Decision Making in Structural Health Monitoring Using Large Language Models

K. SMARSLY, Y. A. CHMELNIZKIJ, K. DRAGOS, J. PERALTA, T. AL-ZURIQAT, M. E. AHMAD, C. CHILLON GECK, P. PERALTA and L. SEITZ

ABSTRACT

Structural health monitoring (SHM) is fundamental in decision making for damage identification and prescriptive maintenance of civil infrastructure. Traditional decision-making methods often fall short in integrating heterogeneous sensor data, inherent to SHM, and the natural language processing required to interpret, e.g., inspection reports, expert assessments, and historical documentation relevant to prescriptive maintenance. To address these limitations, this study introduces a framework for integrating large language models (LLMs) into SHM workflows, facilitating decision making in damage identification and prescriptive maintenance. The proposed framework links convolutional neural networks (CNNs) with generative LLMs. Unlike approaches based solely on prompt engineering, this study applies task-specific fine-tuning via low-rank adaptation (LoRA) to the Mistral-7B-Instruct-v0.1 model, using CNN-generated output and damage metadata as input. The results demonstrate – apart from the proof of concept – a successful generalization to previously unseen CNN-generated output, enabling context-sensitive damage identification. In conclusion, integrating LLMs into SHM workflows allows synthesizing heterogeneous sensor data and natural language, thus enhancing decision making for damage identification and prescriptive maintenance of civil infrastructure.

INTRODUCTION

The challenges posed by deteriorating civil infrastructure exert significant societal impact, affecting public safety, transportation, and economic stability. The combined effects of urbanization and climate change further intensify the challenges, as increasing population density and extreme weather events elevate the risk of structural failures and dysfunctional conditions [1]. Structural health monitoring (SHM) is essential for ensuring safety and functionality of civil infrastructure [2]. By recording sensor data relevant to the structural condition, SHM systems identify early signs of damage or deterioration, facilitating long-term condition assessment, while providing a sound basis for maintenance strategies [3]. In this context, reliable decision making is crucial for determining how the sensor data, usually being heterogeneous, is analyzed and translated into structural condition indicators and into actionable maintenance strategies [4]. In

addition to analyzing sensor data, decision making requires incorporating textual records of natural language derived from manual inspections.

A plenitude of approaches towards decision making in SHM have been proposed, which may be classified into (i) traditional methods and (ii) artificial-intelligence-based methods. The first category, traditional methods, includes decision making methods that have been the backbone of early maintenance systems, operating with relatively straightforward and interpretable rules [5]. Despite the clear interpretability and simplicity, the traditional methods are often limited in handling the large amount of heterogeneous data, fail to include textual records, and are often computationally expensive. The second category – based on artificial intelligence (AI) or machine learning (ML), respectively – has gained significant traction in recent years, due to the advancements in computational power and the expanding availability of sensor data. For example, deep learning models, such as convolutional neural networks and recurrent neural networks, have proven effective in processing time-series data, identifying anomalies, or forecasting degradation over time [6].

In summary, traditional decision-making methods are straightforward and interpretable, but of limited ability to handle large, complex, or heterogeneous data (typically subsumed as "big data"). AI methods, on the other hand, are flexible and capable of processing large amounts of sensor data in real-time but face challenges related to (i) processing unstructured, heterogeneous data of different type, as well as to (ii) integrating natural language processing, which is particularly relevant for extracting insights from reports, inspection logs, archival documentation, and other textual records relevant to decision making in prescriptive maintenance. To overcome the abovementioned limitations, integrating large language models (LLMs) into SHM workflows present a promising future direction. LLMs, typically based on transformer architectures for efficient processing of large-scale text data, are capable of integrating unstructured, heterogeneous data of different type and of generating nuanced insights by processing and understanding input presented in natural language [7].

