

Received 4 September 2026, accepted 21 September 2026, date of publication 30 September 2026,
date of current version 9 October 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3739116

RESEARCH ARTICLE

Saliency Map-Guided Knowledge Discovery for Subclass Identification With LLM-Based Symbolic Approximations

TIM BOHNE^{1,2}, ANNE-KATHRIN PATRICIA WINDLER², AND MARTIN ATZMUELLER^{1,2}

¹Semantic Information Systems Group, Osnabrück University, Osnabrück, 49090 Niedersachsen, Germany

²German Research Center for Artificial Intelligence (DFKI), Osnabrück, 49084 Niedersachsen, Germany

Corresponding author: Tim Bohne (tbohne@uni-osnabrueck.de)

The DFKI Niedersachsen is sponsored by the Ministry of Science and Culture of Lower Saxony and the Volkswagenstiftung.

The paper describes work carried out in the context of the funded project Railway-X (BMW, FKZ: 13IPC039B).

ABSTRACT This paper proposes a novel neuro-symbolic approach for sensor signal-based knowledge discovery, focusing on identifying latent subclasses in time series classification tasks. The approach leverages gradient-based saliency maps derived from trained neural networks to guide the discovery process. To illustrate the potential, multiclass time series classification problems are transformed into binary classification problems through label subsumption, and classifiers are trained for each of these to yield saliency maps. The input signals and saliency maps, grouped by predicted class, are clustered under three distinct configurations. The centroids of the final set of clusters are provided as input to an LLM for symbolic approximation (textual description) and fuzzy knowledge graph matching to discover the underlying subclasses of the original multiclass problem. Experimental results on well-established time series classification datasets demonstrate the effectiveness of our saliency map-driven method for knowledge discovery, outperforming signal-only baselines in both clustering and subclass identification.

INDEX TERMS Neuro-symbolic learning, knowledge discovery, saliency maps, time series classification, large language models, explainable AI, generalized category discovery.

I. INTRODUCTION

Saliency maps, class activation maps, attribution maps, or, in general, methods highlighting regions of interest (ROI) that were decision-relevant in a deep learning system's prediction are widely used [1], [2], [3], [4]. However, beyond their typical application in explanation, they can also be considered as a first step towards knowledge discovery. Knowledge discovery in this work refers to the process of identifying interpretable, symbolic representations that reveal latent, semantically meaningful structures. Unlike pattern discovery, which identifies discriminative structures without semantics, knowledge discovery implies the extraction of structured, symbolic information, e.g., in the form of knowledge graph properties. In [1] and [2], gradient-based saliency maps derived from trained neural networks are leveraged as useful

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Wei¹.

explanation resources, highlighting issues such as overfitting and deviation from expert judgments. Yet, saliency maps can also support the identification of latent subclasses in classification problems, providing valuable knowledge that is unavailable a priori. The objective of this work is to develop and evaluate a saliency map-driven method for knowledge discovery that starts with a given n -class (e.g., binary) classification problem and identifies potentially existing and initially unknown subclasses beyond the original n . Therefore, the method can be applied directly in scenarios where a known classification problem does not precisely reflect all underlying classes. It will be exemplified and evaluated by converting multiclass classification problems into binary ones and then (re)discovering the original classes. Nonetheless, it generally showcases the potential of saliency maps for pattern and knowledge discovery. In principle, subclass identification can be achieved via clustering of the input samples, which serves as a baseline. We then show

that the clustering of input samples is measurably improved across several datasets and scenarios when saliency maps are added for each signal as a second dimension, demonstrating the meaningful nature of the information conveyed by the saliency maps in terms of pattern discovery. Following this, we present evidence that the centroids of these improved clusters are better suited for KG matching and, ultimately, subclass discovery. The fundamental premise is that the various subclasses should manifest in distinct saliency maps, which can subsequently be facilitated in clustering, i.e., subclass discovery via multivariate saliency map-aided clustering. The centroids of the resulting clusters are provided as input to an LLM for symbolic approximation and fuzzy knowledge graph (KG) matching. This establishes a bidirectional synergy that bridges neural network-based pattern discovery and KG-based knowledge discovery. According to the taxonomy in [5], which categorizes neuro-symbolic systems, the approach belongs to the *Neuro | Symbolic* category. Our core contributions are the demonstration of the potential of saliency maps for pattern and knowledge discovery that goes way beyond their typical application as explanation or debugging tools, as well as the development and evaluation of an effective, novel neuro-symbolic approach for saliency map-driven knowledge discovery, aimed at identifying latent subclasses in time series classification tasks. Crucially, the approach is general and domain-agnostic. This work demonstrates the principle and effectiveness with time series data from various domains, but it is also applicable to, e.g., image data.

The remainder of the paper is structured as follows: Section II discusses related work. After that, Section III presents our approach, before evaluating it in Section IV. Finally, Section V concludes with a summary and promising directions for future research.

II. RELATED WORK

In [1] and [2], the potential of saliency maps for knowledge discovery is anticipated, i.e., allowing the identification of latent conceptual structures within the data. While the saliency maps play a subordinate role as an explanatory component in a larger neuro-symbolic architecture for anomaly detection and fault diagnosis in [1] and [2], the present work aims to study the hypothesis of such saliency maps enabling knowledge discovery. Although saliency map generation for deep learning is mainly applied to image data, there are some evaluations of consistency and robustness of their explanations when applied to time series classification, e.g., [6]. It is about evaluating the saliency map generation techniques themselves, not their use as means of knowledge discovery. Most of the literature is either concerned with generating saliency maps, e.g., *Grad-CAM* [7], *Grad-CAM++* [8], *Score-CAM* [9], and *LayerCAM* [10], or with applying them as explanation tools. A recent case of the latter is the work of Hossain et al. [3] in which the authors apply saliency maps generated with *Grad-CAM* to histopathology

images for classification of certain diseases, pinpointing crucial areas for predictions.

Mahapatra et al. [4] work on self-supervised generalized zero shot learning for medical image classification using interpretable saliency maps. Moreover, they propose a novel method to generate *Grad-CAM* saliency maps that are supposed to highlight diseased regions with greater accuracy. They enforce the saliency maps of different classes to be different and ensure that clusters in the space of image and saliency features yield centroids with similar semantic information. In particular, they use the saliency maps as an additional constraint for the clustering, as an effective way to incorporate an attention mechanism. Mahapatra et al. [4] share some common goals with this work, e.g., exploiting information from saliency maps to improve a clustering process. Nevertheless, they incorporate them into the clustering in a different way, i.e., as part of the optimization criterion. In contrast, we perform multivariate *Dynamic Time Warping Barycenter Averaging* (DBA) *k*-means clustering [11] with one dimension being the signals themselves and the other the corresponding saliency maps. Additionally, they are not considering the identification of subclasses and also not knowledge discovery.

LLMs have recently been applied to various time series tasks, including reasoning and the alignment of time series to textual descriptions. For instance, Fons et al. [12] evaluate the performance of five general-purpose LLMs in matching time series to textual descriptions. They observe a close-to-perfect accuracy of *GPT-4* in intra-dataset matching, where qualitative and quantitative descriptions are given. However, a significantly lower accuracy is observed in cross-dataset matching with qualitative descriptions only. There has also been work on LLMs specialized on time series reasoning, such as *ChatTS* [13]. A multimodal LLM that facilitates understanding and reasoning on multivariate time series and the alignment of time series with textual descriptions has been demonstrated to outperform *GPT-4o* in these tasks. Thus, LLMs are a viable tool for matching time series to textual descriptions. Similar to [12], this work utilizes these matching capabilities, in this case for symbolic approximation in a knowledge discovery scenario.

The problem studied in this work is closely related to the field of *Fine-Grained Category Discovery Under Coarse-Grained Supervision* (FCDC). In the FCDC setting, only coarse-grained labels are available during training and the objective is to identify unknown fine-grained categories within each coarse-grained class. The main difference between FCDC and our problem formulation is the symbolic approximation based on fuzzy knowledge graph matching, an additional step performed after fine-grained categories are discovered (pattern discovery). A representative work from the FCDC literature is [14]. Their proposed architecture learns text representations using a combination of coarse-grained supervision and weighted self-contrastive learning. While samples belonging to the same coarse-grained class are encouraged to remain close

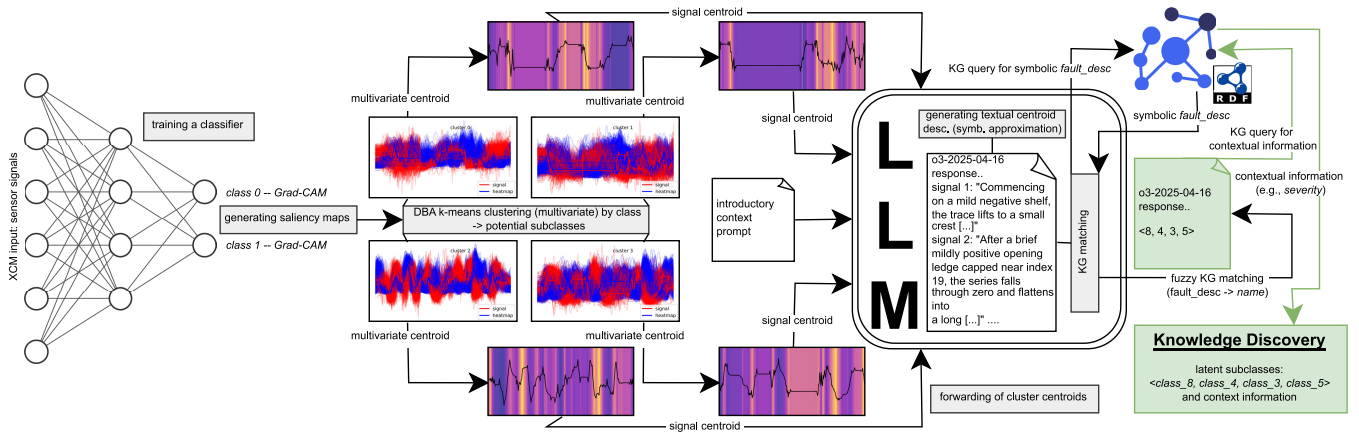


FIGURE 1. Pipeline of saliency map-guided (multivariate) knowledge discovery for subclass identification, starting at step (4).

to one another in the learned representation space, the contrastive objective prevents them from collapsing into a single region and promotes the emergence of separable, fine-grained structures. After representation learning, k -means clustering is applied to the resulting BERT¹ representations to discover fine-grained categories.

Building upon the work of An et al. [14], subsequent research has focused on improving the learned representation space for fine-grained category discovery. For example, the DNA framework [15] focuses on learning more compact fine-grained cluster representations through neighborhood-based representation learning. It extends the traditional contrastive loss to a form with multiple positive keys obtained from the nearest neighbors of a query sample. First, the nearest neighbors are filtered according to three principles to eliminate false positives. The remaining neighbors are then aggregated through *Denoised Neighborhood Aggregation* (DNA), which aims to pull queries and their neighbors closer together in order to learn compact, separable cluster representations. The authors theoretically prove that their learning objective is equivalent to a clustering loss.

