM and LLM2Vec embedding models.

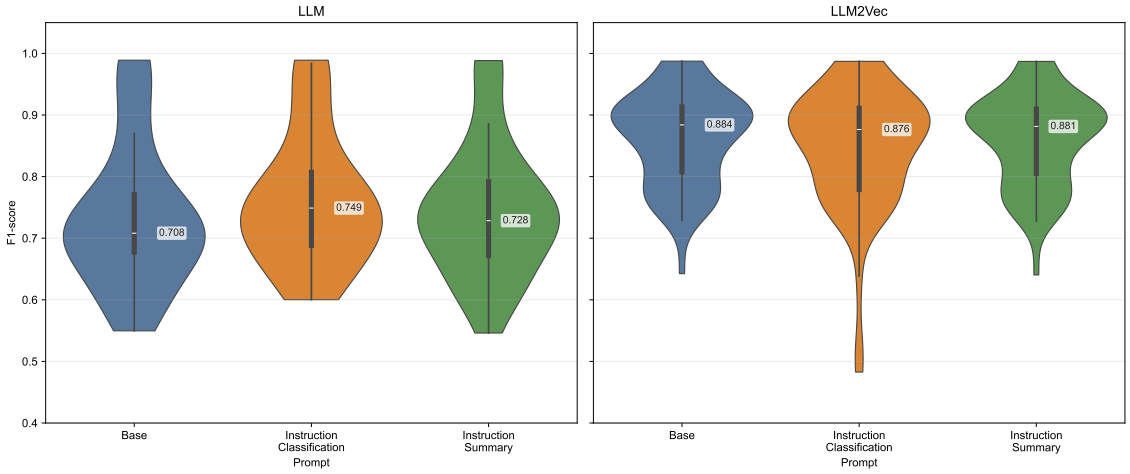


Figure 2. Distribution of F1-scores obtained by KNN across 11 datasets for LLM and LLM2Vec embeddings under three prompt configurations: Base (BP), Instruction Classification (ICP), and Instruction Summary (ISP).

A first observation concerns the substantial performance gap between the two model families. LLM2Vec consistently achieves higher median F1-scores across all prompt configurations, with values ranging from 0.876 to 0.884, while LLM embeddings yield noticeably lower medians, between 0.708 and 0.749. Depending on the prompt configuration, LLM2Vec outperforms LLM by 16.9% to 24.9% relative to the corresponding LLM baseline, indicating that the bidirectional contextualization mechanism and contrastive training objectives of LLM2Vec produce representations that are considerably more discriminative for downstream classification.

Regarding sensitivity to prompt engineering, the two model families exhibit contrasting behaviors. For LLM embeddings, the Instruction Classification Prompt (ICP) yields the highest median F1-score (0.749), representing a relative improvement of 5.8% over the Base Prompt (0.708), while the Instruction Summary Prompt (ISP) achieves an intermediate gain of 2.8%. This pattern suggests that task-specific instructions provide meaningful guidance for shaping the hidden-state representations of decoder-only LLMs, particularly when the instruction is aligned with the target task. For LLM2Vec, however, the three configurations produce nearly equivalent median performance, with ICP and ISP deviating by only -0.9% and -0.3% relative to the Base configuration (0.884), indicating that the model's training already yields robust representations largely invariant to prompt choice.

A notable finding concerns the variance induced by ICP in the LLM2Vec setting. While its median F1-score (0.876) remains competitive, the corresponding violin exhibits a pronounced lower tail, with performance dropping to approximately 0.48 on certain datasets. This indicates that, although ICP does not degrade average performance for LLM2Vec, it introduces instability across datasets, potentially because explicitly em-

bedding class label information into the prompt may conflict with the model's internally learned representations on datasets with specific structural characteristics. In contrast, both the Base and ISP configurations produce compact, high-concentration distributions, reflecting consistent performance across the benchmark. Taken together, these results indicate that prompt engineering has a more meaningful and reliable impact on standard LLM embeddings than on LLM2Vec, where it offers limited benefit and may, in specific configurations, compromise robustness.

To assess whether the observed differences across prompt configurations are statistically significant, a Friedman test followed by a Nemenyi post-hoc test was conducted over the 11 datasets. Figure 3 presents the Critical Difference (CD) diagram for the LLM model. The Instruction Classification Prompt achieves the best average rank (1.455), followed by the Instruction Summary Prompt (1.909) and the Base Prompt (2.636). Accordingly, ICP is statistically superior to the BP, whereas the differences between ICP and ISP, and between ISP and Base, do not reach statistical significance. This result confirms that providing explicit task-aligned instructions improves the quality of LLM-based representations in a statistically reliable manner, even though intermediate prompt strategies (ISP) do not yield a conclusive advantage over either extreme. For LLM2Vec, the Friedman test found no statistically significant difference across prompt configurations, which is consistent with the near-identical median F1-scores observed in Figure 2 and further corroborates the robustness of LLM2Vec representations to prompt variation.

