Information Processing and Management 61 (2024) 103638

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/ipm

MDLR: A Multi-Task Disentangled Learning Representations for unsupervised time series domain adaptation

Yu Liu [a,b], Duantengchuan Li [a,*], Jian Wang [a,*], Bing Li [a,c,*], Bo Hang [b]

[a] School of Computer Science, Wuhan University, Wuhan 430072, China
[b] School of Computer Engineering, Hubei University of Arts and Science, Xiangyang 441100, China
[c] Hubei Luojia Laboratory, Wuhan 430079, China

ARTICLE INFO

Keywords:
Domain adaptation
Time series domain adaptation
Disentangled representation learning
Multi-task learning
Domain-invariant space

ABSTRACT

Unsupervised Time Series Domain Adaptation (UTSDA) is a method for transferring information from a labeled source domain to an unlabeled target domain. The majority of existing UTSDA approaches focus on learning a domain-invariant feature space by reducing the gap between domains. However, the single-task representation learning methods have limited expressive capability, while ignoring the distinctive season-related and trend-related domain-invariant mechanisms across different domains. To address this, we introduce a novel approach, distinct from existing methods, through a theoretical analysis of UTSDA from the perspective of causal inference. This analysis establishes a solid theoretical foundation for identifying and modeling such consistent domain-invariant mechanisms, which is a significant advancement in the field. As a solution, we introduce MDLR, a multi-task disentangled learning framework designed for UTSDA. MDLR utilizes a dual-tower architecture with a trend feature extractor (TFE) and a season feature extractor (SFE) to extract trend-related and season-related information. This approach ensures that domain-invariant features at different scales can be better represented. Additionally, MDLR is designed with two tasks: a label classifier and a domain classifier, enabling iterative training of the entire model. The experiments conducted on three datasets, namely UCIHAR, WISDM, and HHAR_SA, along with visualization results, have shown the effectiveness of the proposed approach. The source code for our MDLR model is available to the public at https://github.com/MoranCoder95/MDLR/.

1. Introduction

Unsupervised domain adaptation (UDA) is a method in machine learning where a model trained on labeled data from one domain is adapted to work well on an unlabeled different domain (Alqahtani, Al-Twairesh, & Alsanad, 2023; He, Xia, et al., 2023; Yang, Ding, Zheng, Xu, & Li, 2023). The essence of UDA is to utilize the knowledge from the source domain to improve the performance in the target domain, despite the distinct distributions between the two domains. Due to its ability to reduce the costs of model development and training in practice, the strategy of UDA is extensively used in fields such as computer vision (Oza, Sindagi, Sharmini, & Patel, 2023) and text categorization (Trung, Van, & Nguyen, 2022). Given the ubiquity of time series data across numerous applications and the intricate complexities inherent within it, investigating the unsupervised time series domain adaptation (UTSDA) has become a meaningful task (Ozyurt, Feuerriegel, & Zhang, 2023). Time series data can vary significantly due to differences in data collection methods among different domains. This variation often results in distinct data distributions, making

* Corresponding authors.
E-mail addresses: dtclee1222@gmail.com (D. Li), jianwang@whu.edu.cn (J. Wang), bingli@whu.edu.cn (B. Li).

https://doi.org/10.1016/j.ipm.2023.103638
Received 1 November 2023; Received in revised form 13 December 2023; Accepted 30 December 2023
Available online 9 January 2024

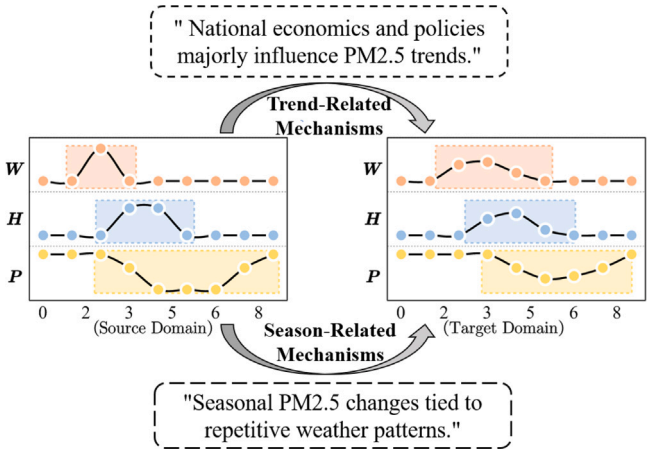

Fig. 1. Illustration of the invariant mechanism in the process of adapting from the source domain to the target domain. The uppercase bold letters represent wind speed (W), humidity (H), and PM2.5 concentration (P), respectively. In the short-term time series, there is a very clear relationship where an increase in "wind speed" and "humidity" leads to a decrease in "PM2.5 concentration. The colored blocks represent the dynamic nature of these variables, indicating that they are constantly changing. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

model generalization a challenging endeavor. For example, in medical time series data, the differences in data collection methods among different clinical sites (domains) result in distinct data distributions (He, Queen, et al., 2023). Consequently, a model trained on data from one clinic may perform poorly when tested on data from another clinic.