An example from the field of computer vision is [16]. Rather than relying on a separate clustering step, the proposed FALCON framework jointly infers latent, fine-grained categories and the relationships between coarse- and fine-grained classes. During training, specialized loss functions encourage neighboring samples in the representation space to receive consistent fine-grained predictions while at the same time promoting confident predictions. Additional regularization prevents degenerate solutions in which all samples collapse into a single, fine-grained category.

Similar to [14], we employ k -means clustering to discover fine-grained categories while making use of a supervised deep classifier trained on coarse-grained labels. However, unlike [14], [15], and [16], our objective is not to use specialized loss functions that explicitly encourage the

emergence of fine-grained, cluster-like structures in the representation space. Instead, we investigate the potential of finding these clusters in the raw input time series with the help of saliency maps obtained from a standard classifier trained solely with coarse-grained supervision. In contrast to existing FCDC approaches, our classifier is trained using conventional cross-entropy loss without a mechanism that explicitly promotes fine-grained separability. Consequently, our goal is to examine whether standard supervised learning implicitly captures meaningful fine-grained distinctions that can be uncovered from saliency maps generated as a byproduct of the training process. Furthermore, existing FCDC research has largely focused on applications in natural language processing and computer vision. To the best of our knowledge, no prior work has explicitly investigated FCDC in the context of time series data, nor has it combined FCDC with symbolic approximation via knowledge graph matching.

A further related field is *Generalized Category Discovery* (GCD), which aims to identify both known and novel categories from unlabeled data using limited labeled data from known categories. Fan et al. [17] propose a method for LLM-enhanced GCD via iterative graph diffusion. In particular, they develop a dynamic optimization model that leverages diffusion graphs and LLM annotations for obtaining high-quality supervision. It involves a two-stage LLM prompting strategy to accurately assign selected samples to existing or novel categories. The main contributions of the work are a semantic correlation graph over node representations that enables the identification of two types of high-value samples (hard negatives and central samples), and a label propagation mechanism using that graph. Overall, unlabeled data is encoded using a pre-trained embedding model, followed by clustering and graph construction. Then, high-value samples are selected, and the two-stage prompting strategy classifies them into novel or known, followed by selective label diffusion. Finally, the embedding model is fine-tuned on the expanded dataset.

¹Bidirectional Encoder Representations from Transformers.

Another recent GCD method is proposed by Zou et al. [18] and relies on diverse LLM feedback. Similar to [17], the target domain is natural language. The authors provide a unified framework for GCD that actively learns from LLM feedback of three types: (1) improving instance-level contrastive features, (2) generating category descriptions for new categories, and (3) aligning uncertain instances with LLM-selected category descriptions. Thus, LLM feedback is used as a cost-effective alternative to human annotations, aiming for additional supervision. Initially, data is forwarded to a text encoder to extract features for clustering. Then, high-entropy instances are flagged as ambiguous and prepared for the LLM feedback. Finally, the feedback is used for neighborhood contrastive learning and pseudo category alignment. The framework is explicitly limited to textual data.

Banerjee and Biswas [19] introduce a language-assisted feature representation and lightweight active learning approach for *On-the-Fly Category Discovery* (OCD), a particularly challenging variant of GCD. The framework relies on the powerful knowledge approximation of LLMs to obtain unique feature representations. A contrastive loss function is used to sync image embeddings with semantic information embedded within class-specific textual descriptions. During inference, samples are categorized as known or unknown based on an adaptive threshold, and there are test-time updates of the classifier. The goal is to learn an association from image to latent space, such that images of novel classes will be embedded in a semantically meaningful way. The authors point out that a potential limitation is the reliance on LLMs for obtaining class representations, in particular, if a class is uncommon and may not have been present in LLM training.

Discussion.

In the following, similarities and differences between GCD and the present work are discussed. Obviously, this work shares several high-level similarities with GCD, but it tackles the problem from a different angle. The three main similarities are the aim of identifying latent categorical structures within data, the use of LLMs to bridge the gap between data patterns and human-readable descriptions, and the reliance on clustering algorithms as a preliminary step for identifying new classes. Despite being conceptually compatible in the pursuit of discovering new categories, the methodologies differ considerably. This paper utilizes gradient-based saliency maps from trained neural networks to guide a clustering process, whereas the GCD approaches rely on techniques such as contrastive learning, entropy-based uncertainty scoring, and graph diffusion for category separation. The roles of LLMs also differ greatly. While LLMs are used in post-processing steps at the end of the pipeline in this work (translating subsymbolic patterns into symbolic approximations and performing fuzzy matching with KG entries to ground discoveries in established semantics), LLMs are more directly involved in the iterative discovery process in the GCD methods.

One seeming difference is the open-world nature of GCD compared to the bounded latent subclass discovery in this work. Since this work begins with a known classification problem, it appears to only aim to discover latent, fine-grained subclasses within the predicted, coarse-grained macro classes. Thus, GCD appears to look outward to discover entirely new classes in unlabeled data whereas this work appears to look inward to identify latent hierarchical structure within established classes. This is because we assume that each sample belongs to an overarching, known category (e.g., binary macro labels). The discovery occurs within the boundaries of these fully labeled macro classes. GCD, on the other hand, works with partial / open-world supervision. The discovery process involves separating the known from the unknown in an unconstrained space. However, both GCD and this work belong to the family of open-world machine learning and category discovery. GCD can be characterized as horizontal expansion, e.g., when classifying animals. Let the initial labels be “Cat” and “Dog”. Then, imagine there is a large, unlabeled dataset containing samples of cats, dogs, birds, and snakes. GCD recognizes cats and dogs and expands the categories horizontally to discover the new labels “Bird” and “Snake”. This work can, a priori, be characterized as vertical (hierarchical) expansion. Once again, consider an animal classification scenario. This time, however, every single sample is already labeled as either “Mammal” or “Non-Mammal”. There is no unlabeled data. The method aims to look vertically downward into the “Mammal” category to discover the hidden subclasses “Cat” and “Dog”. Therefore, GCD relies on measuring distances between known categories and unknown data points to discover novel categories in an unconstrained environment, while this work already has a boundary (e.g., “Mammal”). To find subclasses hidden inside the boundaries of macro classes, it uses the classifier’s internal logic to generate saliency maps. Thus, from a methodological perspective, our approach is coarse-to-fine category discovery. It can group the animals based on what features were most important, i.e., the reasons for being classified as “Mammal” can be diverse. In short, from this perspective, GCD performs disjoint category discovery for missing labels while this work discovers finer granularity hidden inside the existing label space. However, if we assume the original classes to not necessarily be correct, the differences in the end result diminish. For instance, if the original binary problem contains labels “Cat” and “Dog” and a presumed subclass representing “Bird” is discovered (via symbolic concept matching), the original binary setup is imprecise, as it is in fact a parallel class, not a subclass. Then, effectively, it is an open-world discovery. In such a scenario, the imprecise, high-level labels “Cat” and “Dog” function similarly to the limited labeled support set in GCD. Hence, the seeming difference of horizontal vs. vertical discovery, or open-world vs. closed-world, only holds from an algorithmic perspective with regard to the first stage of our work.

In the first stage, by definition, discoveries are considered subclasses (i.e., subordinate structure), though they are not necessarily de facto subclasses from a semantic perspective. In principle, the results can be open-world, i.e., parallel novel categories. There is no fundamental difference. We only require subclasses from the process view, not the semantic view. Essentially, it is about whether the coarse-level classification is successful and whether there is meaningful saliency structure that distinguishes the latent categories, regardless of whether they are parallel or vertical in the semantic sense. This is also confirmed by some of the datasets used in the evaluation. To conclude, if the initial labels are imprecise, our method acts as a de facto open-world discovery tool. Otherwise, it performs subclass discovery (FCDC). Since this work employs a two-step neuro-symbolic approach, the final symbolic step can effectively break the hierarchy imposed by the neural step. During the neural phase, the data is forced into predefined categories. At this stage, clusters appear to be subclasses mathematically, but ultimately, a semantic resolution takes place. In the end, the result may align perfectly with the standard GCD open-world outcome, but the method acts as a divide-and-conquer heuristic rather than a global clustering approach.

An interesting follow-up question is whether it is more likely to discover actual semantic subclasses or new, parallel classes as a result. It depends on the domain. For instance, in binary anomaly detection, novel classes will most likely be actual subclasses because anomaly detection is inherently binary (i.e., anomaly or regular). Yet, one can contrast this with many problems in which parallel discoveries are likely. Both cases are feasible, and the likelihood depends on the application domain, labeling quality, and, e.g., the introduction of unseen classes.

Another valid question is whether the reliance on a predefined KG as in this work limits the applicability compared to the more generative LLM labeling used in the GCD works. It certainly limits the general applicability, but it also increases the reliability because it is based on actual knowledge, not potentially hallucinated facts. Relying on a KG serves as a strict guardrail against hallucinations, trading boundless open-world flexibility for grounded reliability. In the event that the LLM's description does not align with a KG entry, the system does not simply invent a new category. Instead, it flags the cluster as an unknown pattern discovery that necessitates human expert verification. Additionally, once a match is made, the system inherits verified contextual information, such as fault severity, from the KG, thereby ensuring that any downstream decisions are based on established facts rather than potentially incorrect generated text. Therefore, GCD methods generally focus on exploration with a risk of hallucination. This work takes a stricter approach, assuming that if a pattern is meaningful, it must either map to a verified fact or be assessed manually by a human expert. Finally, all three GCD approaches operate with data that is different in nature compared to this work:

[17] and [18] employ natural language datasets, while [19] relies on image data.

III. METHOD

The saliency map-based knowledge discovery comprises seven high-level steps, which are elaborated in the following subsections:

- 1) Defining an exemplary symbolic knowledge representation, i.e., an ontology (cf. Sec. III-A)
- 2) Selecting suitable (criteria-compliant) multiclass time series classification datasets from the UCR archive [20] (cf. Sec. III-B)
- 3) Transforming the multiclass classification problem into a binary one via label subsumption (cf. Sec. III-C)
- 4) **Training a classifier** for the (binary) classification task (cf. Sec. III-D)
- 5) **Generating saliency maps** based on the trained classifier and test dataset (cf. Sec. III-E)
- 6) **Clustering** of the input signals, grouped by predicted class (c_0, c_1): univariate (signal only), univariate (saliency map only), **multivariate** (signal + saliency map) (cf. Sec. III-F)
- 7) **Forwarding of cluster centroids** to the LLM for **symbolic approximation** and **fuzzy KG matching** (cf. Sec. III-G)

Crucially, we regard each dataset as a binary classification problem, although both classes comprise a number of subclasses that we aim to (re)discover. This is done to demonstrate the feasibility of the method in a setup in which ground truth information is available. When applying the approach, it begins at step (4), omitting those that are part of the evaluation protocol (cf. Fig. 1). Moreover, only the multivariate clustering is employed in step (6) in Fig. 1, since the others are only introduced for evaluation in Sec. IV.

