



Relationship Discovery for Heterogeneous Time Series Integration: A Comparative Analysis for Industrial and Building Data

Lucas Weber ¹ and Richard Lenz ¹

Abstract: Cyber-physical systems like buildings and power plants are monitored with ever-increasing numbers of sensors, gathering massive and heterogeneous time-series datasets collected in data lakes. Appropriate meta-data, describing both the function and location of each sensor, is essential for any profitable use of the data but is often not available or incomplete. While various approaches exist for meta-data extraction from relational databases, the unique characteristics of heterogeneous time-series data necessitate specialized algorithms. Among the general algorithms developed for time-series meta-data inference, only a few are concerned with relationship discovery despite the critical importance of this information in many meta-data formats.



In contrast to time series integration, other fields of research offer a variety of measures for relationship discovery in homogeneous time-series collections. In this paper, we aim to leverage this knowledge for heterogeneous time-series data integration. We consolidate over 40 different measures and evaluate their performance on seven datasets from different industrial facilities to extract promising relationship measures and show that there are other better-performing candidates than the common Pearson Correlation Coefficient.

Keywords: Relationship Inference, Time Series Analysis, Data Integration, Data Profiling

1 Introduction

With the rise of the internet of things (IoT) and data-driven analysis of complex systems, there is an ever-increasing amount of time-series data [Ke23]. Experts can use this data to optimize buildings or industrial complexes like power plants [So20, WL23]. While these developments are of utter importance for cost, energy, and environmental efficiency, missing meta-data is named as a challenge across all approaches by the industry [WL23], researchers [BM24], and government agencies [So20]. Since manually adding semantic information is time-consuming and cost-intensive, automatic approaches are becoming increasingly important. While some algorithms analyze existing meta-data [WSS21, A123], the information of the given labels is often incomplete. As a consequence, data-driven approaches, inferring information from the raw time series, have received growing attention.

The semantic information about sensors and their corresponding time series data can be encoded into different meta-data schemas that reach from proprietary naming

¹ Friedrich-Alexander-Universität Erlangen-Nürnberg, Computer Science 6, Martensstr. 3, 91058 Erlangen, Germany, lucas.weber@fau.de,  <https://orcid.org/0000-0002-6877-6935>;
richard.lenz@fau.de,  <https://orcid.org/0000-0003-1551-4824>

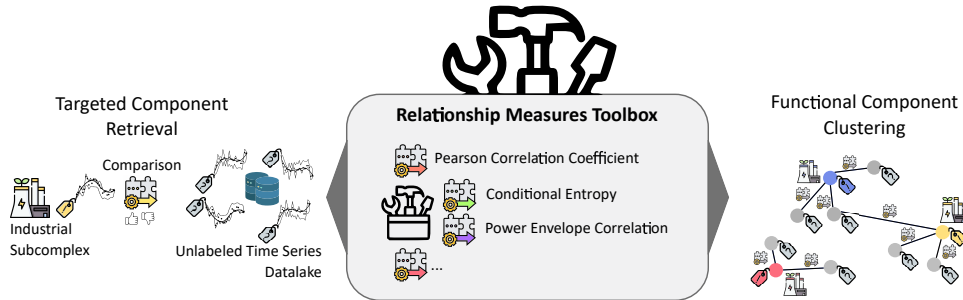


Fig. 1: Visualization of the usage of relationship measures. Centered is the available toolbox of different relationship measures from which we evaluate different measures for time series data integration. Left and right are the possible applications for a selected relationship measure (icons partially from flaticon.com).

conventions like the Kraftwerk-Kennzeichnungs-System (KKS) [VG18] to ontology representations using semantic web technologies [Ba18]. Regardless of the schema, the label always contains two different categories of information. On the one hand, there is information about the type of measurements or sensor, e.g., temperature, revolutions, pressure, system state. On the other hand, there is information about the association of sensors to different subsystems, e.g., all sensors in the kitchen or all measurements in the water-cooling cycle of a gas turbine.

Identifying the sensor type can be approached with many tools from pattern recognition and is treated widely in the related work [BM24]. In contrast, the inference of associations has received less attention in the context of building data integration, and fewer approaches have been published. Determining the absolute association of an individual sensor for previously unseen data by analyzing its time series in isolation is, in most cases, nearly impossible. To address this challenge, sensors are typically contextualized by grouping their time series based on a relationship measure [WL23]. If one sensor within the group can be identified as originating from a specific room or functional part of a power plant, that association can then be extended to the entire group.

The data-management community offers several ideas for relational data profiling and relationship discovery [AGN15, Ha23], but the approaches mainly consider relational data. This contrasts with the prevalence of time series in many data collection systems and leaves room for further developments. In contrast, relationship discovery for time series is very prominent in the fields of dynamical-systems analysis from physics, medicine, causal discovery, network/graph inference, and signal processing [BGBW19, Ru19, CI23, MMB23]. With this paper, we want to analyze how these approaches transfer from their original purpose to the problem of heterogeneous time-series data integration.

At this point, an example scenario might help illustrate our motivation. Unlike standardized products like cars, power plants are highly unique in their configuration, design, and use case. Moreover, these plants often operate for decades, undergoing extensions

and partial upgrades to meet evolving requirements. While plant operators focus on daily operations and economic management, they frequently rely on external experts to validate functionality, ensure regulatory compliance, and optimize performance. These experts, in turn, depend on time-series data to feed their analysis tools and models.

Consider an expert tasked with optimizing the efficiency of a combined-cycle power plant. Such plants generate electricity using gas and steam turbines, with each block typically comprising three turbines – two gas turbines and one steam turbine powered by waste heat from the gas turbines. The expert identifies reduced efficiency during full load and suspects an issue in the steam-generation pipeline, specifically within the Heat Recovery Steam Generator (HRSG) of block one. Following an HRSG upgrade, however, only the most essential control signals are properly labelled and integrated. Unfortunately, the time-series data needed for a detailed analysis remains unlabelled.

Locating these signals is challenging, given the plant's multiple blocks, numerous turbines, HRSG systems, and the variety of sensors (e.g., pressure, temperature, flow) within each HRSG. The expert must manually sift through the data, relying on domain knowledge and repeated pairwise comparisons with visual inspections – an effort-intensive process that consumes valuable time and resources. A toolbox equipped with robust relationship measures could significantly streamline this task (see Fig. 1 for a visualization). By selecting a labelled signal from the HRSG of interest, the toolbox could automatically retrieve signals of interest by estimating relatedness to the original query using different relationship measures. As the search for signals is only an intermediary step, the relationship measures must be simple and robust [WL23].

This paper aims to collect and transfer time-series relationship measures from their original application areas to the domain of heterogeneous time-series integration for building and industrial data. Relationship discovery in this field has received relatively little attention despite its significance in existing metadata schemas. Unlike traditional applications of these measures – typically used on time series from similar sensors – we focus on heterogeneous time-series collections. These collections consist of data gathered from diverse sensor setups with varying units of measurement.

Most of the methods we aim to evaluate, were not originally designed to address the challenges of heterogeneous, extensive, and noisy time-series data from buildings and industrial facilities. However, in a landscape where linear correlation is often used as a baseline for relationship discovery, we are particularly interested in whether methods from other fields can perform as well or better. Using a large collection of different measures, we provide insights into their performance across datasets from different industrial facilities and aim to identify the most promising measures for time-series data integration.

2 Related Work

2.1 Data Profiling for Meta-Data Extraction

The data-management community has been developing data-driven or query-driven approaches for data profiling and meta-data extraction for some time now [AGN15, Pa15, HL20, Hu19, La21, Ha23]. Almost all of these approaches are focused on inferring semantic information for relational data. While there are multi-column approaches for relationship discovery, such as frequent patterns [Pa15], approaches for heterogeneous time series are underrepresented in this area. This contrasts with the prevalence of time-series data in cyber-physical systems. We are motivated to close this gap by focusing our research on approaches suitable for extracting relationships in heterogeneous time-series collections.