This study presents a framework for integrating LLMs and convolutional neural networks (CNNs) into SHM workflows to support decision making in damage identification and prescriptive maintenance. The LLM-CNN framework combines a CNN for classifying structural response data, represented as spectrograms obtained from the Garbor transform [8], with a fine-tuned Mistral-7B-Instruct-v0.1 model [9], a transformer-based LLM with seven billion parameters. Fine-tuning is performed using damage probability distributions generated by the CNN from acceleration data collected in controlled laboratory experiments. The fine-tuned model translates the damage probability distributions into natural-language damage descriptions and actionable recommendations for prescriptive maintenance. The remainder of this paper is structured as follows. First, the design and implementation of the LLM-CNN framework, based on the open-source Mistral-7B-Instruct-v0.1 model, are described. Next, the validation of the framework is presented. Finally, the results of this study are summarized, conclusions are drawn, and potential future research directions are proposed.

DESIGN AND IMPLEMENTATION OF THE LLM-CNN FRAMEWORK

This section presents the design and implementation of the proposed LLM-CNN framework, which enables efficient identification of structural damage based on acceleration data. First, the architecture and the key components of the proposed LLM-

CNN framework are explained, followed by a description of the laboratory experiments, devised to derive acceleration data. Next, the CNN-based damage classification and the LLM fine-tuning are presented.

Architecture and key components of the LLM-CNN framework

At the core of the framework lies the pre-trained, open-source Mistral-7B-Instruct-v0.1 language model, which is adapted to the SHM domain via low-rank adaptation (LoRA) [10]. LoRA enables efficient fine-tuning by updating only a small subset of the language model parameters through low-rank decomposition, significantly reducing the computational effort. To bridge the gap between acceleration data and natural language processing, a CNN is introduced. The CNN is trained on spectrograms derived from Gabor-transformed acceleration data, capturing time-frequency characteristics of structural responses [11]. The CNN, trained with a softmax output layer, produces damage probability distributions over predefined damage classes. The damage probability distributions are combined with "structured" text containing damage class descriptions and recommendations for remedial action to form the training dataset for fine-tuning the LLM. The training dataset essentially consists of structured text files that pair each CNN output with semantically rich, domain-specific maintenance information, fed to the LLM via byte-pair encoding. Figure 1 illustrates the data flow of the LLM-CNN framework, including (i) preprocessing and classification of acceleration data using