A. SYMBOLIC KNOWLEDGE REPRESENTATION

An aspect of the approach that should not be neglected is the existence of a symbolic knowledge representation, which ultimately turns pattern discovery into knowledge discovery. Although even the domain-agnostic characteristics of the symbolic information can be arbitrary for the approach, we will consider general sensor faults as a placeholder for any kind of symbolic description of sensor signals due to the practical plausibility and the thematic link to [1] and [2]. Maintenance systems often describe signal characteristics linked to problems. The symbolic class descriptions for UCR datasets are treated as representative sensor fault types. To capture this domain-specific knowledge, an ontology is defined, and a corresponding KG is constructed for each dataset by populating it with instance data. The facts are stored in the *Resource Description Framework* (RDF) format and hosted on an *Apache Jena Fuseki*² server. Knowledge retrieval as well as KG updates work via predefined SPARQL

²<https://jena.apache.org/documentation/fuseki2/>

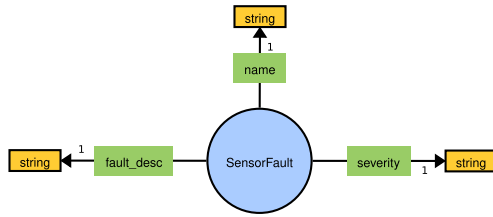


FIGURE 2. Minimalistic sensor fault ontology.

queries and HTTP requests to the respective endpoints. The minimalistic ontology used in this paper is shown in Fig. 2.

This could be regarded as an excerpt from a larger ontology contextualizing the sensor faults, e.g., the one used in [1]. However, as only this one concept is required to demonstrate the methods in this work, anything superfluous has been omitted. The main concern with richer ontologies is knowing where to look for matching entries, i.e., knowing where signal descriptions are stored. Since this would apply to a scaled-up version of the ontology, richer ontologies would not be an issue. A more crucial potential issue is an excessive number of signal descriptions in a large-scale instantiated KG. Evaluating whether the LLM's matching performance would degrade with too many different descriptions is difficult because of the many other factors involved, such as the proximity of the textual descriptions. Notably, LLM-based matching functions only as a post hoc step that translates discoveries. The actual subsymbolic discovery remains unaffected by this, even in the most challenging scenarios. Therefore, when promising subsymbolic discoveries do not lead to successful matchings, this issue may be present, and it is identifiable. In short, scaling the ontology and KG does not affect pattern discovery, but it can affect LLM-based matching if there are huge numbers of signal descriptions that are hardly distinguishable. In the following, each class of each dataset is represented as an instance of **SensorFault**. The **name** property is always instantiated with the ground truth label of the corresponding class and **fault_desc** represents the crucial textual representation of the corresponding signal, e.g., *"Beginning slightly below zero, it climbs sharply to its main crest, drops into a mid-level dip, and then fades through progressively weaker ripples into an extended negative stretch"*. Finally, **severity** is set to arbitrary values and serves as a placeholder for all kinds of conceivable auxiliary information. Facilitating this KG, the results of the saliency map-based pattern discovery can be used to assign each discovered subclass to a potentially existing counterpart in the KG, thereby turning it into an actual symbolic knowledge discovery. If a detected cluster can be matched to some **SensorFault**, it is a potential knowledge discovery. Otherwise, a new subclass not yet part of the KG might have been identified, which should be forwarded to a human expert for verification. There are three cases:

- 1) **Knowledge discovery**: Identified cluster, matched symbolic description (with uncertainty, cf. Sec. IV)
- 2) **Pattern discovery**: Identified cluster, no matching, unknown category to be labeled by human, i.e., KG extension
- 3) **No discovery**: Identified cluster, no matching, poor quality

The first two are desirable. The last one could be prevented via cluster filtering, e.g., based on ground truth independent metrics presented in Appendix A. Given that we transform multiclass classification problems into binary classification problems, the primary goal is to strive for the knowledge discovery case.

B. DATASET SELECTION

The aim is to transform a multiclass classification problem into a binary one and subsequently (re)discover the remaining subclasses with the help of saliency maps in order to obtain a ground truth evaluable setup. For this, suitable multiclass datasets from the wide-spread UCR (University of California, Riverside) time series archive [20] are used. For a dataset to be considered suitable, we define certain requirements. It is important to acknowledge that the method is not intended to be successful for every dataset. There exist certain natural requirements that do not constitute a weakness; rather, they are an inherent aspect of the problem nature. Generally, the proposed method is applicable to all time series classification problems. However, the following criteria can help guide whether its application is appropriate for a particular dataset, and success can be estimated using quantitative metrics.

Evidently, the approach demands significantly more than two classes Y . Datasets with $|Y| \in [6, 16]$ are considered, in order to avoid super-classes that exhibit too much variance through a large number of subclasses per class Y_i . To ensure a meaningful evaluation, we only consider datasets that have a large test set with a large number of samples per class – specifically those containing at least 100 time series per class Y_i . Datasets comprising time series shorter than 100 time steps are also excluded from consideration. Image-to-time-series datasets are disregarded in order to maintain a focus on inherently temporal data. An intriguing question is whether the best-reported classifier accuracy in the literature is an appropriate criterion. It could lead to a bias towards favoring easy-to-classify datasets. The improvement through saliency maps may be particularly significant when the dataset is difficult to classify. On the other hand, an accurate classifier should correlate with an enhanced probability of obtaining reliable and useful saliency maps. If a classifier produces a large number of misclassifications, it is expected that the class activation maps of the misclassified time series are less meaningful. Thus, the classification problem should be reasonably tractable to solve, meaning that it is possible to train a classifier with a decent test accuracy. Ultimately, we opt for a conservative classifier accuracy threshold of 0.6 according to [21], who studied the performance of different deep neural network architectures on the UCR time

series datasets. This serves to prevent meaningless saliency maps while ensuring that it remains easily attainable for most datasets.

When filtering the UCR datasets according to these requirements, the following remain: *InsectWingbeatSound*, *Mallat*, and *UWaveGestureLibraryAll*. The three selected datasets stem from distinct domains, enabling the evaluation of the approach across varied datasets with differing semantics. *InsectWingbeatSound* [22] captures the power spectrum of the sound of insects passing through a sensor. Each of the 11 classes represents a certain type of insect. It has a training set of size 220 and a test set of size 1980, with a time series length of 256. *UWaveGestureLibraryAll* [23] consists of recordings of a three-axis accelerometer. The 8 classes represent gestures performed while holding a handheld accelerometer device. Each time series has a length of 945 and consists of the concatenation of the x , y and z axes. It has a training set of size 896 and a test set of size 3582. The time series are downsampled to a length of 256 in the experiments. *Mallat* is a simulated dataset comprising 8 classes. The details of its construction are described in [24]. It has a training set of size 55, a test set of size 2345, and the original length of the time series is 1024, which is downsampled to 256 again.

A key consideration is how distinct classes must be (inter-class distance) and how similar samples within a class must be (intra-class distance) for the approach to work. It is imperative that samples belonging to a subclass are sufficiently homogeneous. In the end, the selected datasets stem from distinct domains, enabling the evaluation of the approach across varied datasets with differing semantics and difficulties.

C. CREATING BINARY CLASSIFICATION PROBLEMS VIA LABEL SUBSUMPTION

As previously discussed, each selected dataset requires label subsumption, i.e., conversion of a multiclass classification problem into a binary classification problem. This is a straightforward process, in which the class labels Y are subsumed in an alternating fashion:

$$Y_i = \begin{cases} 0 & \text{if } Y_i \bmod 2 = 0 \\ 1 & \text{otherwise} \end{cases} \quad \text{for all } i$$

It is crucial to evenly distribute the original classes among the new binary classes, since subsuming the majority of the classes under one would lead to a strong class imbalance in the binary classification problem. To reiterate, converting a multiclass problem into a binary one serves to demonstrate the approach in a ground truth evaluable setup. We pretend to only know two general classes and assume that the subsumed ones are subclasses of the two classes in the binary classification problem. This enables us to have ground truth subclasses to be discovered, i.e., the label subsumption enables us to know that there is semantically meaningful substructure to discover. Although, from the

evaluation's point of view, it is rediscovery, the system has no access to the original labels, so from its perspective it is genuine discovery. While it is not representative of practical cases where no latent substructure exists (recognizable via quantitative evaluation), it provides a suitable representation of situations in which it does. Given that the multiclass datasets utilized in this work are not constructed for this purpose, but are instead practical and well-established ones, there is no fundamental difference between a real-world scenario in which there is known structure to discover and the present setup. When it is a priori unknown whether there is substructure to discover in practice, validation is based on uninformed quantitative metrics. Moreover, the assumption is minimal – a binary classification problem with random subclass assignments. In other scenarios, there might be more known classes or a more structured assignment. Essentially, this results in a challenging general discovery scenario due to the way the substructure is hidden. In some cases, the subsumption scheme can reduce the difficulty by providing initial structure, or even increase it by constructing particularly challenging setups that exploit the structure. We avoid that and evaluate it without using any prior dataset-specific knowledge. In practice, the approach would be applied to an n -class (e.g., binary) classification problem in order to find potentially existing ($> n$) subclasses. Compared to the multiclass classification problem, the binary problem should, in theory, be considerably less challenging. A question of interest is the importance of aiming for a highly accurate binary classifier. Ideally, this results in enhanced saliency maps, which in turn may facilitate improved clusterings. The number of subclasses that can be discovered is contingent upon the difficulty of distinguishing the classes and the classifier performance. Certain subclasses may prove harder to match than others. It is crucial to note that the objective of this work is not finding the best possible classifier, but rather to demonstrate the feasibility of the principles. Every single subclass that is identified is gained knowledge. The rationale behind the challenge of training a binary classifier with high accuracy is that the two classes may contain remarkably distinct signals (i.e., subclasses) based on the particular dataset.

A future work approach could also investigate the selection of the number of classes based on some estimated groups, i.e., less than the original number of classes, but potentially more than two. Then, it would be easier to distinguish the classes and force the model to learn the differences. However, this would also imply a considerable amount of preliminary analysis. In general, it is plausible to assign similarly-shaped subclasses to one class in label subsumption. However, then it might suffice to learn some high-level distinctions rather than the nuances, i.e., for learning success it might be better to actually assign close subclasses to different classes in the binary problem to enforce learning of nuances. We assign them randomly (alternating). This may be more aligned with the option that emphasizes learning nuances. In essence, we regard each dataset as a binary classification problem,

although both classes comprise a number of subclasses that we aim to (re)discover via saliency map-based patterns.