2.2 Relationship Inference for Time-Series Integration

Several works regarding building data integration discuss the immediate necessity of automated approaches. Especially for Heating, Ventilation and Air Conditioning (HVAC), data integration and meta-data inference is a time-consuming but necessary step to improve the energy efficiency of large buildings [So20]. As buildings' heating and ventilation systems comprise different technical systems, most of the collected data has proprietary labels and no standard labeling format [Ch20]. This motivates many different approaches for time-series integration, where some focus on translating available meta-data, some on identifying sensor types, and others on inferring relationships between raw time series [BM24]. The sought-after relationships can be different in nature. They can be causal, like input-output-relations, but also correlations, e.g., the temperature and carbon dioxide, as well as the movement-sensor change when a person enters a room in a building; in parallel, the pressure shortly increases as the person closes the door. Therefore, we are not restricted to only causal relationships but aim to find all sensors that are attributed to functional groups. Our setting also differs from approaches from multivariate time-series analysis, where the time series are typically connected to the same system. In contrast, in our setting, that is not necessarily the case. Furthermore, the dimensionality in our case is extensive, often reaching hundreds of time series.

Correlation Analysis. Several papers rely on common correlation coefficients between time series to infer relationships and their association with functional subsystems. Koc et al. [KAB14], Park [PLA18], and Yu et al. [Yu22] rely primarily on the Pearson Correlation Coefficient (PCC) and Spearman Rank Correlation to infer relations between the sensors. They also allow time delays when measuring the associations by generalizing the PCC into the Cross-Correlation function. While the PCC has been the de-facto gold standard to measure relationships between time series in the field of meta-data extraction [CI23, BM24], this standard has never been questioned. Therefore, we also want to incorporate other measures for a better comparison.

Regression Analysis. Instead of measuring relationships directly, regression-analysis methods model the relationship between two time series and assess the relationship strength using the fidelity of the trained model. The underlying assumption is that for two related time series, the values of one time series contain enough information to predict the values of another time series. As a consequence, trained models show a low error for pairs of related time series.

Hong et al. [HGW17] develop an elegant way to train first-order autoregressive models using linear algebra to parallelize and speed up computations. Chen et al. [Ch20] combine several regression models to infer relations between sensors. While the idea is intuitive and common in statistics (e.g., Granger Causality), regression analysis also has significant disadvantages. Inferring relations via modeling requires models for every possible pair of signals, which are, therefore, quadratic with the number of signals. This limits the choice of regression models for a feasible analysis. Additionally, the selection of models is non-trivial, as the relations between the sensors are not necessarily linear.

Event Analysis. Buildings and industrial facilities most likely change their operational status several times within short periods (days). This is also true for their underlying functional systems. Most buildings and their respective rooms (functional groups) will be occupied at different times throughout the day. Industrial facilities like power plants contain several subsystems that are active and inactive at different times during the day. These operational state changes are most likely visible in their corresponding measurements. When they occur at different points in time, one can infer relationships by finding sensors with simultaneous events.

Fontugne et al. [Fo12] and Hong et al. [Ho13] implement this idea by decomposing a time series into different frequency components via the Empirical Mode Decomposition (EMD). Arguing that the higher frequency components correspond to events and suppress diurnal cycles, they use the correlation of medium and higher frequency Intrinsic Mode Functions (IMFs) to infer relationships. Later, Hong et al. [Ho19] developed an algorithm that detects events using a Markovian event model. Stinner et al. [St19] use temporal frequent patterns by separating the time series into their transitional states and then counting the associations between them during these transitional states. Gonzalez and Amft [GA15] identify events via numerical derivation. In a previous paper, we used Change Point Detection (CPD) to locate and correlate changes with a similar idea in mind [WL23]. While we recognize the potential in these ideas, we still want to test other measures from various research fields to offer a different perspective on the problem of relationship inference and possibly identify other promising candidates.

Pritoni et al. [Pr15] and Koh et al. [Ko16] take a similar but active approach. Instead of relying on operational status changes, they actively introduce perturbations into the measurements and then search for all sensors that changed. While this is valid and guarantees separable changes per functional group, having active control in a system is uncommon in the context of time-series integration. Access to the system is either not feasible, e.g., for industrial facilities in production, or not possible for historic data.

Supervised Machine Learning. Naturally, the problem of relationship inference can be transformed into a supervised classification problem that uses a labeled dataset of related

time series to train machine-learning models. The literature on relationship classification approaches the problem from two different perspectives. On the one hand, Li et al. [LHW20] and Wu et al. [WYW23] train siamese neural networks to learn an embedding of time series into a metric space, where distance is proportional to the relationship. On the other hand, Stinner et al. [St22] and Wan et al. [Wa23] train models to directly classify whether two time series are related by modeling the inference as a binary classification problem. All approaches differ in their deployed architecture to embed the time series.

The biggest challenges for supervised machine learning for time-series integration are the poor availability of training datasets and the high heterogeneity of the time series, hampering the transferability of trained models [St22]. In other words, it is challenging to obtain labeled data, and due to the heterogeneity of time series and industrial facilities, transferring models is challenging.

3 Methods

This paper analyzes different measures for finding related time series in large collections of heterogeneous time series in the context of time-series integration and meta-data extraction for data sets from buildings and industrial facilities. This is facilitated by the seminal work of Cliff et al. [Cl23] and their corresponding Python package, PySPI. The package compiles over 200 statistics of pairwise interactions (SPIs) from interdisciplinary literature and provides them in a unifying view. While their paper unifies different relationship measures and groups them into similar behavior, we build upon their collection to analyze the performance of different measures for finding related but heterogeneous time series originating from buildings and industrial facilities.

This analysis is motivated by the lack of common baselines in the related work and the common choice of PCC in practical applications. This contrasts with a multitude of other measures available in the literature.

3.1 Relationship Measures

There are six overarching categories of relationship measures defined in the original paper [Cl23]. Table 1 is an overview of all the measures used in our study. In addition to the theoretical measures, each measure can be computed with different estimators from discrete time series. This increases the number of measures from the ones listed in table 1. The problem of inferring relations is inherently quadratic in the number of signals, and the computational demand varies significantly for the different relationship measures. As a consequence of initial test runs, we limited our measurements to the 'fast' subset defined in the original paper [Cl23]. For details about the relationship measures, we refer the interested reader to the appendix of Cliff et al. [Cl23].

Some of the employed relationship measures are dissimilarities and others are similarities. Not all measures are mathematical norms [Ba22]. In our context, a high similarity indicates

that two time series are related, while a high dissimilarity indicates that two time series are unrelated. To resolve this counterdirection, we individually assigned each measure to the category of similarity or dissimilarity. To be able to compute all metrics for dissimilarities and similarities in the same way, we invert the dissimilarities using the exponential heuristic in combination with column-wise min-max-normalization [MGdL22]:

$$\text{Similarity} = \exp \left\{ -\frac{\text{Dissimilarity} - \min(\text{Dissimilarity})}{\max(\text{Dissimilarity}) - \min(\text{Dissimilarity})} \right\} \quad (1)$$

3.2 Combination of Measures

Intuitively, there can not be a singular measure performing exceptionally well for all datasets, as the relationships between time series take different forms. Based on this intuition, the next logical step is to combine multiple measures to account for a wider range of relationships. This can be done using supervised learning, but we want to keep it to unsupervised combinations for similar reasons as discussed in section 2.2. We test and employ four different linear approaches to combine multiple measures. Before combining the measures, we employ column-wise z-score normalization of each similarity matrix to account for different ranges of each relationship measure.

The straightforward linear combination is the sum relationship measures, which is directly proportional to computing the mean of all measures. To obtain a weighted linear combination, we flatten the similarity matrices and calculate the Principle Component Analysis (PCA), keeping only the major principle component for a weighted linear combination of all measures. We then also compute the median of all measures. Additionally, we employ a popular method to combine similarity networks based on genomic data called Similarity Network Fusion (SNF) [Wa14].