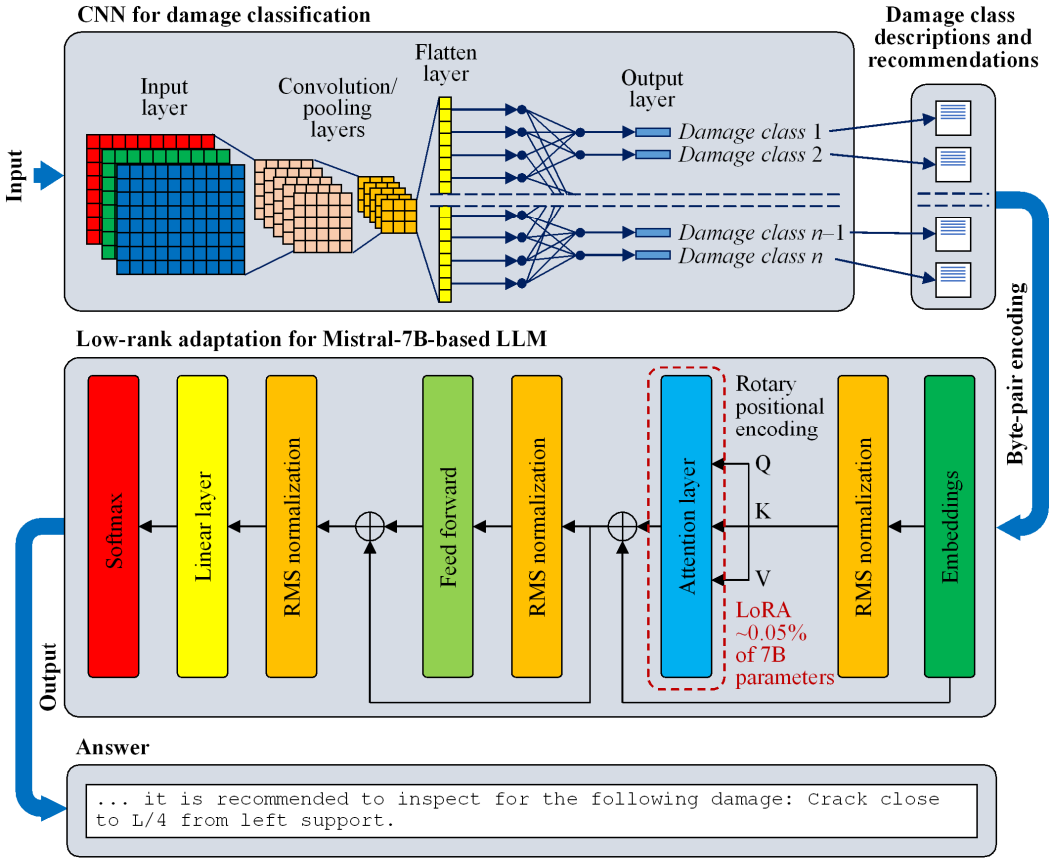

Figure 2. Overview of the proposed LLM-CNN framework.

the CNN, (ii) generating structured text to populate the training dataset for LLM fine-tuning via LoRA, and (iii) generating natural language output, i.e. a system "answer" to the user, with recommendations in response to the respective damage class. The acceleration data, used in this study, is obtained from laboratory experiments on a steel beam subjected to dynamic loading by a moving mass, under different damage classes, as shown in the next subsection.

Laboratory experiments

The experimental setup used for generating acceleration data, shown in Figure 2, consists of a steel beam with an HEB 100 cross section (width and height: 100 mm; flange thickness: 10 mm; web thickness: 6 mm) and an effective span of L = 5830 mm. The beam is supported by an adjustable strut system to ensure precise leveling. A moving mass system, operated via pulleys and an electric drive, allows for repeatable dynamic excitation at variable speeds between 40 mm/s and 500 mm/s for each damage class. The beam is instrumented with three sensor nodes, one close to the midspan (at a distance of 9L/20 from the left support) and two placed at quarter-length distances from each support for capturing the dynamic response corresponding to at least the first two vibration modes. The sensor nodes are equipped with accelerometers measuring vertical acceleration. Damage is systematically introduced into the beam, resembling realistic damage classes relevant to structural health monitoring (i.e. flexural cracks and loss of fixity in supports). The convolutional neural network is employed for classifying the spectrograms into predefined damage classes, as described in the following subsection.

Figure 2: Setup of the laboratory tests devised to generate training data.

CNN-based damage classification

Five damage classes are introduced to the beam, each implemented in two severity levels, for example through variations in crack depth or substituting fixed supports with springs of variable flexibility. Table 1 provides an overview of the considered damage classes.

TABLE I. DAMAGE CLASSES DEFINED FOR TRAINING THE LLM-CNN FRAMEWORK.

ID	Description
DC1	Intact beam
DC2	Crack close to L/4 from left support
DC3	Crack close to L/2 from left support
DC4	Crack close to 3L/4 from left support
DC5	Loss of fixity at left support
DC6	Loss of fixity at right support

For each damage class and severity level, 30 experiments are conducted, yielding a set of softmax-derived damage probability distributions generated by the CNN. The distributions reflect varying levels of classification confidence, depending on sensor node placement and the nature of the structural response induced by the respective damage. In some cases, no dominant damage class can be clearly identified, illustrating the inherent ambiguity present in realistic SHM damage classification tasks. To prepare the training dataset for fine-tuning the LLM, each damage probability distribution is paired with a structured text file containing a damage description and corresponding maintenance recommendations. An example of the training dataset, stored in JavaScript Object Notation (JSON) format, is shown in Listing 1. As can be seen from Listing 1, the data is structured in three categories, "Damage probability distribution", "Damage class", and "Recommendation". It should be mentioned that the CNN output describes a damage probability distribution over the defined damage classes, with each value corresponding to one of the classes DC1 to DC6 in Table 1. Since the softmax layer of the CNN is configured for six output classes, the output vector always contains six entries.

```
"Damage probability distribution": [0.1, 0.4, 0.0, 0.3, 0.2, 0.0]
"Damage class":   "Crack close to L/4 from left support"
"Recommendation": "Given the damage probability distribution [0.1,
                   0.4,0.0, 0.3, 0.2, 0.0], it is recommended to
                   inspect for the following damage: Crack close to
                   L/4 from left support."
```

Listing 1. Example of text-based training data used for LLM fine-tuning (in JSON format).

