Self-Organizing Maps for Lightweight Prompt Adaptation

Samuel Henrique Silva[1], Allan Miller Silva Lima[1], Denis Mayr Lima Martins[2], Fernando Buarque de Lima Neto[1]

[1]Departamento de Engenharia da Computação – Universidade de Pernambuco
Recife – PE – Brasil

[2]Departamento de Computação e Matemática – Universidade de São Paulo
Ribeirão Preto – SP – Brasil.

{shss,amsl,fbln}@ecomp.poli.br, martins.denis@usp.br

Abstract. *We present Self-Organizing Prompt Maps[1] (SOPM), an intelligent system that unites topological clustering with generative capabilities to enable in-context, parameter-free adaptation of Large Language Models (LLMs) without fine-tuning. By organizing historical prompts onto a 2D topological grid via a Self-Organizing Map (SOM), SOPM establishes a continuous semantic space where context-aware retrieval is performed via neighborhood search, ensuring spatial transparency for human-in-the-loop validation. Experimental results demonstrate that integrating SOM-based prompt retrieval with generative architectures enhances downstream task accuracy. SOPM offers a practical pathway toward interpretable and efficient LLM prompt adaptation.*

Resumo. *Apresentamos Self-Organizing Prompt Maps[1] (SOPM), um sistema inteligente que une agrupamento topológico com capacidades generativas para permitir adaptação de prompts sem ajuste de parâmetros em Grandes Modelos de Linguagem (LLMs). Ao organizar prompts históricos em uma grade topológica 2D via Self-Organizing Map (SOM), o SOPM cria um espaço semântico contínuo onde a adaptação de prompts ocorre via busca por vizinhança, garantindo transparência espacial e validação humana. Resultados demonstram que integrar recuperação de prompts via SOM com arquiteturas generativas melhora acurácia em tarefas subsequentes. O SOPM oferece um caminho prático para ajuste de prompts de LLM interpretável e eficiente.*

1. Introduction

Large Language Models (LLMs) have demonstrated remarkable versatility across diverse domains, yet their deployment is increasingly bottlenecked by the complexity of prompt engineering. As models scale in parameter count and capability, the manual curation of static prompt templates or the reliance on brittle few-shot examples fails to generalize effectively under distributional shifts or user-specific constraints [Arora et al. 2023]. While Parameter-Efficient Fine-Tuning (PEFT) methods like Low-Rank Adaptation (LoRA) [Han et al. 2024] offer a path toward customization, they often incur significant computational overhead and limited interpretability. Consequently,

[1]Demonstration video: https://youtu.be/J2ds1wWBUJk

there is an urgent need for a lightweight, interpretable mechanism that bridges the gap between semantic understanding and structured prompt management.

To alleviate this issue, we introduce Self-Organizing Prompt Maps (SOPM), a novel framework that leverages the competitive learning and neighborhood preservation properties of the Self-Organizing Map (SOM) to construct a latent grid of prompt prototypes. SOPM maintains a topological ordering of prompts, allowing for interpolation between strategies and robust retrieval. We hypothesize that introducing a topology-preserving structure (specifically via SOM) can map prompt strategies into a low-dimensional manifold where neighborhood relationships correspond to functional similarities in downstream tasks.

The core contribution lies in coupling LLM semantic embeddings with SOM weight updates: (1) *Topological Adaptation*, where the map evolves to capture context-specific variations without modifying the base model weights; and (2) *Interpretability*, where a SOM-based visualization provides a map of the prompt space. The source code is publicly available at `https://github.com/samhenrique/sopm-interface`.

2. Related Work

Prompt engineering and tuning [Mao et al. 2025] are central in LLM-based applications, where simple prompt templates often fail to meet user-specific preferences. Prompt engineering [Li et al. 2025] and Retrieval Augmented Generation [Yazan et al. 2025] offer mechanisms for incorporating additional information into user prompts for effective prompt adaptation and personalization. However, these approaches typically require manual prompt crafting or rely on task-specific techniques for extracting curated examples and instructions, limiting scalability across diverse tasks.

Self-Organizing Maps [Kohonen 1990] have been extensively employed in clustering applications and data visualization. Recent advancements, such as DP-SOM [Manduchi et al. 2019] and SOM-CPC [Huijben et al. 2023], offer potential for scalable, adaptive clustering of high-dimensional embeddings. However, in the context of natural language processing, their integration with LLMs remains underexplored.