3.3 Evaluation Metrics

To compare the different relationship measures, we require scalar metrics that capture the performance of each measure across multiple datasets. There is no agreement on a single evaluation metric in the related work. As a consequence, we chose multiple metrics that capture different aspects. Firstly, we adopt the perspective of Information Retrieval (IR), where algorithms are assessed on their ability to retrieve the top k most relevant results for a given query (fig. 1 left side). Secondly, we interpret the time-series collection as a graph, with each vertex representing a time series and edges representing the relationships between sensors. We evaluate this graph by clustering the nodes (fig. 1 right side). In total, we employ seven different evaluation metrics to assess the performance of the relationship measures discussed in this paper.

Tab. 1: List of relationship measures used in this study. Similarity and distance are abbreviated with Sim. and Dist., respectively. Overall, there are 42 measures and 216 estimators.

Category	Methods	Specifier	Estimators	Sim.	Dist.
Basic Statistics	Covariance	cov	11	✓	
	Precision	prec	10	✓	
	Cross Correlation	xcorr	6	✓	
	Kendall's Rank Correlation Coefficient	kendalltau	2	✓	
	Spearman's Rank Correlation Coefficient	spearmanr	2	✓	
Distance Similarity	Barycenter	bary	4	✓	
	Distance Correlation	dcorr	2	✓	
	Hilbert-Schmidt Independence Criterion	hsic	2	✓	
	Gromov-Wasserstein Distance	gwtau	1		✓
	Pairwise Distance	pdist	6		✓
Causal Inference	Additive Noise Model	anm	1	✓	
	Conditional Distribution Similarity Fit	cds	1	✓	
	Information-Geometric Causal Inference	igci	1	✓	
	Regression-Error Based Causal Inference	reci	1		✓
Information Theory	Granger Causality	gc	1	✓	
	Mutual Information	mi	4	✓	
	Time-lagged Mutual Information	tlmi	4	✓	
	Stochastic Interaction	si	3	✓	
	Transfer Entropy	te	1	✓	
	Cross-Map Entropy	xme	6	✓	
	Conditional Entropy	ce	2		✓
	Joint Entropy	je	3		✓
Spectral	Coherence Magnitude	cohmag	6	✓	
	Directed Coherence	dcoh	6	✓	
	Debiased squared phase lag index	dspli	6	✓	
	Debiased sq. weighted phase lag index	dswpli	6	✓	
	Directed transfer function	dtf	6	✓	
	Direct directed transfer function	ddtf	6	✓	
	Imaginary Coherence	icoh	6	✓	
	Partial Directed Coherence	pdcoh	6	✓	
	Generalised partial directed coherence	gpdcoh	6	✓	
	Coherence Phase	phase	6	✓	
	Group Delay	gd	3	✓	
	Pairwise Phase Consistency	ppc	6	✓	
	Phase Lag Index	pli	6	✓	
	Weighted phase lag index	wpli	6	✓	
	Phase Locking Value	plv	6	✓	
	Phase Slope Index	psi	9	✓	
	Spectral Granger Causality	sgc	24	✓	
	Misc.	Cointegration	coint	11	✓
Power Envelope Correlation		pec	6	✓	
Linear Model Fit		lmfit	5		✓

3.3.1 Information Retrieval

Information Retrieval is the process of obtaining relevant information from a large repository based on user queries or search criteria [MC18]. In our context, we want to retrieve all related time series for a query time series representing a particular functional group. We employ three of the most commonly used metrics from the field of IR to evaluate the retrieval performance. Formulas and notation in this paragraph are taken from Mitra et al. [MC18]. See table 2 for a description of all symbols.

Tab. 2: Notation for IR metrics.

Symbol	Meaning
Query q	The q denotes a single query, i.e., finding all related time series to a query time series.
Set of Queries \underline{Q}	Multiple queries build a query set \underline{Q} .
Document d	Each query q retrieved multiple documents d which correspond to time series in the scope of this paper.
Set of Documents D	The collection of documents d to retrieve is noted as D .
Tuple in Result $\langle i, d \rangle_q$	Retrieved Document d at rank i .
Set of Ranked Results R_q	The ranked set of time series returned by query q . The lowest rank has the strongest relation to the query.
Relevance $rel_q(d)$	The relevance of a document d for query q . In our case, the relevance is binary. Where one stands for a related time series.

Mean Reciprocal Rank (MRR). The MRR is calculated as the average of the reciprocal ranks of the first relevant result for a set of queries. The MRR is one for a perfect retrieval and decreases with the quality of the retrieval. The Reciprocal Rank (RR) for a single query can be computed as follows:

$$RR_q = \max_{\langle i, d \rangle_q \in R_q} \frac{rel_q(d)}{i} \quad (2)$$

Mean Average Precision (MAP). MAP is calculated as the mean of the average precision scores for a set of queries, where average precision is the average of precision values at relevant result positions. The average precision for a single query can be computed as follows:

$$AP_q = \frac{\sum_{\langle i, d \rangle_q \in R_q} Prec_{q,i} * rel_q}{\sum_{d \in D} rel_q(d)} \quad \text{with } Prec_{q,i} = \frac{\sum_{\langle 1, d \rangle_q}^{i, d \rangle_q} rel_q(d)}{|R_q|} \quad (3)$$

Normalized Discounted Cumulative Gain (NDCG). NDCG measures the usefulness of a ranked list of results by considering the position and relevance of each result. The initial gain of a query is noted as Discounted Cumulative Gain (DCG) and is normalized by the

gain of a perfect retrieval, noted as ideal Discounted Cumulative Gain (IDCG), to facilitate comparison across different queries. The NDCG of a query can be computed as follows:

$$NDCG_q = \frac{DCG_q}{IDCG_q} \quad \text{with } DCG_q = \sum_{\langle i,d \rangle_q \in R_q} \frac{rel_q(d)}{\log_2(i)}, \quad IDCG_q = \sum_{i=1}^{|R_q|} \frac{1}{\log_2(i)} \quad (4)$$

In addition, we include **Triplet Accuracy (TA)** as it is used for evaluation in the related work [WYW23]. TA measures the proportion of correctly ordered triplets out of the total number of assessed triplets for similarity learning. Triplets always consist of three time series: one time series is the anchor, and the other two are a negative (unrelated time series) and a positive example (related time series). A triplet is correct if the positive example has a higher similarity than the negative example. TA is computed using all possible triplets.

3.3.2 Clustering Performance

In addition to the retrieval quality, we also evaluate the inferred time-series graph. Unfortunately, determining the real and continuous relationship weights is challenging for all applications and datasets. Therefore, we rely on binary, undirected edges for the ground-truth relations between time series. Time series from the same functional group are viewed as one connected cluster. We use spectral clustering on the inferred relationships to extract clusters from the similarity matrix. We chose this algorithm because of the number of functional groups (clusters). Clusters are not necessarily uniform nor circular, and due to the use of non-euclidean measures. We selected three common clustering metrics for the evaluation [HA85, Ro16]. All measures are defined using the symbols from the following contingency table 3, where the numbers n_{ij} in each cell are the number of time series assigned to cluster i for clustering result V and assigned to cluster j in clustering result U , v_i and u_j are the sizes of the cluster V_i and U_j for each clustering result respectively.

Tab. 3: Contingency table with notations for clustering.