The classification results of the CNN are illustrated in Figure 3, showing softmax outputs for all damage classes. Consistent predictions are obtained for the classes DC1, DC5, and DC6, while the classes DC2, DC3, and DC4 exhibit diffuse damage probability distributions, indicating low classification confidence. The uncertainty particularly increases for less severe damage cases, suggesting that the CNN alone may struggle to distinguish between subtle structural changes. While major damage classes lead to clear and dominant class predictions, minor damage results in overlapping outputs and misclassifications. The ambiguities highlight the limitations of purely classification-based CNNs and support the integration of LLMs for improving the interpretability and robustness of decision making. The fine-tuning of the LLM using the training dataset, is shown in the following subsection.

Fine-tuning the large language model

The fine-tuning of the LLM model is conducted on a single NVIDIA RTX 4090 GPU using LoRA. Fine-tuning is performed over three epochs with an initial learning rate of 2·10-5, using AdamW optimization and mixed precision (fp16) to reduce memory consumption. Only the LoRA adapter layers are updated, while the core model weights remain frozen. The Mistral model is loaded in 8-bit format to reduce VRAM usage, and training is conducted with an effective batch size of 16 samples using gradient accumulation.

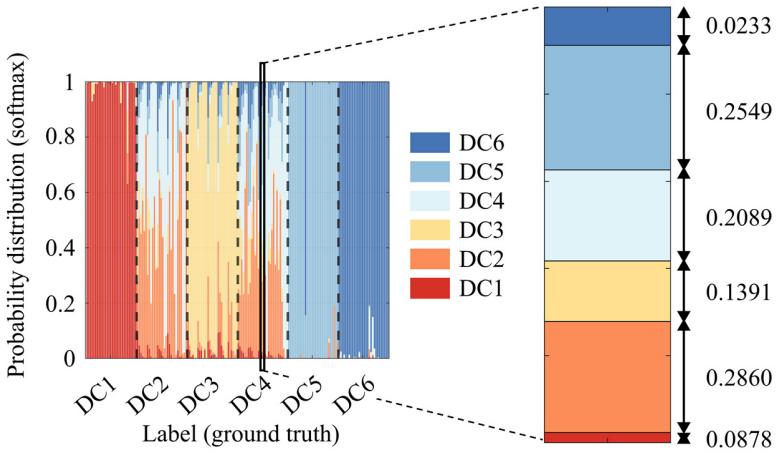

Figure 3: Softmax outputs across all damage classes.

To showcase the effectiveness of the fine-tuning process, the convergence behavior is monitored during training. The training dataset is split into 80% training subset and 20% validation subset. The validation is based on cross-entropy loss, which quantifies how well the damage-related probabilities predicted by the LLM match the reference labels. Figure 4a illustrates the cross-entropy loss per training step. A steadily decreasing curve indicates stable convergence of the LLM toward the training data, whereas plateaus or fluctuations may signal overfitting or ineffective learning. Figure 4b shows the gradient norm, which measures the magnitude of LLM parameter updates during optimization. High values may reflect instability, while very low gradients may indicate vanishing updates and slow learning progress. The perplexity and the gradient norm together provide insights into training dynamics and optimization behavior, which, as observed in Figures 4, are, in this study, characterized by a stable and well-behaved fine-tuning process without signs of overfitting or instability.