SOPM addresses the issues of traditional approaches by SOM's ability to preserve semantic relationships in the data, which enables efficient retrieval of semantically similar prompts so that prompt traits can be leveraged for prompt adaptation. SOPM goes beyond visualization by embedding SOM directly into the prompt adaptation pipeline, *without requiring task-specific signals nor fine-tuning*.

3. Technical Overview

The inner data processing pipeline of SOPM is divided into two sequential phases, as depicted in Figure 1. In the offline phase, historical prompts from the database are preprocessed by a lightweight embedding model to generate semantic vector representations (i.e., embeddings), which are then organized by the SOM-based retrieval module. In the online phase, a user query prompt is processed by the same embedding model used in the offline phase and matched through the SOM, yielding retrieved prompts that are subsequently leveraged as contextual examples to produce the final tuned prompt.

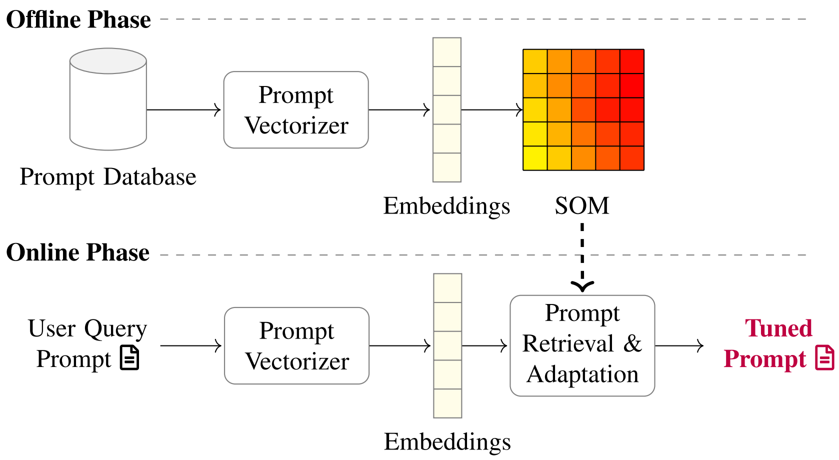

Figure 1. System architecture overview. The offline phase consists of the generation of prompt embeddings and clustering via SOM. The online phase processes a user query prompt and perform SOM-informed adaptation.

3.1. Offline phase

For each prompt p in a prompt database, we leverage the `all-MiniLM-L6-v2` [Wang et al. 2020] embedding model to generate a high-dimensional vector representation $\mathbf{z}_p \in \mathbb{R}^{384}$. Embeddings are L2-normalized to ensure unit vectors for similarity computation.

Given a prompt embedding dataset $\mathcal{Z} = \{\mathbf{z}_i \in \mathbb{R}^d\}_{i=1}^N$ and a set of SOM prototype vectors (neurons) $\{\mathbf{w}_j \in \mathbb{R}^d\}_{j=1}^K$ arranged on a fixed lattice, the SOM is trained via competitive learning with neighborhood adaptation as follows (see Figure 2 for an illustrative example). For each $\mathbf{z} \in \mathcal{Z}$, the Best Matching Unit (BMU) is identified as $c = \arg\min_j \|\mathbf{z} - \mathbf{w}_j\|_2$. A neighborhood function centered at the BMU is defined as $h_{cj}(t) = \exp\left(-\|\mathbf{r}_c - \mathbf{r}_j\|^2 / 2\sigma(t)^2\right)$, where $\mathbf{r}_j$ denotes the lattice position of neuron j and $\sigma(t)$ is a decreasing neighborhood radius. The prototype vectors are then updated according to $\mathbf{w}_j(t+1) = \mathbf{w}_j(t) + \eta(t)\, h_{cj}(t)\, (\mathbf{z} - \mathbf{w}_j(t))$, where $\eta(t)$ is a learning rate that, along with $\sigma(t)$ decays over time. This process is repeated for $t = 1, \ldots, T$ epochs. The training procedure yields a topology-preserving mapping that balances global ordering and local quantization accuracy. In this sense, SOPM leverages SOM as a clustering mechanism that groups semantically similar prompts in neighboring lattice units.

We optimized SOM hyperparameters via grid search. The best configuration found was a 5×10 rectangular map, $\eta(0) = 0.1$ and $\sigma(0) = 2$ that achieved quantization error of ≈ 0.637 and topographic error of ≈ 0, frequently used metrics in the literature [Kohonen et al. 1996].