The main idea of UTSDA is to learn the invariant information across domains, where invariant mechanisms represent the underlying patterns and relationships that directly govern the behavior of time series data (Li et al., 2022). However, transferring invariant information across domains remains an open question. This is primarily because of the unique characteristics of time series data, including time-dependency, non-stationarity, and sequence interaction (Li et al., 2022; Ragab, et al., 2023). As illustrated in Fig. 1, the three time series exhibit non-stationary patterns. Nevertheless, there is an underlying causal mechanism linking variables such as wind speed (W), humidity (H) and PM2.5 concentration (P). This mechanism suggests that PM2.5 concentration is directly influenced by past values of temperature and wind speed, and this non-linear dependence is particularly evident in the short-term span shown in the figure. In addition, the historical trends of the data can also influence the future time series patterns. This type of time-dependent pattern is often observed in time series data. Furthermore, time series data also contain various types of noise due to the inherent uncertainty in the data generation processes, data entry errors or external factors (He, Queen, et al., 2023). This noise needs to be carefully taken into account in many time series tasks. Consequently, learning invariant mechanisms across domains is a critical task in UTSDA.

Currently, representation learning has gained significant attention due to its capacity to generate high-dimensional representation spaces with enhanced expressiveness and generalizability (Liu, Lin, & Sun, 2023). Some approaches (He, Queen, et al., 2023) have been proposed for UTSDA that leverage representation learning and cross-domain feature alignment. The main goal of these methods is to make the representation spaces from the source and target domains more alike. This helps models trained on the source domain work well in the target domain (Jin, Park, Maddix, Wang, & Wang, 2022; Kang, Jiang, Yang, & Hauptmann, 2019). For example, Deep CORAL (Sun & Saenko, 2016) is an extension of the original CORAL (Sun, Feng, & Saenko, 2016) method that aligns the second-order statistical properties in source and target distributions using a nonlinear transformation within deep neural networks. This enhancement leads to superior performance on standard datasets. AdvSKM (Liu & Xue, 2021) improves domain adaptation by leveraging a hybrid spectral kernel network to redefine the MMD metric, addressing the limitation of low-order and local statistics in discrepancy metrics. Furthermore, SASA (Cai et al., 2021; Li et al., 2022) utilizes sparse attention mechanisms to uncover associative patterns in time series data and presents an innovative model for aligning these sparse associative structures to facilitate domain adaptation.

Despite the success achieved by the aforementioned approaches, several challenges still exist in UTSDA. On one hand, time series data contain intricate components (e.g., trend, seasonality, noise, and nonlinear dependencies), posing challenges for traditional single-task representation learning methods, which may struggle to accurately encapsulate all of these aspects simultaneously (Fu, 2011). On the other hand, some previous works overlooked time series' inherent separable trend and seasonal components. They directly modeled the original data to create domain-invariant spaces, which might fail to capture the underlying mechanisms related to trends and seasonal patterns within that domain-invariant space. For instance, as shown in Fig. 1, the economic development level of a country might consistently influence the long-term trend of PM2.5 concentrations across its various regions. Therefore, extracting information from the long-term trend components in time series data can better reflect the mechanisms that influence the data's long-term trajectory. However, in the short term, changes in PM2.5 concentrations in each region are likely influenced by factors like rainfall or wind speed. The nonlinear relationships between these factors might be more evident due to the absence of smoothing or averaging effects in these short-term seasonal time series components. As a result, this underscores the necessity for

methods in time series DA methods that **(1)** possess stronger representation learning capabilities to handle complex time series data, and **(2)** construct a domain-invariant space that preserves both trend-related and season-related mechanisms across the domains.