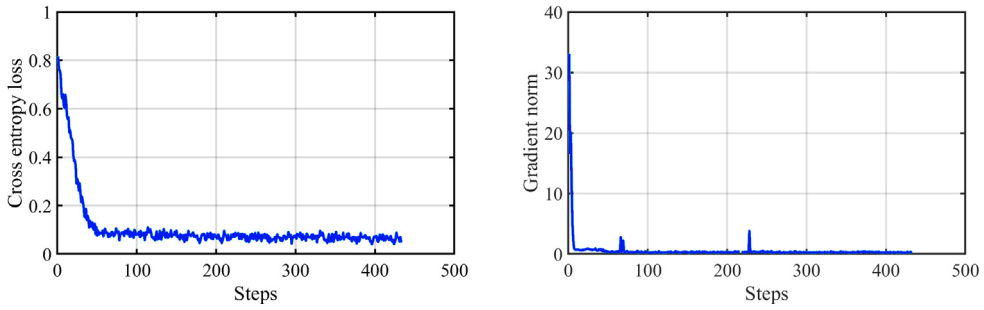

Figure 4: (a) Cross entropy loss for each learning step and (b) gradient norm for each learning step.

VALIDATION OF THE LLM-CNN FRAMEWORK

The fine-tuned Mistral model is validated using representative test data in structured JSON format. The data comprises a damage probability distribution over the predefined

damage classes, generated by the CNN, and the corresponding reference label with an associated maintenance recommendation. The validation test, presented in this section, focuses on the generalization capabilities of the LLM. The ability of the Mistral model to generalize beyond the training data, i.e. to interpret inputs not seen during training, is tested by introducing a previously unseen damage probability distribution, characterized by the absence of a dominant class, to the model. As can be seen in Figure 5, despite the inherent ambiguity, the fine-tuned LLM successfully identifies the correct damage class (in this case, DC2) and provides a plausible maintenance recommendation. While the results indicate promising generalization, caution is warranted regarding the behavior of the underlying pre-trained model. Typical effects, such as hallucinations or inconsistencies, may be inherent to the pre-trained model and persist despite the domain-specific fine-tuning. Moreover, fine-tuning may compromise capabilities of the pre-trained model, depending on the relative weighting of new parameters, controlled by the scaling factor in the LoRA configuration. The aforementioned observations highlight the need for responsible LLM deployment in engineering contexts, where explainability, reliability, and reproducibility are essential.

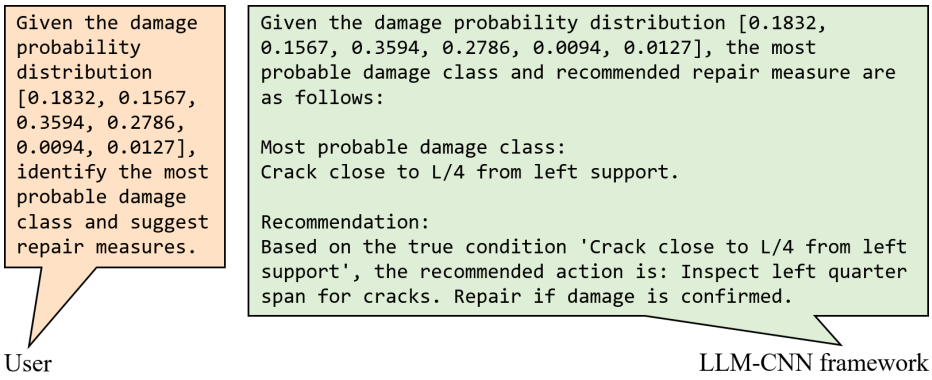

Figure 5: Example conversation of a user with the LLM-CNN framework.

SUMMARY AND CONCLUSIONS

Structural health monitoring has become a key enabler for decision making in damage identification and prescriptive maintenance of civil infrastructure. Conventional SHM methods often struggle to integrate heterogeneous sensor data and to process natural language content relevant to engineering decisions. This study has presented the CNN-LLM framework that links CNNs for damage classification with the generative capabilities of a fine-tuned Mistral 7B-Instruct-v0.1 large language model. The CNN processes spectrograms derived from acceleration data, while the LLM, adapted via low-rank adaptation, interprets damage probability distributions, identifies damage classes, and provides recommendations. Validation experiments using a laboratory beam setup with various damage classes have demonstrated the ability of the model to generalize to previously unseen data.