$V \setminus U$	U_1	U_2	\dots	U_s	$v_i = \sum_{j=1}^s n_{ij}$
V_1	n_{11}	n_{12}	\dots	n_{1s}	v_1
\vdots	\vdots	\vdots	\ddots	\vdots	\vdots
V_r	n_{r1}	n_{r2}	\dots	n_{rs}	v_r
$u_j = \sum_{i=1}^r n_{ij}$	u_1	u_2	\dots	u_s	$n = \sum_{i=1}^r \sum_{j=1}^s n_{ij}$

Adjusted Rand Index (ARI). The ARI evaluates the similarity of two clustering results by comparing similarly clustered pairs of points to all pairs of points. The adjustment accounts for the chance of randomly pairing points correctly. Correctly paired points are the ones

that are in the same cluster for both results or in different clusters for both results. The ARI is defined as follows:

$$\text{ARI} = \frac{\sum_{i=1}^r \sum_{j=1}^s \binom{n_{ij}}{2} - \left[\sum_{j=1}^r \binom{u_j}{2} \sum_{i=1}^s \binom{v_i}{2} \right] / \binom{n}{2}}{0.5 \left[\sum_{j=1}^r \binom{u_j}{2} + \sum_{i=1}^s \binom{v_i}{2} \right] - \left[\sum_{j=1}^r \binom{u_j}{2} \sum_{j=i}^s \binom{v_i}{2} \right] / \binom{n}{2}} \quad (5)$$

Adjusted Mutual Information (AMI). The AMI evaluates the similarity of two clustering results by comparing the amount of shared information between the clusterings while accounting for the possibility of chance agreements. It is computed using different information theoretic entropies:

$$\text{AMI}(U, V) = \frac{\text{MI}(U, V) - \mathbb{E}[\text{MI}(U, V)]}{\max(H(U), H(V)) - \mathbb{E}[\text{MI}(U, V)]} \quad (6)$$

$$\text{with } \text{MI}(U, V) = \sum_{i=1}^r \sum_{j=1}^s \frac{n_{ij}}{n} \log \frac{n_{ij} * n}{v_i * u_j}, \quad \mathbb{E}[\text{MI}(X, Y)] = \sum_{i=1}^r \sum_{j=1}^s \frac{v_i * u_j}{n^2} \log \frac{v_i * u_j}{n}$$

$$\text{and } H(X) = - \sum_{i=1}^r \frac{v_i}{n} \log \left(\frac{v_i}{n} \right), \quad H(Y) = - \sum_{j=1}^s \frac{u_j}{n} \log \left(\frac{u_j}{n} \right)$$

V-Measure (VM). The V-Measure combines two other metrics, homogeneity and completeness, into a single value. Homogeneity measures how well each cluster contains only data points of a single class. Completeness measures how well all data points of a given class are assigned to the same cluster. The V-Measure is the harmonic mean of homogeneity and completeness. It is defined as follows:

$$V = 2 * \frac{H * C}{H + C} \quad \text{with } H = 1 - \frac{\sum_{i=1}^r \sum_{j=1}^s \frac{n_{ij}}{n} \log \frac{n_{ij}}{v_i}}{\sum_{i=1}^r \frac{v_i}{n} \log \frac{v_i}{n}}, \quad C = 1 - \frac{\sum_{i=1}^r \sum_{j=1}^s \frac{n_{ij}}{n} \log \frac{n_{ij}}{u_j}}{\sum_{j=1}^s \frac{u_j}{n} \log \frac{u_j}{n}} \quad (7)$$

3.4 Final Ranking

Our final goal is to compare different relationship measures by their performance on different test datasets. As one metric alone does not capture every piece of information, we collected the most common ones from information retrieval and clustering in the previous chapter. We are not interested in the absolute performance of each relationship measure but in how they compare. Therefore, we do not evaluate absolute values but rank each relationship measure on the corresponding metric. For each metric, the best relationship measure achieves rank one, and rank L is assigned to the worst measure for L different relationship measures. This makes the measures comparable over different datasets, which might vary in difficulty and, therefore, absolute achievable performance. We then compute the mean rank on all metrics and all datasets for the final ranking. When n relationship measures perform equally well, we apply average ranking, where the rank is the average of all n measures, sorted by their initial position in the input vector. With this, we want to mitigate the edge case, where many relationship measures perform well, and one does not at all. In these cases, the low-performing measure would receive rank two, conveying the impression that this measure works well in comparison.

4 Evaluation

4.1 Available Datasets

For our evaluation, we gathered several datasets, each with different properties in terms of contained measurements and functional groups (see tab. 4). We found two publicly available building datasets containing measurements from different university buildings, Soda Hall (soda) and Sutardja Dai Hall (keti) at UC Berkeley, containing data from approximately one week. These datasets have been made public by Wu et al. [WYW23] and Hong et al. [HGW17], for which we are very thankful. Each dataset contains temperature, carbon dioxide, humidity, luminosity, and motion sensors (PIR) with meta-data stating the room in which each sensor is located. Unlike Wu et al. [WYW23], we do not exclude the PIR measurements from the dataset, even though changes in these sensors are exceptionally rare. In addition to the public datasets, we enriched our collection with additional proprietary datasets to achieve higher data diversity. One dataset contains the recorded time series of 14 wire-braiding machines, where the signals include rotation speed, rotation temperature, and machine status. The functional groups for this dataset are the different machines. The recorded data spans approximately three months. There is no known issue with missing data or malfunctioning sensors. All machines work in different cycles on different products. One cycle typically takes around three days to complete.

All of the previous datasets contain only a few time series per functional group. To provide larger groups, we also use a large collection of time series recorded in different combined-cycle gas power plants. These plants can be separated into blocks, where each block is a small plant on its own. Each of these blocks contains two gas-powered turbines and a steam turbine, which is powered by steam generated using hot exhaust gases. The measurements span approximately four months. In contrast to the other dataset, each turbine contains 100 to 150 sensors, resulting in a size of over 100 time series per functional group and way fewer different groups than the other datasets. The signals are diverse and include temperature, pressure, electric, and flow measurements. Temperature and pressure are the most common measurements [WL23]. We include data from four blocks (plant 1 – 4). Shutdowns and start-ups are not regular; on average, there is one every two weeks.

The datasets contain different impurities. For the building datasets, we expect that seasonalities heavily influence each time series due to diurnal and environmental influences, while the changes due to human presence are minor. The precision of our industrial data sets is limited due to dead-banding and irregular measurements. Additionally, one of the power plant datasets contains periods of imputed and steady data where the SCADA system was on hold for maintenance operations.

4.2 Preprocessing

It is not feasible to deploy sophisticated preprocessing routines to each dataset as the time series are diverse and have no accompanying meta-data information. Consequentially, we

Tab. 4: Datasets Information.

Name	Origin	Signals	Functional Group	Duration	Resampling
Rotary	Braiding Machines	42	Machines (14)	1 month	5 min
Keti	Building	255	Rooms (47)	1 week	1 min
Soda	Building	394	Rooms (78)	1 week	1 min
Plant 1	Power Plant	407	Turbines (3)	2 month	10 min
Plant 2	Power Plant	408	Turbines (3)	2 month	10 min
Plant 3	Power Plant	407	Turbines (3)	2 month	10 min
Plant 4	Power Plant	408	Turbines (3)	2 month	10 min

keep the preprocessing steps to a minimum. We resample the time series using linear interpolation so the samples are equidistant and synchronous for all time series within one dataset. For each dataset, we choose a sampling rate that is appropriate to capture changes and ensures that all time series from all datasets are roughly equal in length (see tab. 4). With the resampling, each time series has a length of roughly 10,000 samples. We then exclude signals that have a standard deviation of less than 1^{-10} as these signals are constant over almost the complete duration. Intuitively, these time series do not contain enough information to be affiliated with a functional group. Finally, the time series are z-score normalized and de-trended, which is good practice and is required by most relationship measures.

4.3 Implementation and Evaluation Details

Due to the number of different relationship measures, we restrict the overall computation time to a feasible level. As the acquisition of pairwise relationships is inherently quadratic, we define a time limit for each pair and multiply this limit by the number of signals squared to reach the final time budget for each measure. This is in line with other comparative papers, e.g., for time-series-anomaly detection [SWP22]. We set the time limit per pair to 250 ms. We distribute the computations over 64 cores using an AMD EPYC 7713 CPU and 512GB RAM. Computations took about four days, and computing all relationship measures for each dataset took twelve hours on average.