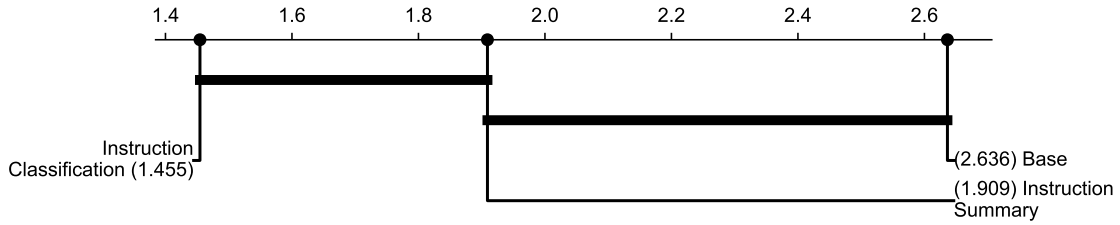
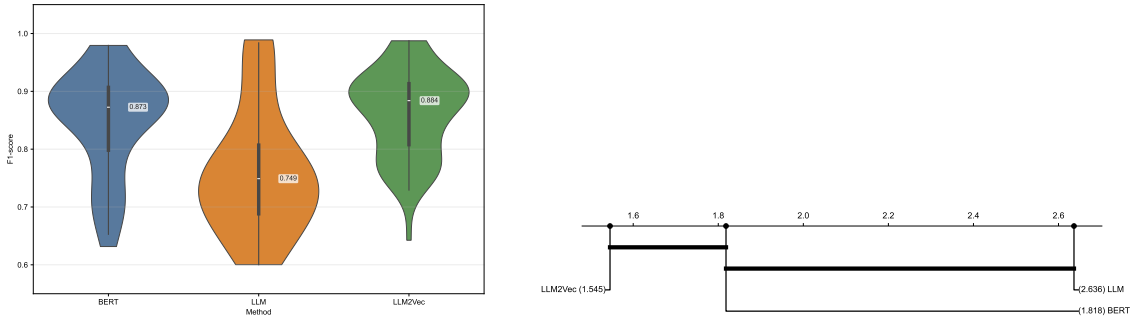


Figure 3. Critical Difference diagram for the Nemenyi post-hoc test comparing prompt configurations for LLM embeddings with KNN classification. Average ranks are shown in parentheses. Methods connected by a horizontal bar are not significantly different at the 0.05 significance level.

Figure 4a compares the F1-score distributions obtained by KNN using the best-performing prompt configuration for each generative model, namely the Instruction Classification Prompt for LLM and the Base Prompt for LLM2Vec, against BERT-based embeddings across the 11 datasets. LLM2Vec achieves the highest median F1-score (0.884), closely followed by BERT (0.873), representing a marginal relative improvement of 1.3% over BERT. Both models substantially outperform LLM embeddings (0.749), with BERT and LLM2Vec yielding gains of 16.6% and 18.0%, respectively, relative to the LLM baseline. Beyond median performance, the distribution shapes reveal additional differences. BERT and LLM2Vec exhibit compact violins with mass highly concentrated in a narrow band above 0.8, reflecting stable and consistent performance across datasets. LLM, by contrast, presents a considerably wider distribution spanning from approximately 0.6 to 1.0, indicating that its representations yield highly variable classification quality depending on the dataset. These results suggest that while LLM hidden-layer representations can be improved through prompt engineering, they remain insufficient to match the classifi-

cation quality of encoder-oriented architectures, whether classical bidirectional models such as BERT or hybrid approaches such as LLM2Vec.

Figure 4b presents the Critical Difference diagram comparing BERT, LLM, and LLM2Vec under their best-performing prompt configurations. LLM2Vec achieves the best average rank (1.545), followed by BERT (1.818) and LLM (2.636). The only statistically significant difference is between LLM2Vec and LLM, while the pairs LLM2Vec–BERT and BERT–LLM do not differ significantly. This result indicates that LLM2Vec and BERT occupied a statistically comparable performance tier, both superior in ranking to LLM, though only the gap between LLM2Vec and LLM is confirmed as significant. Taken together with Figure 4a, these findings reinforce that encoder-oriented architectures, whether classical as BERT or hybrid as LLM2Vec, consistently produce more discriminative representations for text classification than raw hidden-layer embeddings from decoder-only LLMs.



(a) F1-score distributions. **(b) Critical Difference diagram.**

Figure 4. Comparison of BERT, LLM, and LLM2Vec embeddings under their best-performing prompt configurations. The violin plot reports the F1-score distributions obtained by KNN across 11 datasets, while the Critical Difference diagram reports the average ranks and statistically significant differences among the evaluated representation families.