In conclusion, integrating fine-tuned LLMs into SHM workflows represents a paradigm shift in civil engineering decision making, as LLMs are capable of synthesizing heterogeneous sensor data and domain-specific, natural language, offering context-sensitive, interpretable outputs that go beyond rigid classification schemas.

Since infrastructure systems become increasingly complex, LLMs offer significant potential for enhancing data-driven, prescriptive maintenance strategies. To ensure responsible deployment of LLMs in civil engineering, future research may prioritize transparent and verifiable model behavior. Deterministic decisions in safety-critical contexts demand high levels of explainability and trust, aligning with ongoing developments in explainable AI. Ultimately, these efforts are expected to establish LLMs as foundational tools in engineering practice, bridging the gap between complex data, interpretability, and actionable decision support.

ACKNOWLEDGMENTS

This research was conducted as part of a joint initiative involving multiple research projects. The authors gratefully acknowledge the financial support provided by the German Research Foundation (DFG) under grants SM 281/9-3, SM 281/22-1, SM 281/32-1, and SM 281/33-1. Additional support was provided by the German Federal Ministry for Digital and Transport (BMDV) within the mFUND program under grants 01FV2013B and 01FV2059C. The authors also acknowledge the support provided by the Department of Education, Universities and Research of the Basque Government through the program Ikerketa Taldeak, supporting Grupo IT1519-22 and Grupo IT1443-22. Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES

1. Mohamadiazar, N., Ebrahimian, A., & Hosseiny, H., 2024. "Integrating deep learning, satellite image processing, and spatial-temporal analysis for urban flood prediction," *Journal of Hydrology*, 639(2024), 131508.
2. Al-Zuriqat, T., Chillón Geck, C., Dragos, K., & Smarsly, K., 2023. "Adaptive fault diagnosis for simultaneous sensor faults in structural health monitoring systems," *Infrastructures*, 8(2023), 39.
3. Dragos, K. & Smarsly, K., 2015. "A Comparative review of wireless sensor nodes for structural health monitoring," in: *The 7th International Conference on Structural Health Monitoring of Intelligent Infrastructure*. Turin, Italy, 07/01/2015.
4. Dragos, K. & Smarsly, K., 2017. "Decentralized infrastructure health monitoring using embedded computing in wireless sensor networks," in: Sextos, A., Manolis, G. (eds.) *Dynamic response of infrastructure to environmentally induced loads*. Lecture Notes in Civil Engineering, Volume 2. Springer, Cham.
5. Marcher, C., Giusti, A., & Matt, D. T., 2020. „Decision Support in Building Construction: A Systematic Review of Methods and Application Areas," *Buildings*, 10(10), 170.
6. He, Y., Chen, H., Liu, D., & Zhang, L., 2021. "A framework of structural damage detection for civil structures using fast Fourier transform and deep convolutional neural networks," *Applied Sciences*, 11(19), 9345.
7. Kumar, P., 2024. "Large language models (LLMs): Survey, technical frameworks, and future challenges," *Artificial Intelligence Review*, 57(2024), 260.
8. Gabor, D., 1946. "Theory of Communication. *Journal of the Institution of Electrical Engineers – Part III: Radio and Communication Engineering*," 93(26), 429-457.
9. Albert, Q. J., Sablayrolles, A., Mensch, A., & Bamford, C., 2023. „Mistral 7B," https://arxiv.org/abs/2310.06825.
10. Hu, E., et al., 2022. "LoRA: Low-rank adaptation of large language models," in: *The 10th International Conference on Learning Representations*. Virtual conference, 04/25/2022.
11. Dadoulis, G., Manolis, G. D., Katakalos, K., Dragos, K., & Smarsly, K., 2025. „Damage detection in lightweight bridges with traveling masses using machine learning," *Engineering Structures*, 322(2025), 119216.