Notably, some measures did not enter their main computational pipeline. They timed out while acquiring hundreds of gigabytes of memory for internal arrays. With an internal limitation on memory acquisition, the available memory was never exceeded. In addition to our timing restrictions, we excluded similarity matrices containing undefined values (NaN). Table 5 shows how many relationship measures terminated for the different datasets. Measures that timed out or had memory issues stem mainly from the spectral category. Only a few measures from other categories timed out. This is likely tied to the implementation of the spectral measures, as most of them use the same underlying library. In conclusion, out of the 216 different measures, only 68 terminated for all datasets. See fig. 2 for a listing of successfully terminated relationship measures.

Tab. 5: Numbers of terminated and excluded relationship measures. Initially, the computation started with 216 measures per dataset. Exclusion was due to time, memory, and NaN.

Dataset	Initial Number	Time limit	Memory Issue	Partial NaNs	Finalized
Rotary	216	-43	0	-7	166
KETI	216	-13	-56	-66	81
Soda	216	-43	-61	-23	89
Plant 1	216	-13	-62	-68	73
Plant 2	216	-43	-62	-32	79
Plant 3	216	-45	-62	-32	77
Plant 4	216	-34	-62	-44	76

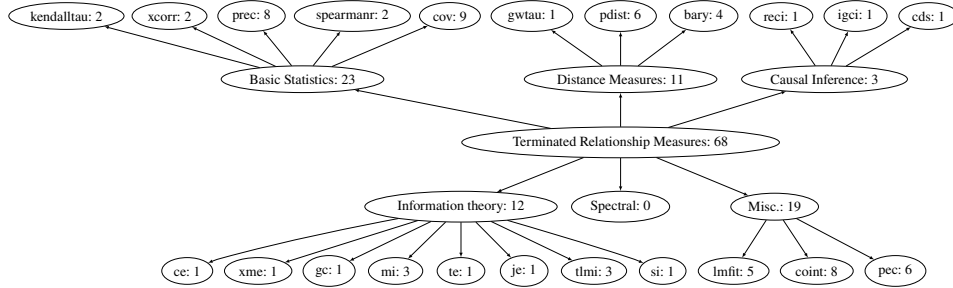


Fig. 2: Relationship measures that we included for analysis. Split by categories and specifiers.

4.4 Results

For brevity, we only show the results of the top-ranked relationship measures. We refer the reader to the accompanying GitHub repository for the complete results [We24]. To make our results comparable with the related work, note the following parallels:

Cov.: For z-scored signals, the off-diagonal elements of the covariance matrices are equal to the PCC. Therefore, the results of the relationship measure *cov* are equal to using the PCC as described in section 2.2.

Lmfit/Gc.: While not completely equal, *lmfit* and *gc* follow a similar idea as the related work in section 2.2. These measures create linear models and use their predictive power to indicate relationship strength.

Tab. 6 describes the value distribution of the relationship measure performances per dataset and evaluation metric. For all metrics, higher values are better. Looking at the minimum and maximum, we can see that the performances for the information retrieval metrics mostly cover a large part of the value range for each metric. In other words, there are good and bad relationship measures for each IR metric and dataset. The difference in MAP/TA and MRR shows that on average, the nearest neighbour is often related (high MRR), but not all neighbours are (MAP/TA not completely reaching their maximum). For the clustering

metrics, even the best performing best-performing relationship measures are relatively far from the theoretical maximum value, indicating that clustering is the harder task. We can also see that the methods show, on average, similar performance for all the power plant datasets, while they perform very differently within the building datasets and the rotary datasets. In conclusion, deriving functional relations is more challenging on the plant and the Keti datasets than on rotary and soda.

Tab. 6: Absolute metric values for all datasets. Higher is better for all metrics. The statistics are calculated over all relationship measures per evaluation metric and dataset. Using \wedge : Minimum, \varnothing : Mean, \vee : Maximum as symbols.

Metric		Plant1	Plant2	Plant3	Plant4	Rotary	Soda	Keti
MRR	\wedge	0.00	0.00	0.00	0.00	0.03	0.01	0.01
	\varnothing	0.73	0.76	0.77	0.79	0.71	0.96	0.23
	\vee	0.93	0.93	0.93	0.93	1.00	1.00	0.71
MAP	\wedge	0.30	0.31	0.30	0.28	0.06	0.00	0.03
	\varnothing	0.45	0.49	0.51	0.54	0.62	0.66	0.10
	\vee	0.54	0.60	0.63	0.70	0.96	0.99	0.32
NDCG	\wedge	0.68	0.68	0.68	0.68	0.23	0.12	0.22
	\varnothing	0.81	0.83	0.83	0.85	0.74	0.72	0.33
	\vee	0.87	0.88	0.89	0.91	1.00	1.00	0.57
TA	\wedge	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	\varnothing	0.54	0.58	0.59	0.61	0.70	0.83	0.53
	\vee	0.64	0.69	0.74	0.78	0.87	1.00	0.72
ARI	\wedge	-0.01	-0.01	-0.01	-0.01	-0.03	-0.00	-0.01
	\varnothing	0.04	0.14	0.19	0.19	0.52	0.46	0.02
	\vee	0.37	0.36	0.42	0.43	1.00	0.98	0.12
AMI	\wedge	-0.00	-0.00	-0.00	-0.00	-0.07	-0.04	-0.04
	\varnothing	0.05	0.16	0.19	0.21	0.59	0.62	0.07
	\vee	0.29	0.35	0.37	0.42	1.00	0.98	0.27
VM	\wedge	0.00	0.00	0.01	0.01	0.46	0.65	0.31
	\varnothing	0.06	0.16	0.20	0.22	0.81	0.91	0.51
	\vee	0.29	0.36	0.37	0.43	1.00	1.00	0.67

The ranks for each of the seven top-performing measures are shown in fig. 7. On the left side, all metrics and datasets are used. On the right side, the datasets are grouped into building and power plant datasets. The order of the legend elements corresponds to their ranking (the top element has the highest ranking). Each horizontal line corresponds to a different metric. The bold lines mark the mean rank averaged over all metrics for a single dataset with its name on the right side of the plot. The perfect relationship measure would be visualized as a vertical line on the rightmost side of the plot, ranking high throughout all metrics and datasets. A line on the left side indicates bad performance. Due to average ranking, the lines sometimes collectively dip to the left if there are a multitude of well-performing measures. An example can be seen for the MRR in Fig. 7c (first line).

Instead of separating the ranking by datasets, we can also separate by metrics. In fig. 7, we indicate this separation by using normal font style for the information-retrieval metrics and italic style for the clustering metrics. More directly, this can be seen in fig.

3-5, where the best-performing measures are on the left, and their corresponding line indicates their rank on a decreasing axis. See the section 5.1 for a detailed discussion of the results.

As mentioned in section 3.2, we also want to evaluate different measure combinations of relationship measures. We selected the following measures based on their good performance overall (**fused-perf**):

- Power Envelope Correlation (PEC) and Precision Matrix (squared) (Prec-sq) are the best-performing measures for both the power plant and building datasets for both clustering and IR metrics (fig. 7b and 7c).
- Time-lagged Mutual Information (TIMI) and Stochastic Interaction (SI) perform second best for the IR metrics separated by dataset type (fig. 4a and 5a).
- Granger Causality (GC) and Conditional Entropy (CE) both perform well for clustering metrics (fig. 4b and 5b).

In addition, we select the covariance matrix, the precision matrix, and the PEC as they can be computed relatively fast and perform comparatively well (**fused-speed**). All of the three measures only take seconds for the largest dataset containing 407 signals. The results in fig. 6 show that combining measures improves the ranks over using singular measures.