D. BINARY TIME SERIES CLASSIFICATION

The task comes down to binary univariate time series classification. For a time series $V \in \mathbb{R}^n$, performance was best when standard z -normalization was applied to the raw time series data, i.e., $V' := \{\frac{x_i - \mu V}{\sigma V} | x_i \in V\}$, with mean μV and standard deviation σV . An additional preprocessing step is to downsample each time series to a length of 256 as a compromise between achieving the highest accuracy and ensuring a reasonable runtime in later steps.

Since the main focus of this work is not to propose a superior ANN architecture for binary classification of time series data, we compared several standard architectures from the *tsai*³ library and finally selected the best-performing one that allows for a relatively straightforward extraction of saliency maps, which was the *Explainable Convolutional Neural Network for Multivariate Time Series Classification* (XCM) [25]. An aspect to keep in mind is that the worse the classifier, the worse the saliency maps – and thus, the higher the chance of the input signals in isolation being more successful in clustering. Therefore, obtaining high-accuracy binary classifiers remains a concern. The hyperparameters are always based on several initial test experiments for each dataset. Table 1 shows the binary classification results. The third column refers to the batch size (training / validation), and the fourth to the test accuracy. The following sections are based on the performance on the test data.

TABLE 1. Hyperparameters / results for binary classification.

dataset	architecture	bs	epochs	acc
<i>Mallat</i>	XCM	8/16	300	0.957
<i>UWaveGestureLibraryAll</i>	XCM	32/64	300	0.974
<i>InsectWingbeatSound</i>	XCM	8/16	20	0.865

E. SALIENCY MAP GENERATION FOR TIME SERIES

There are several techniques used in deep learning to visualize areas of an input that are most relevant to predicting a certain class, e.g., *Grad-CAM* [7], *Grad-CAM++* [8], *Score-CAM* [9], and *LayerCAM* [10]. They provide a way to interpret the decision made by an ANN model (with compatible architecture) by highlighting the regions of the input that contribute the most to the classification result. The basic idea behind vanilla *Grad-CAM* is to use the gradients of the output class with respect to the feature maps of the last convolutional layer in the network [7], i.e., to ask how much the output would change if the corresponding element in the feature map were to change. The resulting saliency maps can be overlaid on top of the original input to provide a visual explanation. The regions with higher values in the saliency map correspond to the regions of the input that had the greatest impact on the model's prediction. The general method of using gradients to visualize the importance of

input features is based on the idea of a hierarchy of feature representations, as in CNNs, where the output of each layer can be interpreted as a set of learned features that capture increasingly complex aspects of the input.

With time series data, the gradient-based approaches work with 1D convolutional layers along the temporal dimension of the input. Each of the methods receives the normalized time series values $V' \in \mathbb{R}^n$, the trained model M , and an optional prediction y (default is the “best guess”, i.e., $y = \operatorname{argmax}_i P(i | V') \forall i \in \{0, 1\}$) as input, and outputs a heatmap $H \in [0, 1]^n$. Each value $h_i \in H$ rates the importance of a corresponding time series value $v_i \in V', \forall i \in \{1, \dots, n\}$. Since the XCM [25] model allows for relatively straightforward extraction of saliency maps, we use it to demonstrate the principle. In future work, it might be worthwhile to compare different generation techniques and evaluate their respective advantages and disadvantages for knowledge discovery. Notably, observations that are key to a classification (e.g., an insect's wing beat) are not assumed to be temporally aligned in the time series. Since the saliency maps are clustered using DBA, temporal displacements can be disregarded.

F. UNIVARIATE AND MULTIVARIATE CLUSTERING

This section is concerned with the clustering of the input signals and saliency maps, grouped by predicted class (c_0, c_1), resulting from the binary classification, with the objective of subclass identification. We evaluate three distinct clustering modes:

- univariate (input signals only)
- univariate (saliency maps only)
- multivariate (input signals + saliency maps, cf. Fig. 1)

Therefore, in total, there are 18 different configurations, i.e., clusterings: 3 datasets $\times$ 2 classes $\times$ 3 modes. The aim is to assess whether the generated saliency maps are beneficial for the clustering process. Given that all of the employed clustering methods are k -means variations, the initial step is to determine a reasonable k value, i.e., the estimated number of subclasses. Since we are treating it as a knowledge discovery problem, the number of subclasses to be expected is unknown and must be estimated. This is achieved by using the *Elbow method* with *Dynamic Time Warping* (DTW) as metric and $k \in [1, 10]$. After experimenting with different k -means approaches and metrics, *Dynamic Time Warping Barycenter Averaging* (DBA) k -means proved to be the most effective. The k range is the result of experiments and aligns with the number of classes range defined as a criterion in the dataset selection (cf. Sec. III-B), as well as the fact that the two classes are clustered separately. Setting the upper bound of the k range to an excessively high value can distort the *Elbow* estimation. To avoid local minima, the DBA k -means is always performed 20 times with different centroid seeds. The maximum number of iterations is set to 500 for the k -means algorithm and to 300 for computing the barycenter (centroid) of each cluster.

³<https://timeseriesai.github.io/tsai/>

DTW (used in DBA k -means) is not scale-invariant, i.e., differences in scale across variables can dominate the distance calculation. We are typically more interested in relative shape similarity than absolute magnitude of the time series. Thus, having no or poor normalization can distort the clusters. In the multivariate case, the first dimension corresponds to the z -normalized signals, while the second dimension corresponds to a saliency map (heatmap) for each. To ensure that each dimension contributes equally, z -normalization is also applied to the saliency maps. Experimentally, this turned out to be an advantage. To provide two examples, the resulting multivariate clusters for *InsectWingbeatSound* (c_0) are shown in Fig. 3 and the ones for *Mallat* (c_0) in Fig. 4. It is evident that some clusters contain rather uniform samples, while others appear more chaotic.

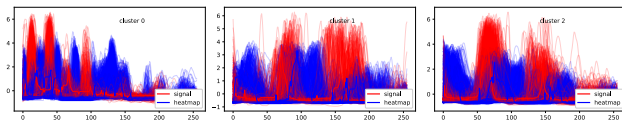


FIGURE 3. Multivariate clusters *InsectWingbeatSound* (c_0).

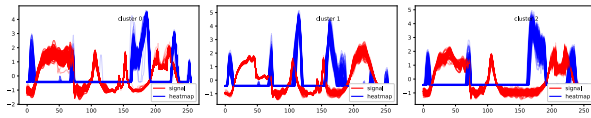


FIGURE 4. Multivariate clusters *Mallat* (c_0).

The saliency maps can be regarded as a relevance guide. Thus, the clustering is based on both the shape (via the signal) and the temporal relevance (via the saliency map). It is important to note that clustering with input samples in isolation is not merely clustering of input samples, because it uses the information of the binary predictions – it is a kind of informed clustering of input samples. Adding the saliency maps as a second channel results in an even more informed clustering. Another crucial aspect is that the saliency maps are obviously only required in relatively hard to distinguish cases, where additional information is needed. Therefore, the approach assumes a certain level of difficulty in order to be effective. In datasets with easily distinguishable classes, it may be less beneficial, but such datasets are also not very problematic.

G. LLM-BASED SYMBOLIC APPROXIMATION AND FUZZY KG MATCHING

The aim of this section is to associate the established cluster centroids (barycenters, i.e., “average” time series computed using DBA) with symbolic knowledge. The identified subclasses are matched to instances in an existing KG to retrieve corresponding semantic information. This constitutes a fusion of subsymbolic and symbolic representations, where learned patterns in sensor signals are mapped to interpretable concepts from a KG. It can be regarded as probabilistic semantics, i.e., LLM-based approximation

of semantics and subsequent fuzzy matching of these approximations to symbolic properties. The procedure begins with constructing a KG containing symbolic attributes for each class (cf. Sec. III-A). Subsequently, an LLM is provided with an introductory context prompt, as well as the cluster centroids and generates textual features that can be matched to KG entries (cf. Fig. 1). Accordingly, the LLM plays a dual role: (1) translating subsymbolic patterns (e.g., centroids and saliency maps) into symbolic approximations, and (2) performing fuzzy matching against predefined symbolic properties of the KG. The core pattern discovery occurs at the subsymbolic level. The LLM serves as a bridge by assigning names and contextual information to cluster centroids, thereby facilitating knowledge discovery. All experiments indicate that matching is effective using cluster centroids, despite them not being actual samples. In the multivariate case, saliency maps are likewise represented by centroids. If a centroid is successfully matched, all associated symbolic contextual information is available. Otherwise, a previously unknown pattern is discovered that should be investigated by a domain expert. For expert verification, saliency maps can again be very valuable, since they help domain experts to judge whether what the model learned is reasonable. Essentially, it is a form of supporting evidence.

Without conducting a large-scale comparison of different state-of-the-art LLMs, several *OpenAI* models have been compared (accessed via the *OpenAI* platform API) in terms of their ability of signal description, matching and the consistency in that task. A very clear winner in that was *o3-2025-04-16*, which, according to the documentation, is *OpenAI*’s “most powerful reasoning model” at the time of writing this paper. It has a context window of 200.000, a maximum output token number of 100.000, and it supports both image and text input. Nonetheless, it is noteworthy that the approach does not depend on this particular LLM model; it may also function with similarly well-suited open-weight models.

In order to obtain a structured process across the different datasets, the initial textual descriptions (**fault_desc**) of classes (**name**) that are part of the KG (cf. Sec. III-A) are also generated with the LLM (in the evaluation, not in practice) as a sort of approximated summarization of the signals. Obviously, they are not generated based on the clustering results, but based on the initial z -normalized dataset. Specifically, each class description is generated by forwarding the corresponding medoid (real sample that best represents the entire set for each class) to the LLM for description. Let S_c denote the set of all time series in the dataset that belong to class c :

$$\text{medoid}(S_c) = \arg \min_{x_i \in S_c} \sum_{x_j \in S_c} d(x_i, x_j)$$

The distance function d represents the pairwise *Dynamic Time Warping* (DTW). Initially, the symbolic descriptions of each class of a dataset were created manually based on corresponding medoids. However, in order to adopt a more

systematic approach without manual biases, we decided to provide the medoids of a dataset to an LLM that generates the descriptions. The used model is again *o3-2025-04-16*. This process involved quite a bit of prompt engineering, e.g., deciding on whether it is allowed to include raw values and information about positive and negative parts, or whether it should just be about the general shape. After conducting several experiments, it turned out to be beneficial to generally incentivize brevity and generality, i.e., to not include raw values. The performance deteriorated as the descriptions became more nuanced. The LLM focuses extensively on the exact specifics of the medoids then (overfitting). Two options were implemented for the manner in which the LLM receives the medoids (and centroids): as *base64*-coded images or as lists of floats. Since the performance was similar in both cases, we opted for the more direct approach of providing medoids as lists of floats. The exact prompting and extraction of symbolic approximations can be found in the open source repository.⁴

Based on the experiments conducted, the following general conclusions can be drawn: It is impractical to provide the LLM with entire clusters, as this merely adds input tokens and complexity. The centroids are ideal representations of the clusters' samples. Additionally, the saliency maps are not beneficial for the matching process. As before, keeping the prompt simple helps, not being overly specific. We avoid using raw values via prompts and rely on general shape descriptions. Based on the signal descriptions, the ontology is instantiated for the particular dataset, i.e., corresponding semantic facts are generated (or exist in practice). Afterwards, the LLM analysis takes place, i.e., the centroids (cf. Fig. 5, Fig. 6) are forwarded to the LLM. On the one hand, for symbolic approximation, and on the other hand, for potentially matching KG entries.