3.2. Online phase

Given a user query prompt q, we apply the same embedding model to generate a vector representation $\mathbf{z}_q$ and feed it to the SOM to find its BMU. The prompt retrieval and adaptation module leverages the SOM structure to search for sets of positive and prompts centered at the query prompt BMU.

Based on those, SOPM constructs a meta-prompt that instructs an LLM (here, Qwen3-8B [Yang et al. 2025]) to generate a tuned prompt variant. The meta-prompt in-

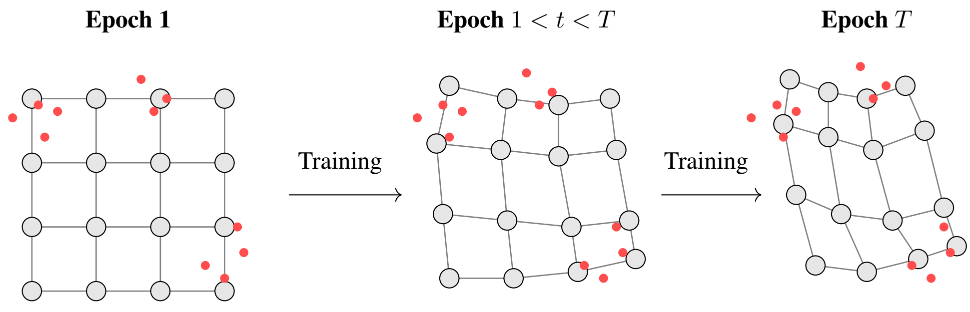

Figure 2. SOM lattice during training. From left to right, the lattice progresses from its initial state to an arrangement aligned with the data distribution.

cludes: (a) The original query prompt; (b) Positive examples to demonstrate successful patterns; (c) Negative examples to highlight contrasting formulations; and (d) Instructions to maintain task consistency while incorporating effective patterns.

SOM-based adaptation supports prompt reuse, variation generation, and robust zero-shot applications.

3.3. System Setup

SOPM was implemented as an interactive web application using Streamlit. The `all-MiniLM-L6-v2` embedding model [Wang et al. 2020] occupies approximately 90.9 MB on disk (22.7M parameters) and maps each prompt to a 384-dimensional `float32` vector (1.5 KB per vector). Prompt embeddings are persisted in a lightweight SQLite database with a key-value upsert cache to avoid redundant re-encoding. For the 100-prompt corpus used in our experiments, the embedding database occupies ≈180 KB. The trained SOM weight matrix ($5 \times 10 \times 384$) requires ≈75 KB; the full serialized artifact (including weights, the prompt corpus, and the embedding matrix) totals ≈270 KB. The LLM component is served locally via an Ollama endpoint using `Qwen3-8B` [Yang et al. 2025].

The interface is structured into two panels (see Figure 3). The left panel implements the main workflow: (a) user prompt entry followed by SOM-guided improvement; (b) display of the LLM-generated tuned prompt; (c) tabular inspection of retrieved positive and negative examples with cosine similarity scores; and (d) direct LLM execution with configurable output token budget. The right panel renders an interactive U-Matrix overlaid with the prompt corpus (*i.e.*, retrieved positives in green, negatives in purple, and the query in gold).

4. Experimental Analysis

We conducted an experimental evaluation on the AG News dataset [Zhang et al. 2015], a benchmark for text classification. We derived a training data dataset of 100 prompts with varying formulations, including: (a) "Classify this article: [text]" (direct classification); (b) "What label best describes this article? [text]" (question-first); (c) "Is this about [options]? [text]" (multiple-choice); (d) "Would you recommend this to [persona]? [text]" (recommendation); and (e) "Which newspaper section? [text]" (section assignment).