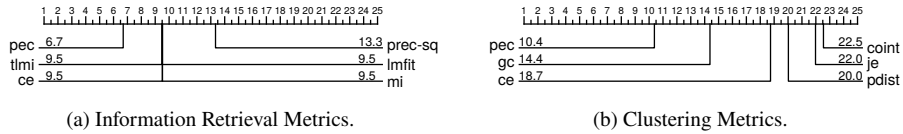


Fig. 3: Mean ranks for different metric sets averaged over **all datasets**.

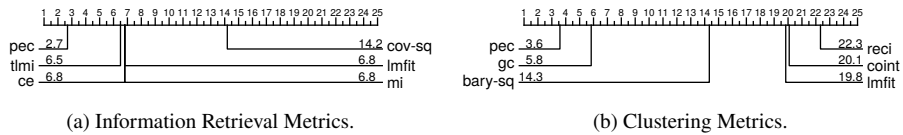


Fig. 4: Mean ranks for different metric sets averaged over all **power plant datasets**.

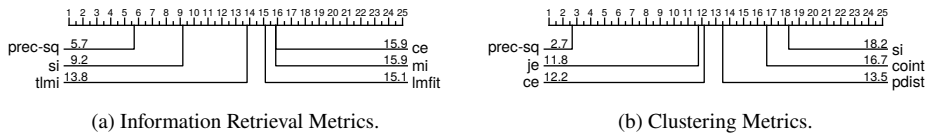


Fig. 5: Mean ranks for different metric sets averaged over all **building datasets**.

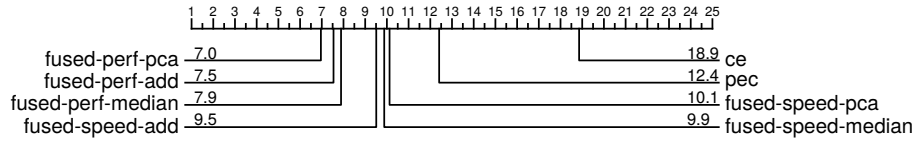


Fig. 6: Mean ranks with different combination methods and selections of fused measures averaged over **all datasets** and all metrics.

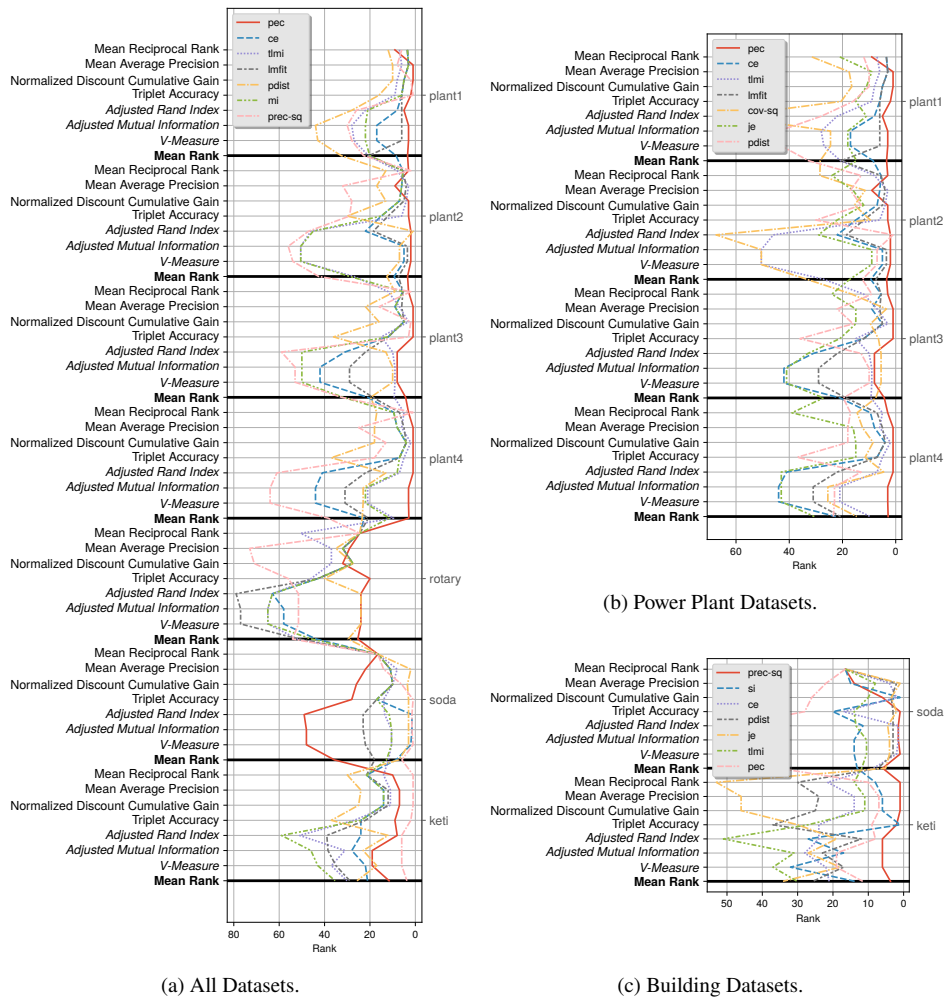


Fig. 7: Resulting ranks over different metrics and different datasets for the top seven relationship measures. The legends are sorted by average ranking over all metrics and datasets. The top-ranking measure is listed first. The evaluation metrics for clustering and IR are separated by font.

5 Discussion

5.1 Discussion of Results

Overall, PEC is the best-performing relationship measure (fig. 7a). This is surprising as the power envelope smoothes the signal, eliminating high-frequency variations. This seems contradictory to the ideas in section 2.2, where it is stated that events are of higher frequency than the normal variations. Comparing figure 7a and the results for different datasets in figures 7b and 7c, it becomes evident that PEC mainly performs well on the power-plant datasets. While it still is seventh place for the building datasets, several other measures are outranking it.

We attribute this to the selectivity and prominence of events in the different datasets. When taking the perspective of the related work from section 2.2, events pertaining to a single functional subgroup (high selectivity) help to identify related signals if they clearly affect all the signals (high prominence). Events influencing all signals equally can be seen as confounding variables. In the power plant context, the main events are shutdowns and start-ups. The turbines are turned down individually, with delays occurring between the shutdowns. With this, the shutdowns have a high selectivity and high prominence as they cause clearly measurable changes in all affected components. For the building data, the presence of a person (high selectivity) in a room most likely only influences the measurements minimally (low prominence). In contrast, day and night cycles most likely heavily influence the sensors (high prominence). For example, sunlight and office hours influence all building rooms equally (low selectivity).

CE is the second best relationship measure (fig. 7a). The lower the CE for one time series is, the less chaotic its values are for a fixed value of the other time series. A low CE indicates that given the other signals' value, only a few values are possible for the current signal. This ranking is somewhat stable comparing the different dataset types, with it taking second place for the power plant datasets (fig. 7b) and third place for the building datasets (fig. 7c). CE is linked to Joint Entropy (JE) and Mutual Information (MI) by additive and subtractive relationships. It is, therefore, not surprising that these measures also rank relatively high.

Regression-based measures (gc and lmfit) also perform well overall, which is in line with the related work (see section 2.2). While they are among the top-performing measures for the power-plant datasets, they are not performing similarly well for the building datasets. We assume this can be attributed to a similar reason as discussed for PEC. With highly prominent and low selectivity day and night cycles, regression models will perform well even for unrelated pairs of time series. As the error of the regression models is averaged over a long period, the highly selective events, such as the presence of persons in individual rooms, do not influence the result as much.

In contrast to the power-plant datasets, the precision matrix performs exceptionally well for the building datasets. This is again strongly linked with the argument of confounding

events (high prominence, low selectivity). The off-diagonal elements of the precision matrix are related to the partial correlation of the two elements. By inverting the covariance matrix, the effect of confounding variables is reduced. While the day and night cycle as a confounder is not measured directly, we assume its significant representation in all of the time series is enough to reduce the spurious correlations.