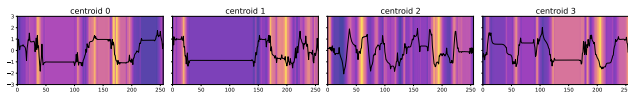


FIGURE 5. Centroids for *UWaveGestureLibraryAll* (c_1).

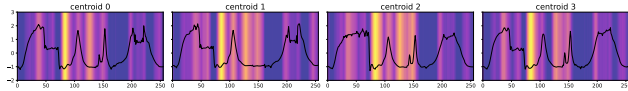


FIGURE 6. Centroids for *Mallat* (c_1).

It is important to note that, in this case, the centroids are multivariate, consisting of the signals themselves and the overlaid saliency maps (cf. Fig. 1). Nevertheless, the LLM only receives the signals for description and matching (signal centroid in Fig. 1), since it would be unrealistic to assume saliency map-like features in an existing KG. Practically, it is a lot more plausible to obtain some symbolic descriptions of, e.g., fault curves, and not particular highlighted segments. The shape of the final result follows a simple scheme,

e.g., $\langle \text{class_7}, \text{class_1}, \text{class_3} \rangle$. An exemplary instance of a full LLM response is provided in Appendix B. Thus, in this case, the LLM received three centroids and matched all of them to a particular instance (cf. **name** in Sec. III-A) in the KG. Ultimately, additional information for each class can be retrieved from the KG, e.g., the **severity** of a fault. In general, such a matching provides valuable context information. To consider the inherent uncertainty of the LLM-based description and matching and to study the stability of predictions, the process is repeated five times for each dataset (r_1 to r_5). The results are summarized in Table 2. Each cell indicates the predicted class for each forwarded cluster centroid, and the result column corresponds to a most prominent prediction for each. The ground truth column indicates the most prevalent ground truth label of the corresponding clusters. The last column counts the matched selected clusters, i.e., the identified subclasses, and the ground truth classes in brackets. Matching is only reasonable for the input samples in isolation and the multivariate case, not for the saliency maps in isolation.

IV. EVALUATION OF KNOWLEDGE (RE)DISCOVERY

This section is concerned with the empirical evaluation of the success of the knowledge (re)discovery process, i.e., the number of identified classes, the number of correctly matched samples, and the comparison of the saliency map-aided variant and the signal-only variant. A valid initial question is whether it is not possible to simply match the input signals to the KG entries. In the end, the signal dimension of the multivariate centroids is matched, not the saliency maps. Thus, in principle, it would be possible to only cluster the input signals. However, the clustering is significantly improved by incorporating the saliency maps as a cornerstone for the knowledge discovery. In other words, it is advantageous to consider not only the signals themselves, but also some information about ROIs (attention over time). The saliency maps provide additional knowledge, not only the signals themselves, but also where to look at in the signals, which seems to be a crucial information in this context. Likewise, the saliency maps in isolation only provide information regarding points in time that are deemed important; the multivariate case combines this information with the actual signal. The expectation should be that the heatmap-aided clustering is substantially superior to the clustering of the input samples alone across multiple datasets. Finally, for the purpose of evaluation, the most prominent ground truth label in each cluster is determined to be the assumed class. To demonstrate a successful saliency map-based knowledge discovery, two criteria must be fulfilled:

- 1) The multivariate (heatmap-aided) clustering must be substantially superior to the clustering of the input signals in isolation.
- 2) The number of identified subclasses (and matched signals) based on the multivariate clustering results must be substantially higher compared to the one of the signals alone.

⁴https://github.com/tbohne/saliency_kd

TABLE 2. Accumulated results for the LLM-based matching: multivariate, input.

dataset_class	r ₁	r ₂	r ₃	r ₄	r ₅	result	ground truth	identified
UWaveGestureLibraryAll_c0	< 1, 2, 5, 7 >	< 1, 2, 8, 7 >	< 3, 2, 8, 7 >	< 1, 2, 5, 7 >	< 3, 2, 1, 7 >	< 3, 2, 5, 7 >	< 3, 7, 5, 1 >	2/4(4)
UWaveGestureLibraryAll_c1	< 5, 1, 2, 8 >	< 4, 1, 3, 5 >	< 5, 1, 3, 4 >	< 5, 4, 3, 8 >	< 8, 1, 3, 1 >	< 5, 4, 3, 8 >	< 6, 4, 2, 8 >	2/4(4)
InsectWingbeatSound_c0	< 11, 1, 1, 5 >	< 2, 1, 5 >	< 11, 4, 5 >	< 11, 4, 5 >	< 11, 1, 8 >	< 11, 1, 5 >	< 11, 1, 7 >	2/3(6)
InsectWingbeatSound_c1	< 3, 2, 4 >	< 3, 2, 4 >	< 3, 2, 4 >	< 3, 2, 4 >	< 3, 2, 4 >	< 3, 2, 4 >	< 4, 10, 8 >	0/3(5)
Mallat_c0	< 1, 4, 8 >	< 2, 4, 3 >	< 1, 4, 7 >	< 5, 4, 7 >	< 8, 1, 7 >	< 1, 4, 7 >	< 5, 1, 7 >	1/3(4)
Mallat_c1	< 5, 2, 3, 4 >	< 4, 2, 3, 4 >	< 8, 1, 3, 4 >	< 8, 2, 3, 4 >	< 2, 2, 3, 4 >	< 8, 2, 3, 4 >	< 8, 2, 6, 4 >	3/4(4)
UWaveGestureLibraryAll_c0	< 1, 2, 1, 1 >	< 1, 7, 1, 2 >	< 1, 7, 1, 1 >	< 1, 2, 1, 1 >	< 7, 2, 8, 1 >	< 1, 2, 8, 2 >	< 1, 7, 3, 5 >	1/4(4)
UWaveGestureLibraryAll_c1	< 4, 8, 7 >	< 4, 8, 7 >	< 4, 8, 7 >	< 4, 5, 7 >	< 4, 8, 7 >	< 4, 8, 7 >	< 4, 8, 2 >	2/3(4)
InsectWingbeatSound_c0	< 3, 4, 11 >	< 3, 4, 11 >	< 3, 4, 11 >	< 6, 4, 11 >	< 6, 4, 11 >	< 3, 4, 11 >	< 7, 1, 11 >	1/3(6)
InsectWingbeatSound_c1	< 11, 4, 2 >	< 6, 4, 2 >	< 6, 4, 2 >	< 6, 4, 2 >	< 2, 4, 2 >	< 6, 4, 2 >	< 10, 4, 6 >	1/3(5)
Mallat_c0	< 6, 2, 4 >	< 6, 2, 4 >	< 1, 8, 4 >	< 3, 8, 4 >	< 3, 8, 1 >	< 6, 8, 4 >	< 5, 7, 1 >	0/3(4)
Mallat_c1	< 6, 2, 2 >	< 6, 2, 2 >	< 6, 2, 2 >	< 6, 2, 2 >	< 6, 2, 2 >	< 6, 2, 2 >	< 6, 8, 2 >	2/3(4)

Thus, in general, it has to be demonstrated that saliency maps can play a crucial role in subclass identification. We begin with a threefold comparison of the clustering performances based on the three modes outlined in Sec. III-F. Let $X := \{X_1, X_2, \dots, X_r\}$ denote an established clustering and $Y := \{Y_1, Y_2, \dots, Y_s\}$ the ground truth clustering for the total number of samples n . As motivated in Section III, the ground truth clustering is given via the original multiclass labels. Each entry n_{ij} represents the number of common samples between X_i and Y_j , i.e., $n_{ij} = |X_i \cap Y_j|$. Also, $a_i = \sum_{j=1}^s n_{ij}$ and $b_j = \sum_{i=1}^r n_{ij}$.

The initially considered metric is the *Adjusted Rand Index* (ARI):

$$\text{ARI} := \frac{\sum_{ij} \binom{n_{ij}}{2} - \frac{\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}}{\binom{n}{2}}}{\frac{1}{2} \left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2} \right] - \frac{\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}}{\binom{n}{2}}} \in [-1, 1]$$

The second one is the *Normalized Mutual Information* (NMI), measuring homogeneity and completeness for Y and X :

$$\text{NMI}(Y, X) := \frac{2 \cdot I(Y, X)}{H(Y) + H(X)} \in [0, 1]$$

I measures the mutual information using the Shanon entropy H :

$$I(Y, X) := H(Y) - H(Y|X)$$

Finally, we also consider the purity P of clusters X and classes Y , i.e., the extent to which clusters contain a single class:

$$P := \frac{1}{n} \sum_{X_i \in X} \max_{Y_j \in Y} |X_i \cap Y_j| \in [0, 1]$$

Although the focus is on relative comparisons, a perfect score for all three metrics would be 1.0, indicating a perfect match between ground truth labels and clustering. The following three tables (Tab. 3, Tab. 4, Tab. 5) quantify the clustering performances for the UCR datasets under consideration. Given the heavy dependence of many clustering approaches on initialization, the robustness of the results is studied and the claims are strengthened by performing the clustering process three times with different seeds for each dataset, class, and mode. The mean performance μ and standard deviation σ are reported for each. The $k(\text{gt})$ column refers to

TABLE 3. Clustering performance *UWaveGestureLibraryAll* (μ mean, σ standard deviation).