We employed BLEU [Papineni et al. 2002] and ROUGE-1 [Hu and Zhou 2024] to assess whether SOM-guided recommendations improve baseline LLM response qual-

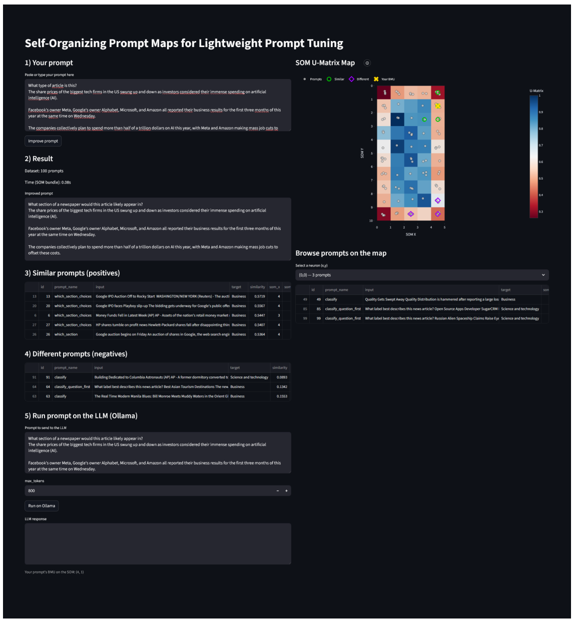

Figure 3. The SOPM interactive interface.

ity. Reported results are aggregated across the 10 executions per class to report mean performance with standard deviations. Overall, `Qwen3` showed an increased BLEU value from 0.0056 to 0.0164 and ROUGE-1 from 0.068 to 0.101, aligning with exact-match gains. Overall, by leveraging contrastive prompts via SOM-based retrieval, `Qwen3` demonstrated 61.4% exact-match accuracy (percentage of responses that exactly match the expected category label), outperforming its baseline results.

5. Conclusion

We introduce Self-Organizing Prompt Maps (SOPM), a SOM-based framework that enables interpretable prompt adaptation by combining topological clustering of SOM with LLM generation, without requiring task-specific fine-tuning. Our evaluation demonstrates that while SOPM offers distinct advantages, including transparency through spatial visualization and explicit retrieval control, its effectiveness is model-dependent. This confirms that the success of retrieval-augmented synthesis relies critically on the base model's resilience to input variations, highlighting SOPM as a valuable tool for scenarios prioritizing rapid prototyping and interpretability over pure optimization speed.

Despite these results, the full potential of SOPM faces non-trivial limitations, primarily regarding fixed topology adaptability to evolving semantic distributions and variability in personalization outcomes across different LLM architectures. Future research must focus on adaptive SOM architectures (e.g., Growing Neural Gas) and model-agnostic strategies that balance topological structure with functional relevance.

Acknowledgment

This study was partially financed by the *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil* (CAPES) – Finance Code 001. The authors also acknowledge the support from *Conselho Nacional de Desenvolvimento Científico e Tecnológico* (CNPq) and *Fundação de Amparo à Ciência e Tecnologia do Estado de Pernambuco* (FACEPE).

References

Arora, S., Narayan, A., Chen, M. F., Orr, L., Guha, N., Bhatia, K., Chami, I., and Re, C. (2023). Ask me anything: A simple strategy for prompting language models. In *The Eleventh International Conference on Learning Representations*.

Han, Z., Gao, C., Liu, J., Zhang, J., and Zhang, S. Q. (2024). Parameter-efficient fine-tuning for large models: A comprehensive survey.

Hu, T. and Zhou, X.-H. (2024). Unveiling llm evaluation focused on metrics: Challenges and solutions.

Huijben, I. A., Nijdam, A. A., Overeem, S., Van Gilst, M. M., and Van Sloun, R. (2023). Som-cpc: Unsupervised contrastive learning with self-organizing maps for structured representations of high-rate time series. In *International Conference on Machine Learning*, pages 14132–14152. PMLR.

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480.

Kohonen, T., Schroeder, M. R., and Huang, T. S. (1996). Engineering applications of the self-organizing map. *Proceedings of the IEEE*, 84(10):1358–1384.

Li, W., Wang, X., Li, W., and Jin, B. (2025). A survey of automatic prompt engineering: An optimization perspective.

Manduchi, L., Hüser, M., Vogt, J. E., Rätsch, G., and Horn, M. (2019). Deep probabilistic clustering with self-organizing maps. *arXiv preprint arXiv:1910.01590*.

Mao, W., Wu, J., Chen, W., Gao, C., Wang, X., and He, X. (2025). Reinforced prompt personalization for recommendation with large language models. *ACM Trans. Inf. Syst.*, 43(3).

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., and Zhou, M. (2020). Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33:5776–5788.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. (2025). Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Yazan, M., Verberne, S., and Situmeang, F. (2025). Improving rag for personalization with author features and contrastive examples. In *European Conference on Information Retrieval*, pages 408–416. Springer.

Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-level convolutional networks for text classification. In Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.