Table 3 reports the average F1-score of each individual model across the 11 datasets under the best-performing prompt configuration. Within the LLM2Vec category, the supervised variants consistently outperform their unsupervised counterparts. The two top supervised models, Llama-3-8B (supervised) and Mistral-7B (supervised), tie at a mean F1-score of 0.886. Comparing each against its unsupervised counterpart, Mistral-7B (supervised) yields a relative gain of 2.8% over Mistral-7B (unsup-SimCSE) (0.862), while Llama-3-8B (supervised) yields a gain of 3.9% over Llama-3-8B (unsup-SimCSE) (0.853). The gap widens to 7.4% when comparing the weakest supervised model against its unsupervised equivalent. This pattern confirms that the supervised contrastive learning stage is a meaningful contributor to representation quality in LLM2Vec.Within the BERT category, all four models cluster in a narrow range from 0.839 to 0.863, a spread of only 2.8%, indicating that the evaluated sentence-transformer variants are largely comparable in this setting. Notably, the strongest BERT model (All-mpnet-base-v2, 0.863) is competitive with the weakest supervised LLM2Vec variant (Sheared-LLaMA (supervised), 0.874). LLM models occupy a clearly distinct and inferior performance tier, ranging from 0.735 (LLaMA-3.2-3B) to 0.774 (Gemma2-9B), a spread of 5.4%. The relatively larger variance within this group suggests that model scale and architecture play a more decisive

role for raw hidden-layer embeddings than for encoder-oriented approaches, where even smaller models such as All-MiniLM-L6-v2 achieve competitive results.

Table 3. Average F1-score of each model across 11 datasets under the best-performing prompt configuration. The best model per category is underlined and the overall best model is shown in bold.

Category	Model	Mean F1-score
BERT	All-mpnet-base-v2	0.863
	All-MiniLM-L12-v2	0.850
	All-MiniLM-L6-v2	0.845
	All-distilroberta-v1	0.839
LLM	Gemma2-9B	0.774
	Qwen2.5-7B	0.769
	Mistral-7B	0.764
	LLaMA-3.2-3B	0.735
LLM2Vec	Sheared-LLaMA (unsup-SimCSE)	0.813
	Sheared-LLaMA (supervised)	0.874
	Llama-3-8B (unsup-SimCSE)	0.853
	Llama-3-8B (supervised)	**0.886**
	Mistral-7B (unsup-SimCSE)	0.862
	Mistral-7B (supervised)	**0.886**

5. Conclusion

This paper presented a comparative evaluation of BERT-based embeddings, raw hidden-layer representations from LLMs, and LLM2Vec embeddings for text classification. The experiments were conducted across 11 datasets using KNN as the downstream classifier, allowing the analysis to focus on how well each embedding strategy organizes textual instances in the representation space.

The results show that LLM2Vec achieved the best overall performance, with the highest median F1-score and the best average rank. However, LLM2Vec and BERT occupied a statistically comparable performance tier, indicating that traditional encoder-based models remain strong and reliable alternatives for text classification. In contrast, raw LLM embeddings obtained lower median performance and greater variability, with only the difference between LLM2Vec and raw LLM embeddings being statistically significant.

The prompt analysis showed that task-oriented instructions can improve raw LLM embeddings, with the Instruction Classification Prompt outperforming the Base Prompt. For LLM2Vec, prompt variations had limited impact, suggesting that its representation learning process already provides robust embeddings with low prompt sensitivity. Future work should extend this evaluation to additional downstream tasks, multilingual datasets, and analyses of computational cost, including inference time and memory consumption.

Acknowledgments

This study was partially funded by the *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior* (CAPES) – Finance Code 001. We also acknowledge support from the *Fundação de Amparo à Pesquisa do Estado de São Paulo* (FAPESP) under grant numbers 2019/07665-4, 2023/10100-4, 2013/07375-0, 2024/08485-8, 2022/03176-1, and 2023/00488-5. Additional support was provided by the *Conselho Nacional de Desenvolvimento Científico e Tecnológico* (CNPq), under grant number 130041/2025-4.

References

Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2623–2631.

Barman, K. G., Caron, S., Sullivan, E., de Regt, H. W., de Austri, R. R., Boon, M., Färber, M., Fröse, S., Golling, T., Lopez, L. G., et al. (2025). Large physics models: towards a collaborative approach with large language models and foundation models. *The European Physical Journal C*, 85(9):1066.

BehnamGhader, P., Adlakha, V., Mosbach, M., Bahdanau, D., Chapados, N., and Reddy, S. (2024). LLM2vec: Large language models are secretly powerful text encoders. In *First Conference on Language Modeling*.