run	mode_class	ARI	NMI	P	k(gt)	distribution
1	input_c0	0.426	0.476	0.712	4(4)	[434, 630, 400, 307]
2	input_c0	0.479	0.516	0.739	4(4)	[304, 398, 442, 627]
3	input_c0	0.438	0.492	0.715	4(4)	[651, 299, 411, 410]
μ	input_c0	0.448	0.495	0.722		
σ	input_c0	± 0.028	± 0.020	± 0.015		
1	input_c1	0.511	0.520	0.700	3(4)	[591, 672, 500]
2	input_c1	0.507	0.524	0.699	3(4)	[486, 611, 666]
3	input_c1	0.492	0.528	0.695	4(4)	[472, 643, 315, 333]
μ	input_c1	0.503	0.524	0.698		
σ	input_c1	± 0.010	± 0.004	± 0.003		
1	saliency_c0	0.106	0.107	0.421	3(4)	[644, 670, 457]
2	saliency_c0	0.108	0.109	0.425	3(4)	[616, 690, 465]
3	saliency_c0	0.101	0.101	0.418	3(4)	[638, 466, 667]
μ	saliency_c0	0.105	0.106	0.421		
σ	saliency_c0	± 0.004	± 0.004	± 0.004		
1	saliency_c1	0.530	0.516	0.759	4(4)	[435, 499, 337, 492]
2	saliency_c1	0.445	0.442	0.646	3(4)	[554, 548, 661]
3	saliency_c1	0.530	0.502	0.763	4(4)	[446, 328, 513, 476]
μ	saliency_c1	0.502	0.487	0.723		
σ	saliency_c1	± 0.049	± 0.039	± 0.066		
1	multivariate_c0	0.615	0.580	0.823	4(4)	[359, 446, 434, 532]
2	multivariate_c0	0.509	0.506	0.769	4(4)	[380, 535, 523, 333]
3	multivariate_c0	0.343	0.352	0.586	3(4)	[527, 664, 580]
μ	multivariate_c0	0.489	0.479	0.726		
σ	multivariate_c0	± 0.137	± 0.116	± 0.124		
1	multivariate_c1	0.773	0.737	0.889	4(4)	[384, 471, 451, 457]
2	multivariate_c1	0.626	0.673	0.723	3(4)	[458, 756, 549]
3	multivariate_c1	0.638	0.693	0.736	3(4)	[773, 449, 541]
μ	multivariate_c1	0.679	0.701	0.783		
σ	multivariate_c1	± 0.082	± 0.033	± 0.092		

both the estimated k using the *Elbow* method and the ground truth number of classes. The final column presents the cluster distribution.

Table 3 shows that the multivariate mode clearly outperforms the clustering based on input samples alone according to every metric. Specifically, the mean performance of the multivariate mode leads to the best result in five out of six cases, with the following percentage improvements (the table shows absolute values): ARI (c_0 : 9.2%, c_1 : 35.0%), NMI (c_0 : -3.2%, c_1 : 33.8%), P (c_0 : 0.6%, c_1 : 12.2%). The only exception is NMI for c_0 , where the mean performance of the input mode is marginally better than the mean performance of the multivariate mode. This is primarily attributable to the third run of multivariate_c0, during which the k estimation was suboptimal. However, even for NMI and c_0 , the best result across all three runs is one from the multivariate mode (NMI of 0.58 for multivariate_c0, run 1). The univariate problem has a simpler optimization space and is more stable across runs, resulting in extremely low standard deviations for the input mode. Adding a further dimension increases the sensitivity to initialization, leading to standard deviations

TABLE 4. Clustering performance *InsectWingbeatSound* (μ mean, σ standard deviation).

run	mode_class	ARI	NMI	P	k(gt)	distribution
1	input_c0	0.018	0.064	0.233	3(6)	[526, 331, 193]
2	input_c0	0.013	0.060	0.229	3(6)	[321, 173, 556]
3	input_c0	0.011	0.056	0.220	3(6)	[340, 549, 161]
μ	input_c0	0.014	0.060	0.227		
σ	input_c0	± 0.004	± 0.004	± 0.007		
1	input_c1	0.007	0.019	0.226	3(5)	[325, 245, 360]
2	input_c1	0.010	0.025	0.228	3(5)	[302, 277, 351]
3	input_c1	0.009	0.023	0.230	3(5)	[352, 341, 237]
μ	input_c1	0.009	0.022	0.228		
σ	input_c1	± 0.002	± 0.003	± 0.002		
1	saliency_c0	0.165	0.274	0.358	3(6)	[272, 525, 253]
2	saliency_c0	0.147	0.234	0.350	3(6)	[207, 484, 359]
3	saliency_c0	0.171	0.277	0.388	4(6)	[221, 313, 368, 148]
μ	saliency_c0	0.161	0.262	0.365		
σ	saliency_c0	± 0.012	± 0.024	± 0.020		
1	saliency_c1	0.118	0.187	0.309	3(5)	[352, 309, 269]
2	saliency_c1	0.134	0.229	0.333	3(5)	[402, 191, 337]
3	saliency_c1	0.139	0.212	0.317	3(5)	[350, 228, 352]
μ	saliency_c1	0.130	0.209	0.320		
σ	saliency_c1	± 0.011	± 0.021	± 0.012		
1	multivariate_c0	0.233	0.368	0.398	3(6)	[307, 281, 462]
2	multivariate_c0	0.287	0.410	0.468	4(6)	[312, 265, 188, 285]
3	multivariate_c0	0.304	0.453	0.497	5(6)	[290, 185, 225, 221, 129]
μ	multivariate_c0	0.275	0.410	0.454		
σ	multivariate_c0	± 0.037	± 0.043	± 0.051		
1	multivariate_c1	0.169	0.248	0.326	3(5)	[254, 303, 373]
2	multivariate_c1	0.208	0.314	0.356	3(5)	[319, 221, 390]
3	multivariate_c1	0.205	0.300	0.358	3(5)	[303, 230, 397]
μ	multivariate_c1	0.194	0.287	0.347		
σ	multivariate_c1	± 0.022	± 0.035	± 0.018		

that are very low to moderate for the multivariate mode. k estimation works equally well for the input-only and multivariate modes in the considered experiments for this dataset. Interestingly, for c_1 , the saliency maps in isolation even lead to comparable mean performances to the input samples in isolation, with the mean purity being even better. Overall, the multivariate case yields slight performance increases for c_0 and major increases for c_1 . For c_1 , it robustly promotes an initial clustering of moderate quality to a satisfactory one.

Table 4 confirms that the multivariate mode substantially outperforms the clustering based on input samples alone. Once more, in accordance with every metric, and this time also in every case, with the following mean improvements: ARI (c_0 : 1864.3%, c_1 : 2055.6%), NMI (c_0 : 583.3%, c_1 : 1204.5%), P (c_0 : 100%, c_1 : 52.2%). For this dataset, pure saliency maps even lead to a considerably better performance than pure input samples in every case. Moreover, k estimation is greatly improved in the multivariate mode for c_0 and the standard deviations are very low across all runs and modes, indicating robust results. In the end, the saliency map-guided clustering transforms an infeasibly poor clustering into a fairly moderate one. Once again, it is not about absolute results, which can be further optimized in future work; it is about the improvement, which is substantial and consistent. The relative improvement is exceptionally high in this case due to the dataset being challenging to classify and the baseline yielding unsatisfactory results.

Finally, Table 5 is a further compelling case, as it is a dataset that is already quite well-solved with the baseline approach. Consequently, there is limited potential for enhancement. First, for c_0 , all metrics are equal to negligible deviations in both the input-only and multivariate cases. Thus, there is no improvement for c_0 . This can be partially attributed to the fact that the dataset is a priori quite

TABLE 5. Clustering performance *Mallat* (μ mean, σ standard deviation).

run	mode_class	ARI	NMI	P	k(gt)	distribution
1	input_c0	0.651	0.807	0.713	3(4)	[608, 290, 251]
2	input_c0	0.651	0.807	0.713	3(4)	[608, 251, 290]
3	input_c0	0.651	0.807	0.713	3(4)	[251, 608, 290]
μ	input_c0	0.651	0.807	0.713		
σ	input_c0	± 0.000	± 0.000	± 0.000		
1	input_c1	0.634	0.803	0.721	3(4)	[278, 625, 293]
2	input_c1	0.634	0.803	0.721	3(4)	[625, 293, 278]
3	input_c1	0.634	0.803	0.721	3(4)	[293, 625, 278]
μ	input_c1	0.634	0.803	0.721		
σ	input_c1	± 0.000	± 0.000	± 0.000		
1	saliency_c0	0.585	0.681	0.702	3(4)	[290, 536, 323]
2	saliency_c0	0.583	0.679	0.701	3(4)	[536, 291, 322]
3	saliency_c0	0.599	0.703	0.707	3(4)	[544, 291, 314]
μ	saliency_c0	0.589	0.688	0.703		
σ	saliency_c0	± 0.009	± 0.013	± 0.003		
1	saliency_c1	0.901	0.909	0.934	4(4)	[269, 338, 292, 297]
2	saliency_c1	0.903	0.913	0.935	4(4)	[338, 293, 297, 268]
3	saliency_c1	0.899	0.907	0.933	4(4)	[292, 339, 268, 297]
μ	saliency_c1	0.901	0.910	0.934		
σ	saliency_c1	± 0.002	± 0.003	± 0.001		
1	multivariate_c0	0.649	0.804	0.713	3(4)	[610, 249, 290]
2	multivariate_c0	0.649	0.804	0.713	3(4)	[290, 249, 610]
3	multivariate_c0	0.649	0.804	0.713	3(4)	[610, 290, 249]
μ	multivariate_c0	0.649	0.804	0.713		
σ	multivariate_c0	± 0.000	± 0.000	± 0.000		
1	multivariate_c1	0.918	0.931	0.940	4(4)	[300, 293, 278, 325]
2	multivariate_c1	0.911	0.924	0.936	4(4)	[304, 293, 278, 321]
3	multivariate_c1	0.911	0.924	0.936	4(4)	[293, 278, 321, 304]
μ	multivariate_c1	0.913	0.926	0.937		
σ	multivariate_c1	± 0.004	± 0.004	± 0.002		

well-solved. In certain instances, the inherent structure may render additional information redundant, thereby diminishing the contribution effect. However, there is a substantial and consistent mean enhancement for c_1 across all metrics: ARI (c_1 : 44.0%), NMI (c_1 : 15.3%), P (c_1 : 30.0%). Additionally, the k estimation is enhanced and aligns with the ground truth in every run. In this case, the saliency maps appear highly advantageous, which is reinforced by the observation that the saliency maps in isolation considerably outperform the signals in isolation for c_1 . Overall, we once again observe a very clear and consistent contribution of the saliency maps, transforming an initially adequate clustering into an almost perfect one. The remarkably low standard deviations indicate highly robust results. Ultimately, it is confirmed that the multivariate clustering substantially outperforms the clustering of the signals in isolation across all considered datasets (1), i.e., the saliency maps do have a positive impact. Despite their occasional utility in isolation, the saliency maps, i.e., ROIs, only truly contribute in conjunction with the signals as context.