It is important to note that the covariance matrix, and with that, the PCC, is not the best-performing relationship measure. It is the eighth best-performing measure but is surpassed by information-theoretic measures in all our comparisons. This is in line with results from causal discovery [Ru19]. Again, we argue that spurious correlations, introduced by the day and night cycles, decrease the performance of the covariance matrix. The covariance performing better for the power plant dataset than on the building datasets supports the argument.

Another observation is the difference in top-ranked measures for the IR and clustering metrics visible in figs. 3-5, when comparing the left and right plots of each figure. Only the top-ranked measure is the same on both sides, while the other measures change. The difference is most likely rooted in the different perspectives. The IR metrics only evaluate the relatedness of the items to be retrieved. In contrast, the clustering metrics assess the complete structure of the inferred time-series graph. With that, not only is the closeness of related signals important, but so is the separation from unrelated signals.

Unsurprisingly, the fused measures rank significantly higher than singular measures, showing a promising prospect for combining multiple relationship measures (fig. 6). Even using only the fastest and most accurate measures yields significant advantages (fused-speed). Except for SNF, there is almost no difference when combining the measures using summation, PCA or median. SNF did not perform well enough to be part of the presented figures. Although our results indicate that the combination of metrics can yield significant improvements, the initial ideas presented here leave room for further exploration. This includes other ways of fusing the measures and if there is an optimal number or if the quality only improves when fusing a growing number of measures. Unfortunately, this is out of the scope of this paper and requires dedicated analysis. We consider this future work.

5.2 Limitations and Future Work

Our results show that the performance of different relationship measures depends on the dataset and the prominence of the events that imply a relationship between two time series. A broader collection of data incorporating even more time series and relations would further stabilize these results. We are aware of this problem, but obtaining public data for evaluations is rare [St22]. We contacted the authors of several studies but could not obtain more public data. Still, we compare different relationship measures using diverse data and different evaluation perspectives. In the related work, often only one measure is evaluated on a singular dataset without comparison to an appropriate baseline.

Another limitation is the usage of default parameters for all relationship measures. Similar to other algorithms, some measures allow for parameter tuning. As the relationship measures stem from various domains and have not been explicitly developed for heterogeneous time-series data integration, the default hyper-parameters might not fit our application. Still, our insights are valuable since the setting is similar to the practical use case. When working with large time-series datasets, meta-data inference is often only the initial step in the pipeline up to an ultimate goal, and the available information on the data is sometimes minimal. In this situation, practitioners will likely not utilize significant computational resources for extended hyperparameter optimization. Additionally, we omit the discussion of runtime as it most likely heavily depends on the respective implementation of each relationship measure, which we did not optimize. Nevertheless, we kept track of the runtimes and provide the measurements in our repository [We24].

While measurements on real-world datasets are desirable to measure practical feasibility, simulated data could improve the analysis of relationship measures. In our case, the data includes different overlapping quality issues, like missing data, noisy signals, and noisy labels. Therefore, separating cause-effect relations of different impurities on the measures is difficult. Additionally, the functional subgroups are not guaranteed to be separable based only on the raw signals. For example, ground truth meta-data could indicate that two signals are from different rooms, while in reality, these rooms are not separated by a wall. Hence, their measurements are likely linked. A targeted analysis of singular impurities in a simulation environment could help extract further insights.

6 Conclusion

This paper analyzes the performance of various relationship measures from diverse domains to infer relationships between heterogeneous time series. It evaluates them from different perspectives using seven information retrieval and clustering metrics. This analysis serves dual purposes. Practically, it offers a valuable reference for selecting an appropriate relationship measure when a quick and informed decision is needed without prior knowledge. Additionally, our analysis provides a reference for selecting a baseline when evaluating novel ideas for relationship inference in heterogeneous time series collections.

No single measure clearly outperforms all competitors. Nevertheless, the results show that while the Pearson Correlation Coefficient is the de-facto gold standard relationship measure, information-theoretic measures, namely conditional entropy or mutual information, are more likely to capture the important relationships in the absence of any prior knowledge. With prominent and selective events, power-envelope correlation performs well, while the precision matrix is most likely to perform well in the case of confounding, low selectivity, and high prominence events such as day and night cycles influencing all signals. Furthermore, combining different measures will most likely improve the relationship inference. For example, simply summing the power-envelope correlation, correlation matrix, and precision matrix captures different relational aspects with only little computational overhead.

Bibliography

- [AGN15] Abedjan, Ziawasch; Golab, Lukasz; Naumann, Felix: Profiling relational data: a survey. *The VLDB Journal*, 24:557–581, 8 2015.
- [AI23] Almashor, Mahathir; Rana, Mashud; McCulloch, John; Rahman, Ashfaqur; Sethuvenktraman, Subbu: What’s The Point: AutoEncoding Building Point Names. In: *Proceedings of the 10th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*. ACM, pp. 256–260, 11 2023.
- [Ba22] Banach, Stefan: Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales. *Fundamenta Mathematicae*, 3(1):133–181, 1922.
- [Ba18] Balaji, Bharathan; Bhattacharya, Arka; Fierro, Gabriel; Gao, Jingkun; Gluck, Joshua; Hong, Dezhi; Johansen, Aslak; Koh, Jason; Ploennigs, Joern; Agarwal, Yuvraj; Bergés, Mario; Culler, David; Gupta, Rajesh K.; Kjærgaard, Mikkel Baun; Srivastava, Mani; Whitehouse, Kamin: Brick : Metadata schema for portable smart building applications. *Applied Energy*, 226:1273–1292, 9 2018.
- [BGBW19] Brugere, Ivan; Gallagher, Brian; Berger-Wolf, Tanya Y.: Network Structure Inference, A Survey. *ACM Computing Surveys*, 51:1–39, 3 2019.
- [BM24] Benfer, Rebekka; Müller, Jochen: Semantic digital twin creation of building systems through time series based metadata inference – A review. *Energy and Buildings*, p. 114637, 8 2024.
- [Ch20] Chen, Long; Gunay, H. Burak; Shi, Zixiao; Shen, Weiming; Li, Xiaoping: A Metadata Inference Method for Building Automation Systems With Limited Semantic Information. *IEEE Transactions on Automation Science and Engineering*, 17:2107–2119, 10 2020.
- [CI23] Cliff, Oliver M.; Bryant, Annie G.; Lizier, Joseph T.; Tsuchiya, Naotsugu; Fulcher, Ben D.: Unifying pairwise interactions in complex dynamics. *Nature Computational Science*, 3:883–893, 9 2023.
- [Fo12] Fontugne, Romain; Ortiz, Jorge; Culler, David; Esaki, Hiroshi: Empirical Mode Decomposition for Intrinsic-Relationship Extraction in Large Sensor Deployments. In: *Workshop on Internet of Things Application*. 2012.
- [GA15] Gonzalez, Luis I. Lopera; Amft, Oliver: Mining relations and physical grouping of building-embedded sensors and actuators. In: *2015 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, pp. 2–10, 3 2015.
- [HA85] Hubert, Lawrence; Arabie, Phipps: Comparing partitions. *Journal of Classification*, 2:193–218, 12 1985.
- [Ha23] Hai, Rihan; Kotras, Christos; Quix, Christoph; Jarke, Matthias: Data Lakes: A Survey of Functions and Systems. *IEEE Transactions on Knowledge and Data Engineering*, 35:12571–12590, 12 2023.
- [HGW17] Hong, Dezhi; Gu, Quanquan; Whitehouse, Kamin: High-dimensional Time Series Clustering via Cross-Predictability. In: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*. pp. 642–651, 2017.