Buckmann, M. and Hill, E. (2024). Logistic regression makes small llms strong and explainable" tens-of-shot" classifiers. *arXiv preprint arXiv:2408.03414*.

da Costa, L. S., Oliveira, I. L., and Fileto, R. (2023). Text classification using embeddings: a survey. *Knowledge and Information Systems*, 65(7):2761–2803.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Burstein, J., Doran, C., and Solorio, T., editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ding, M., Zhou, C., Yang, H., and Tang, J. (2020). Cogltx: Applying bert to long texts. *Advances in Neural Information Processing Systems*, 33:12792–12804.

Gao, S., Alawad, M., Young, M. T., Gounley, J., Schaefferkoetter, N., Yoon, H. J., Wu, X.-C., Durbin, E. B., Doherty, J., Stroup, A., et al. (2021). Limitations of transformers on clinical text classification. *IEEE journal of biomedical and health informatics*, 25(9):3596–3607.

Henry, M. M., Hervanto, N. A., Isnan, M., Kurnianingrum, D., Ratnapuri, C. I., and Pardamean, B. (2024). Llm2vec sentence embeddings analysis in sentiment classification. In *2024 IEEE International Conference on Data and Software Engineering (ICoDSE)*, pages 161–165. IEEE.

Jiang, J., Wang, F., Shen, J., Kim, S., and Kim, S. (2026). A survey on large language models for code generation. *ACM Trans. Softw. Eng. Methodol.*, 35(2).

Muennighoff, N., Tazi, N., Magne, L., and Reimers, N. (2023). MTEB: Massive text embedding benchmark. In Vlachos, A. and Augenstein, I., editors, *Proceedings of the 17th Conference*

of the European Chapter of the Association for Computational Linguistics, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.

Nair, A. R., M, S., and Gupta, D. (2023). Comparative analysis of word embeddings for text classification in spark nlp. In *2023 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*, pages 130–136.

Reusens, M., Stevens, A., Tonglet, J., De Smedt, J., Verbeke, W., vanden Broucke, S., and Baesens, B. (2024). Evaluating text classification: A benchmark study. *Expert Systems with Applications*, 254:124302.

Roumeliotis, K. I., Tselikas, N. D., and Nasiopoulos, D. K. (2025). Fake news detection and classification: A comparative study of convolutional neural networks, large language models, and natural language processing models. *Future Internet*, 17(1).

Sharma, N. A., Ali, A. S., and Kabir, M. A. (2025). A review of sentiment analysis: tasks, applications, and deep learning techniques. *International journal of data science and analytics*, 19(3):351–388.

Shen, L., Wang, Y., Li, Z., and Ma, W. (2025). Sms spam detection using bert and multi-graph convolutional networks. *International journal of intelligent networks*, 6:79–88.

Tao, C., Shen, T., Gao, S., Zhang, J., Li, Z., Tao, Z., and Ma, S. (2024). Llms are also effective embedding models: An in-depth overview. *arXiv preprint arXiv:2412.12591*.

Tennenholtz, G., Chow, Y., Hsu, C.-W., Jeong, J., Shani, L., Tulepbergenov, A., Ramachandran, D., Mladenov, M., and Boutilier, C. (2023). Demystifying embedding spaces using large language models. *arXiv preprint arXiv:2310.04475*.

Utino, M. and Marcacini, R. (2025). Bert vs. llm2vec: A comparative study of embedding models for semantic information retrieval. In *Anais do XXII Encontro Nacional de Inteligência Artificial e Computacional*, pages 427–438, Porto Alegre, RS, Brasil. SBC.

Utino, M. Y. R., Matsushima, E. H., Paes, A., and Mann, P. (2026). Synthetic data for mental health: A comparative analysis of llms, bert, and copy-based augmentation. In de Freitas, R. and Furtado, D., editors, *Intelligent Systems*, pages 393–408, Cham. Springer Nature Switzerland.

Wang, C., Nulty, P., and Lillis, D. (2020). A comparative study on word embeddings in deep learning for text classification. In *Proceedings of the 4th international conference on natural language processing and information retrieval*, pages 37–46.

Wang, P.-Y., Liu, T.-S., Wang, C., Li, Z., Wang, Y., Yan, S., Jia, C., Liu, X.-H., Chen, X., Xu, J., and Yu, Y. (2026). A survey on large language models for mathematical reasoning. *ACM Comput. Surv.*, 58(8).

Zhao, W., Yang, X., Lyu, Z., Xu, C., and Guan, Z. (2025). Road of large language model: Source, challenge, and future perspectives. *Research*, 8:0655.

Zhao, W. X., Zhou, K., Li, J., Tang, T., Dong, Z., Hou, Y., Zhang, B., Min, Y., Zhang, J., Liu, P., et al. (2026). A survey of large language models. *Frontiers of Computer Science*, 20(12):2012627.