Another aspect to be quantified are the subclass predictions introduced in Section III-G. The last column in Table 2 counts the matched selected clusters, i.e., how many classes are discovered, and the ground truth classes in brackets. A class is considered discovered if the dominant ground truth class within the cluster corresponds to the LLM prediction. As can be seen in the last column, the saliency map-aided approach wins in five out of six cases. Thus, it not only improves the clustering results, but those improvements also consistently lead to increased numbers of finally identified subclasses. The sole exception to this is the case of *InsectWingbeatSound_c1*, in which the multivariate case does not result in the identification of any subclasses. This case led to clusters that were very stably predicted wrong by the LLM. This one exception aligns perfectly with the metrics displayed in Table 4. The multivariate approach substantially

improves the clusterings; nonetheless, it remains the most challenging dataset, which also manifests in the final LLM matching. Ultimately, on a more fine-grained level, we are also interested in the fraction of ground truth samples of a class that are matched to a correctly predicted class for each dataset, i.e., the precision (proportion of dominant class in cluster in brackets):

- *UWaveGestureLibraryAll* (**input**):
 $\mathbf{c}_0: \{1 : \frac{321}{437}(\frac{321}{434})\}, \mathbf{c}_1: \{4 : \frac{377}{450}(\frac{377}{591}), 8 : \frac{418}{460}(\frac{418}{672})\}$
- *UWaveGestureLibraryAll* (**multivariate**):
 $\mathbf{c}_0: \{3 : \frac{304}{454}(\frac{359}{304}), 5 : \frac{385}{433}(\frac{385}{434})\}, \mathbf{c}_1: \{4 : \frac{367}{450}(\frac{367}{471}), 8 : \frac{431}{460}(\frac{431}{457})\}$
- *InsectWingbeatSound* (**input**):
 $\mathbf{c}_0: \{11 : \frac{66}{180}(\frac{66}{193})\}, \mathbf{c}_1: \{4 : \frac{58}{180}(\frac{58}{245})\}$
- *InsectWingbeatSound* (**multivariate**):
 $\mathbf{c}_0: \{11 : \frac{152}{180}(\frac{152}{307}), 1 : \frac{121}{180}(\frac{121}{281})\}, \mathbf{c}_1: \emptyset$
- *Mallat* (**input**): $\mathbf{c}_0: \emptyset, \mathbf{c}_1: \{6 : \frac{278}{294}(\frac{278}{278}), 2 : \frac{292}{292}(\frac{292}{293})\}$
- *Mallat* (**multivariate**):
 $\mathbf{c}_0: \{7 : \frac{290}{291}(\frac{290}{290})\}, \mathbf{c}_1: \{8 : \frac{292}{293}(\frac{292}{300}), 2 : \frac{292}{292}(\frac{292}{293}), 4 : \frac{262}{289}(\frac{262}{325})\}$

As demonstrated in this analysis, the multivariate scenario is not only discovering more subclasses, but also consistently leading to better precision in cases where both scenarios find the subclass, e.g., subclass 11 in the *InsectWingbeatSound* dataset with more than twice as many matched samples. The mean sample coverage across all identified subclasses and datasets is 0.73 for the input-only case and 0.87 for the multivariate case. Consequently, success criterion (2) is also satisfied. Finally, for practical purposes, one might be interested in evaluations without ground truth labels, e.g., for initial filtering. Without knowing ground truth labels of signals allocated to a given cluster, alternative (uninformed) quantitative evaluation metrics to estimate the quality of a cluster are needed (cf. Appendix A).

In the end, we offer some reflections on the sensitivity of the results to the label subsumption scheme. One could assume that the empirical conclusions depend on the specific label subsumption construction used (cf. Sec. III-C). Therefore, it is of interest whether similar results would hold under different ways of grouping the original classes. Under different groupings of the original classes, i.e., less challenging ones, the expected results are similar or, most likely, better due to additional structure. We started with a label subsumption scheme that was not entirely random but rather utilized some prior information about the proximity of certain classes in the original multiclass problem. After shifting from an informed label subsumption towards one without using any prior knowledge, the first half of the classes was assigned to one class of the binary problem and the second half to the other. However, there may be an implicit logic to the order of the original classes, so assigning them in order could facilitate that and simplify the problem. Thus, to avoid the use of any prior knowledge, they are finally assigned randomly in an alternating fashion. Although the two schemes yielded only marginal differences in the results (the conclusions were the same), random assignment using

an alternating scheme is the more technically sound option. Of course, since the rule is deterministic, there can be one type of counterexample dataset that aligns perfectly with the subsumption scheme and accidentally creates a meaningful grouping. Oftentimes, there are several dimensions of similarity between classes, e.g., gender and species in *InsectWingbeatSound*, and a subset of those can occasionally be reproduced by any subsumption scheme. While such an example that accidentally recovers the structurally crucial dimension in the original class distribution may represent a rather unlikely edge case for a single dataset, it is extremely unlikely across different datasets with differing semantics and numbers of classes. Hence, it is random for all practical purposes, especially across different datasets.

V. CONCLUSION

Experimental results on well-established time series classification datasets (UCR) demonstrate the effectiveness of our saliency map-driven method for knowledge discovery, substantially outperforming signal-only baselines in both clustering and subclass identification. In Sec. IV, we define two criteria for success, both of which are satisfied. It has been confirmed that the multivariate heatmap-aided clustering is substantially superior to the clustering of the signals in isolation across all considered datasets (1), and that these improvements also consistently lead to considerably increased numbers of finally identified subclasses and matched signals (2). Being able to judge the symbolic matching is one motivation for the end-to-end evaluation enabled through multiclass datasets and label subsumption. The pivotal role of the saliency maps lies in enhancing the quality of clustering, i.e., discriminating classes that are hardly distinguishable in shape. This, in turn, enhances the efficacy of the final LLM-based matching due to more precise cluster centroids that better reflect the underlying classes. Both factors are required: the shape (via the signal) and the temporal relevance (via the saliency map). Thus, we proved our initial hypothesis of saliency maps having a beneficial effect in knowledge (subclass) discovery – despite the fact that state-of-the-art saliency map generation techniques are suboptimal and have certain weaknesses with regard to consistency and robustness when applied to time series data, as argued in [6]. The precise semantics and dynamics of time series are not utilized to their full potential by current saliency map generation techniques. Consequently, enhancements to the techniques could also lead to further enhancements in saliency map-guided knowledge discovery. Finally, all developed software, i.e., every module that contributes to the knowledge discovery pipeline or its evaluation (cf. Sec. III), is released as open source software⁵ to facilitate reproducibility and domain transfer.

There are several natural next steps, e.g., investigating the selection of the number of classes in label subsumption based on a preliminary analysis of several groups, i.e., fewer than the original number, but potentially more than two.

⁵https://github.com/tbohne/saliency_kd

In addition, it could be reasonable to separate hard-to-distinguish subclasses during label subsumption, thereby forcing the model to learn the nuances. If the separation of the two classes is too straightforward, it may result in non-accurate clusters due to the fact that tiny features are simply not required to be learned. Moreover, it might be worthwhile to compare different saliency map generation techniques and evaluate their respective advantages and disadvantages in terms of knowledge discovery. It is particularly promising because this work successfully demonstrates the potential based on vanilla *Grad-CAM*. More sophisticated attribution techniques are expected to further improve the results. In future work, it would also be sensible to investigate other methods of signal summarization, such as characteristic feature extraction via FFT, statistical features, etc. Also, the LLM-based matching could be replaced by fuzzy text matching, cosine similarity over embeddings, or even rule-based matchings. Beyond the replacement of the LLM with alternative techniques, a useful avenue would be the comparison of various LLMs and the analysis of their respective outcomes with respect to signal description and matching. Another extension could be the filtering of established clusters. Not all established clusters must be forwarded to the LLM for symbolic approximation; only those that satisfy certain quantifiable requirements. A filtering algorithm could preselect suited clusters based on quantifying metrics without ground truth labels. Investigating limiting factors for the approach would also be valuable future work. While the dataset selection criteria in this work are very natural and do provide a guideline for other domains and scenarios, it could be of interest to broaden the spectrum and, e.g., evaluate the approach with image data, significantly smaller datasets, or the influence of different classifier accuracies. Saliency maps are more frequently applied to image data, facilitating a seamless transition to image classification tasks. Nevertheless, the dataset requirements are transparent, and there are arguments justifying the selection of each parameter. Eventually, the aim of the empirical evaluation in this work is to show that saliency maps have potential for knowledge discovery, which is a novel way of using them. In future work, it would be reasonable to further study this potential relative to other subclass discovery mechanisms. To conclude, the evaluation on UCR datasets with a controlled rediscovery setup is meaningful, but it would be worthwhile to pair it with a fully open-ended knowledge discovery scenario in a real-world application in future work. We are currently conducting a case study involving such practical discoveries in an industrial domain. Although there is no fundamental difference, the challenging subtleties of less controlled real-world dynamics are always a factor to consider.

APPENDIX A UNINFORMED QUANTITATIVE EVALUATION

For practical purposes, one might be interested in evaluations without ground truth labels, e.g., for initial filtering.

```

response...
resp 6883752e775c819bb5d8b192e7834a1c01bfdb4277b11938
03-2025-04-16
Signal 1
The curve opens a touch below zero and drifts to its shallowest dip at about
index 5. From there it turns upward, crosses the axis and forms an early
rounded hill near index 30; without pausing it keeps climbing into a broader,
taller wave whose smooth crown-and-overall maximum-lies around index 37-40.
An abrupt fall then takes the trace into a narrow trough centred near index 47,
after which it oscillates in low-amplitude ripples, finally sliding through zero
and plunging to its deepest negative pocket at roughly index 74. It pulls out of
that hole, spends the middle third hovering close to the axis with small jitter,
and, during the last quarter, builds one final but distinctly smaller crest near
index 221 before easing back into negative values toward the end.
Best matching class: class_2

Signal 2
Beginning moderately negative, the signal sags a little further to a minimum
close to index 7, then mounts a steady ascent that carries it through zero.
It reaches a first shoulder near index 30 and, a few samples later, swells into
a higher, broad crest whose tip-and global peak-sits around index 44. A sharp
drop follows, bottoming in a valley at about index 47; the next long stretch is
spent in low-level fluctuations that become mildly negative through the third
quarter. Approaching the finish it gathers itself into one last rise, topping
out in a secondary but smaller summit near index 222, after which it slides down
again into the negative zone.
Best matching class: class_2

Signal 3
From an already negative start the trace dips only briefly (minimum near index
7) before launching into a long, smooth climb that carries it through the origin
and spreads into a broad high region. This zone culminates in a clear, overall
maximum close to index 61. The waveform then steps downward through a few
ripples and tumbles into its deepest trough at roughly index 74. A modest
rebound follows, after which the series wanders in gentle undulations, finally
edging upward into a subdued rise that stalls near index 210 and then tails away
with small oscillations on either side of zero.
Best matching class: class_6

Signal 4
The waveform starts negative and continues to fall until about index 7, then
turns upward, crosses zero and produces its first pronounced summit near index
37. After a brief dip it throws up a smaller secondary bump at roughly index 44,
then dives steeply to its greatest negative excursion near index 74. It recovers
through a noisy shoulder, moves sideways for a while, and ultimately gathers
into a sustained late-series surge whose crest-matching or just topping the
early high-occurs around index 214-220. From that plateau it descends gradually
with light ripples, finishing a little below the axis.
Best matching class: class_4

class_2,class_2,class_6,class_4
1.0
ResponseUsage(input_tokens=5219, input_tokens_details=
InputTokensDetails(cached_tokens=0), output_tokens=7996, output_tokens_details=
OutputTokensDetails(reasoning_tokens=7360), total_tokens=13215)
pred class: class_2,class_2,class_6,class_4

```

FIGURE 7. Exemplary LLM-based symbolic approximation and KG matching.