- [HL20] Haller, David; Lenz, Richard: Pharos: Query-Driven Schema Inference for the Semantic Web. In: Communications in Computer and Information Science. volume 1168 CCIS. Springer, pp. 112–124, 2020.
- [Ho13] Hong, Dezhi; Ortiz, Jorge; Whitehouse, Kamin; Culler, David: Towards Automatic Spatial Verification of Sensor Placement in Buildings. In: Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings. ACM, pp. 1–8, 11 2013.
- [Ho19] Hong, Dezhi; Cai, Renqin; Wang, Hongning; Whitehouse, Kamin: Learning from Correlated Events for Equipment Relation Inference in Buildings. In: Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation. ACM, pp. 203–212, 11 2019.
- [Hu19] Hulsebos, Madelon; Hu, Kevin; Bakker, Michiel; Zraggen, Emanuel; Satyanarayan, Arvind; Kraska, Tim; Çagatay Demiralp; Hidalgo, César: Sherlock. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, pp. 1500–1508, 7 2019.
- [KAB14] Koc, Merthan; Akinci, Burcu; Bergés, Mario: Comparison of linear correlation and a statistical dependency measure for inferring spatial relation of temperature sensors in buildings. In: Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings. ACM, pp. 152–155, 11 2014.
- [Ke23] Keogh, Eamonn: Time Series Data Mining: A Unifying View. Proceedings of the VLDB Endowment, 16:3861–3863, 8 2023.
- [Ko16] Koh, Jason; Balaji, Bharathan; Akhlaghi, Vahideh; Agarwal, Yuvraj; Gupta, Rajesh: Quiver: Using Control Perturbations to Increase the Observability of Sensor Data in Smart Buildings. Computing Research Repository (arXiv), 1 2016.
- [La21] Langenecker, Sven; Sturm, Christoph; Schalles, Christian; Binnig, Carsten: Towards Learned Metadata Extraction for Data Lakes. In: Lecture Notes in Informatics (LNI), Proceedings - Series of the Gesellschaft für Informatik (GI). volume P-311. Gesellschaft für Informatik (GI), pp. 325–336, 2021.
- [LHW20] Li, Shuheng; Hong, Dezhi; Wang, Hongning: Relation Inference among Sensor Time Series in Smart Buildings with Metric Learning. Proceedings of the AAAI Conference on Artificial Intelligence, 34:4683–4690, 4 2020.
- [MC18] Mitra, Bhaskar; Craswell, Nick: An Introduction to Neural Information Retrieval, volume 13. Now Foundations and Trends, 2018.
- [MGdL22] Mair, Patrick; Groenen, Patrick J. F.; de Leeuw, Jan: More on Multidimensional Scaling and Unfolding in R: smacof Version 2. Journal of Statistical Software, 102(10):1–47, 2022.
- [MMB23] Ma, Boya; McNeil, Maxwell; Bogdanov, Petko: GIST: Graph Inference for Structured Time Series. In: Proceedings of the 2023 SIAM International Conference on Data Mining (SDM). Society for Industrial and Applied Mathematics, pp. 433–441, 1 2023.
- [Pa15] Papenbrock, Thorsten; Ehrlich, Jens; Marten, Jannik; Neubert, Tommy; Rudolph, Jan-Peer; Schönberg, Martin; Zwiener, Jakob; Naumann, Felix: Functional dependency discovery. Proceedings of the VLDB Endowment, 8:1082–1093, 6 2015.

- [PLA18] Park, June Young; Lasternas, Bertrand; Aziz, Azizan: Data-Driven Framework to Find the Physical Association between AHU and VAV Terminal Unit-Pilot Study. In: ASHRAE Winter Conference Proceedings. 2018.
- [Pr15] Pritoni, Marco; Bhattacharya, Arka A.; Culler, David; Modera, Mark: A Method for Discovering Functional Relationships Between AirHandling Units and Variable- Air-Volume Boxes From Sensor Data. In: Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments. ACM, pp. 133–136, 11 2015.
- [Ro16] Romano, Simone; Vinh, Nguyen Xuan; Bailey, James; Verspoor, Karin: Adjusting for Chance Clustering Comparison Measures. *Journal of Machine Learning Research*, 17:1–32, 2016.
- [Ru19] Runge, Jakob; Nowack, Peer; Kretschmer, Marlene; Flaxman, Seth; Sejdinovic, Dino: Detecting and quantifying causal associations in large nonlinear time series datasets. *Science Advances*, 5:4996–5023, 11 2019.
- [So20] Sofos, Marina; Langevin, Jared; Deru, Michael; Gupta, Erika; Benne, Kyle; Blum, David; Bohn, Ted; Fares, Robert; Fernandez, Nick; Fink, Glenn; Frank, Steven; Gerbi, Jennifer; Granderson, Jessica; Hoffmeyer, Dale; Hong, Tianzhen; Jiron, Amy; Johnson, Stephanie; Katipamula, Srinivas; Kuruganti, Teja; Langevin, Jared; Livingood, William; Muehleisen, Ralph; Neukomm, Monica; Nubbe, Valerie; Phelan, Patrick; Piette, MaryAnn; Reyna, Janet; Roth, Amir; Satre-Meloy, Aven; Specian, Michael; Vrabie, Draguna; Wetter, Michael; Widergren, Steve: Innovations in Sensors and Controls for Building Energy Management: Research and Development Opportunities Report for Emerging Technologies. Technical Report DOE/GO-102019-5234, National Renewable Energy Laboratory (NREL), 2 2020.
- [St19] Stinner, Florian; Raßpe-Lange, Lukas; Baranski, Marc; Müller, Dirk; Takeshi: Application of unsupervised machine learning techniques for topology detection in building energy systems. *Journal of Physics: Conference Series*, 1343:012041, 11 2019.
- [St22] Stinner, Florian; Llopis-Mengual, Belén; Storek, Thomas; Kümpel, Alexander; Müller, Dirk: Comparative study of supervised algorithms for topology detection of sensor networks in building energy systems. *Automation in Construction*, 138:104248, 6 2022.
- [SWP22] Schmidl, Sebastian; Wenig, Phillip; Papenbrock, Thorsten: Anomaly detection in time series. *Proceedings of the VLDB Endowment*, 15:1779–1797, 5 2022.
- [VG18] VGBE: KKS Kraftwerk-Kennzeichensystem, volume 8. Auflage. Verlag Technisch-Wissenschaftlicher Schriften, 8 edition, 1 2018.
- [Wa14] Wang, Bo; Mezlini, Aziz M; Demir, Feyyaz; Fiume, Marc; Tu, Zhuowen; Brudno, Michael; Haibe-Kains, Benjamin; Goldenberg, Anna: Similarity network fusion for aggregating data types on a genomic scale. *Nature Methods*, 11:333–337, 3 2014.
- [Wa23] Wan, Shanshan; Zhao, Mengnan; Chen, Yimin; Yang, Shuyue; Qiu, Dongwei; Lo, L. James: A novel data-driven relationship inference approach for automatic data tagging in building heating, ventilation and air conditioning systems. *Building and Environment*, 246:110968, 12 2023.
- [We24] Weber, Lucas: Network Inference. <https://github.com/Lucew/relationship-discovery>, 2024. Accessed: 2025-01-09.

- [WL23] Weber, Lucas; Lenz, Richard: Machine learning in sensor identification for industrial systems. *it - Information Technology*, 65:177–188, 8 2023.
- [WSS21] Waterworth, David; Sethuvenkatraman, Subbu; Sheng, Quan Z.: Advancing smart building readiness: Automated metadata extraction using neural language processing methods. *Advances in Applied Energy*, 3:100041, 8 2021.
- [WYW23] Wu, Mingzhe; Yao, Fan; Wang, Hongning: An End-to-End Solution for Spatial Inference in Smart Buildings. In: *Proceedings of the 10th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*. ACM, pp. 110–119, 11 2023.
- [Yu22] Yu, Han; Cai, Hongming; Liu, Zhiyuan; Xu, Boyi; Jiang, Lihong: An Automated Metadata Generation Method for Data Lake of Industrial WoT Applications. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52:5235–5248, 8 2022.