Without knowing ground truth labels of signals allocated to a given cluster, we need alternative (uninformed) quantitative evaluation metrics to estimate the quality of a cluster $X_j \in X$ with centroid C_j :

- $DTW_{intra} := \frac{1}{n_j} \sum_{i=1}^{n_j} d(x_i^j, C_j), X_j \in X, x_i^j \in X_j$
- $DTW_{inter} := \frac{1}{r-1} \sum_{k=1, k \neq j}^r d(C_j, C_k) \text{ for } X_j \in X$
- $DTW_{frac} := \frac{DTW_{intra}}{DTW_{inter}}$
- S_{DTW} : *Silhouette* score based on DTW distance
- cluster size entropy $\eta := -\frac{1}{\log r} \sum_{j=1}^r \frac{n_j}{N} \log \frac{n_j}{N}, N = \sum_{j=1}^r n_j$
- $\sigma_s^2 := \frac{1}{n_j T} \sum_{i=1}^{n_j} \sum_{t=1}^T (x_{it}^j - \frac{1}{n_j} \sum_{k=1}^{n_j} x_{kt}^j)^2,$
 $T = |x_i^j|, x_i^j \in X_j$
- $\sigma_i^2 := \frac{1}{n_j T} \sum_{i=1}^{n_j} \sum_{t=1}^T (x_{it}^j - c_t^j)^2,$
 $T = |x_i^j|, x_i^j \in X_j, c_t^j \in C_j$

With the exception of η and S_{DTW} , all metrics can be applied to either the individual clusters or the overall clustering outcome via averaging. We are interested in whether these uninformed metrics can serve as indicators of ground truth performance and therefore calculate the correlation between the two. There are medium to strong correlations with NMI for all metrics except for η . The strongest correlation ($\rho = -0.81$) is observed with σ_s^2 . Further medium to strong correlations with NMI exist for σ_i^2 ($\rho = -0.77$), S_{DTW} ($\rho = 0.72$) and DTW_{frac} ($\rho = -0.72$), with corresponding p -values < 0.01 . Medium correlations of these metrics to ARI and P can also be found; however, these are slightly weaker, and partially insignificant.

Therefore, in general, the metrics do correlate with the ground truth evaluations and are as such good predictors. However, an examination of the correlations between these metrics and the extent to which the centroid of a cluster is correctly matched by the LLM suggests the need for more sophisticated filtering algorithms, e.g., the accumulation of multiple metrics for the purpose of cluster evaluation.

APPENDIX B EXEMPLARY LLM RESPONSE

Cf. Fig. 7

REFERENCES

- [1] T. Bohne, A.-K. P. Windler, and M. Atzmueller, "A neuro-symbolic approach for anomaly detection and complex fault diagnosis exemplified in the automotive domain," in *Proc. 12th Knowl. Capture Conf.*, Dec. 2023, pp. 35–43.
- [2] T. Bohne, A.-K. P. Windler, and M. Atzmueller, "A domain-agnostic neuro-symbolic architecture for multimodal human-in-the-loop anomaly detection and complex fault diagnosis," *IEEE Access*, vol. 13, pp. 210201–210236, 2025.
- [3] M. Hossain, R. Islam, F. Ahamed, M. Ahsan, and J. Haider, "A collaborative federated learning framework for lung and colon cancer classifications," *Technologies*, vol. 12, no. 9, p. 151, Sep. 2024.
- [4] D. Mahapatra, Z. Ge, and M. Reyes, "Self-supervised generalized zero shot learning for medical image classification using novel interpretable saliency maps," *IEEE Trans. Med. Imag.*, vol. 41, no. 9, pp. 2443–2456, Sep. 2022.
- [5] H. Kautz, "The third AI summer: AAAI Robert S. Englemore memorial lecture," *AI Mag.*, vol. 43, no. 1, pp. 93–104, Mar. 2022.
- [6] C. Balestra, B. Li, and E. Müller, "On the consistency and robustness of saliency explanations for time series classification," 2023, *arXiv:2309.01457*.
- [7] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.
- [8] A. Chattopadhyay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 839–847.
- [9] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu, "Score-CAM: Score-weighted visual explanations for convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 111–119.
- [10] P.-T. Jiang, C.-B. Zhang, Q. Hou, M.-M. Cheng, and Y. Wei, "LayerCAM: Exploring hierarchical class activation maps for localization," *IEEE Trans. Image Process.*, vol. 30, pp. 5875–5888, 2021.
- [11] F. Petitjean, A. Ketterlin, and P. Gançarski, "A global averaging method for dynamic time warping, with applications to clustering," *Pattern Recognit.*, vol. 44, no. 3, pp. 678–693, Mar. 2011.
- [12] E. Fons, R. Kaur, S. Palande, Z. Zeng, T. Balch, M. Veloso, and S. Vyatrenko, "Evaluating large language models on time series feature understanding: A comprehensive taxonomy and benchmark," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2024, pp. 21598–21634.
- [13] Z. Xie, Z. Li, X. He, L. Xu, X. Wen, T. Zhang, J. Chen, R. Shi, and D. Pei, "ChatTS: Aligning time series with LLMs via synthetic data for enhanced understanding and reasoning," 2024, *arXiv:2412.03104*.
- [14] W. An, F. Tian, P. Chen, S. Tang, Q. Zheng, and Q. Wang, "Fine-grained category discovery under coarse-grained supervision with hierarchical weighted self-contrastive learning," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2022, pp. 1314–1323.
- [15] W. An, F. Tian, W. Shi, Y. Chen, Q. Zheng, Q. Wang, and P. Chen, "DNA: Denoised neighborhood aggregation for fine-grained category discovery," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2023, pp. 12292–12302.
- [16] M. Grcic, A. Gadetsky, and M. Brbic, "Fine-grained classes and how to find them," in *Proc. ICML*, 2024, pp. 16275–16294.
- [17] K. Fan, Y. Zhao, D. Li, C. Lin, W. Zhang, and Z. Zhong, "LLM-enhanced generalized category discovery via iterative graph diffusion," in *Proc. 34th ACM Int. Conf. Inf. Knowl. Manage.*, Nov. 2025, pp. 645–654.
- [18] H. P. Zou, S. Singh, Y. Nian, J. He, J. Cai, S. Mansour, and H. Su, "Active generalized category discovery with diverse LLM feedback," in *Proc. 19th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2026, pp. 7637–7658.
- [19] A. Banerjee and S. Biswas, "Language-assisted feature representation and lightweight active learning for on-the-fly category discovery," *Trans. Mach. Learn. Res.*, Sep. 2025.
- [20] H. A. Dau, A. Bagnall, K. Kamgar, C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, and E. Keogh, "The UCR time series archive," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 6, pp. 1293–1305, Nov. 2019.
- [21] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Müller, "Deep learning for time series classification: A review," *Data Mining Knowl. Discovery*, vol. 33, pp. 917–963, May 2018.
- [22] Y. Chen, A. Why, G. Batista, A. Mafra-Neto, and E. Keogh, "Flying insect classification with inexpensive sensors," *J. Insect Behav.*, vol. 27, no. 5, pp. 657–677, Sep. 2014.
- [23] J. Liu, L. Zhong, J. Wickramasuriya, and V. Vasudevan, "uWave: Accelerometer-based personalized gesture recognition and its applications," in *Proc. PerCom*, Jun. 2009, pp. 657–675.
- [24] S. Mallat, *A Wavelet Tour of Signal Processing*. Amsterdam, The Netherlands: Elsevier, 1999.
- [25] K. Fauvel, T. Lin, V. Masson, É. Fromont, and A. Termier, "XCM: An explainable convolutional neural network for multivariate time series classification," *Mathematics*, vol. 9, no. 23, p. 3137, Dec. 2021.

TIM BOHNE received the B.Sc. and M.Sc. degrees in computer science from Osnabrück University, Germany, in 2019 and 2022, respectively, where he is currently pursuing the Ph.D. degree with the Semantic Information Systems Group. From 2017 to 2022, he worked as a Research Assistant at the German Research Center for Artificial Intelligence (Department of Plan-Based Robot Control). From 2019 to 2020, he was employed as a Research Assistant at the Combinatorial Optimization Group, Osnabrück University. Since 2022, he has been a Researcher at the German Research Center for Artificial Intelligence (Department of Cooperative and Autonomous Systems). He leads research projects focused on automated fault diagnosis and decision support in industrial domains, with an emphasis on sustainability. His research interests center around neuro-symbolic computing, with a particular focus on neuro-symbolic methods for anomaly detection and fault diagnosis.

ANNE-KATHRIN PATRICIA WINDLER received the B.Sc. and M.Sc. degrees in cognitive science from Osnabrück University, in 2018 and 2022, respectively. She has been a Researcher at the Cooperative and Autonomous Systems Research Department, German Research Center for Artificial Intelligence (DFKI), Osnabrück, since 2022. Her research interests include deep learning, time series classification, data augmentation, neuro-symbolic diagnosis, and interpretable AI.

MARTIN ATZMUELLER received the M.Sc. (Diplom-Informatiker University) degree in computer science and the Ph.D. (Dr. rer. nat.) degree in computer science from the University of Würzburg, Germany, in 2002 and 2006, respectively, and the Habilitation (Dr. habil.) degree in computer science from the University of Kassel, Germany, in 2013.

From 2010 to 2017, he was a Senior Researcher at the University of Kassel. Subsequently, he held positions as an Associate Professor at Tilburg University, The Netherlands, and as a Visiting Professor at Université Sorbonne Paris Nord, France. In 2021, he was appointed as a Full Professor at Osnabrück University, Germany. He is currently a Full Professor with the Institute of Computer Science, Osnabrück University, where he heads the Semantic Information Systems Group. He is also the Scientific Director of the Research Department Cooperative and Autonomous Systems, German Research Center for Artificial Intelligence (DFKI), Osnabrück, as well as the Founding Spokesperson of the Joint Laboratory on Artificial Intelligence and Data Science and a member of the Research Center Data Science, Osnabrück University. His research interests include artificial intelligence (AI), data science, and integrative AI systems, with a particular focus on modeling complex data, explainable AI, interpretability, machine perception, and semantic modeling. This also relates to applications in trusted AI systems design, in particular, relating to robot control and integrative sensor-based AI systems.

• • •