In this paper, inspired by relevant disentangled representation learning work (Lv et al., 2022; Yang et al., 2021), we propose a method called "Multi-Task Disentangled Learning Representations" (MDLR) to obtain a better domain-invariant space. Disentangled representations can effectively separate mixed features in the data into independently interpretable components. Moreover, considering the well-separated nature of time series data between trend components and seasonal components, we employ disentangled representations to extract trend-related and season-related domain-invariant mechanisms more finely. This forms the core motivation behind the MDLR method. Specifically, we leverage causal reasoning theory to theoretically analyze the causal relationships between elements such as trend and season in time series domain adaptation task, revealing the importance of different aspects of domain-invariant information contained in trend and season components of time series data. This provides a solid foundation for our disentangled representation learning method. To translate the theoretical concept into practical application, we design a multi-task learning representations framework that separately extracts trend-related and season-related domain-invariant information after performing causal intervention on the original time series data. Specifically, we simplify the do-operation in causal intervention by separating the original time series into trend and season components. This simplification allows us to perform feature extraction independently on these separated components and ensures that the underlying domain-invariant mechanisms at both long and short-term scales remain disentangled, avoiding their entanglement in the learned representations. Finally, we incorporate labeled classification and domain classification tasks in the framework to enable iterative training of the entire model. By doing so, the model can effectively learn disentangled representations that capture different mechanisms and achieve superior performance in UTSDA.

1.1. Research objective

The primary goal of our proposed method, MDLR, is to enhance the effectiveness of UTSDA by learning disentangled representations that separately capture trend-related and season-related domain-invariant information, thereby ensuring more robust and accurate model performance across diverse domains. To summarize, our main contributions can be summarized as follows:

- We provide a new theoretical perspective on the time series domain adaptation problem via a causal perspective and argue for the benefits of learning disentangled domain-invariant representations for UTSDA.
- We propose MDLR, a multi-task disentangled learning approach that utilizes a dual-tower architecture to learn invariant information across different domains. This invariant information encapsulates the trend-related and season-related mechanisms.
- We demonstrate the superiority of our approach compared to state-of-the-art methods through extensive experiments on real-world time series data. Furthermore, visualization results affirm that the effectiveness of learning trend-related and season-related representations for the UTSDA task.

The paper is structured as follows: Section 2 reviews related work on unsupervised domain adaptation and disentangled representation in time series data. Section 3 presents a causal perspective on UTSDA, laying the foundation for MDLR. Section 4 details the MDLR model's architecture and methodology with a dual-tower approach. Section 5 showcases experimental results, comparing MDLR to baseline methods. Section 6 analyzes the results and underscores MDLR's significance. Finally, Section 7 concludes by summarizing our work and suggesting future research directions.

2. Related work

Our work focuses on unsupervised time series domain adaptation (UTSDA), which mainly focusing on three related main areas: domain adaptation, time series domain adaptation, and disentangled learning representation. In the following subsections, we highlight some methods that are closely related to our work.

2.1. Unsupervised domain adaptation

Unsupervised Domain Adaptation (UDA) is a form of transfer learning designed to tackle the "domain shift" issue (Nozza, Manchanda, Fersini, Palmonari, & Messina, 2021). This approach involves adjusting a model, initially trained in one specific domain (source domain), to function efficiently in a different domain (target domain), without the need for labeled data from this target domain (He, Queen, et al., 2023; Liu & Xue, 2021; Ramponi & Plank, 2020). While various UDA techniques have been developed for visual applications, they generally focus on deriving domain-invariant features using feature extraction networks (Li et al., 2022). Among these techniques, the adversarial-based approach has emerged as particularly significant. Within this framework, a domain discriminator is tasked with identifying whether data originates from the source or target domain. Concurrently, a deep learning classifier is trained to produce features that are indistinguishable to the discriminator, effectively bridging the gap between the two domains (Hoffman et al., 2018; Tzeng, Hoffman, Darrell, & Saenko, 2015). For example, TA3N (Chen et al., 2019) achieved better performance by utilizing domain discrepancy on four video domain adaptation datasets. In addition to the adversarial-based approach, another popular method utilizes the discrepancies between the feature representations of different domains to facilitate more efficient domain adaptation. This discrepancy-based approach is epitomized by techniques like Maximum Mean Discrepancy (MMD) (Rozantsev, Salzmann, & Fua, 2018), which has been used in strategies like the adaptation layer in Deep Domain

Confusion (DDC) (Tzeng, Hoffman, Zhang, Saenko, & Darrell, 2014) to enhance classification effectiveness while maintaining domain consistency. For example, ODIN (Hussein & Hajj, 2022) introduces an SPS domain adaptation architecture that utilizes MMD to model the adaptation loss function. Other commonly used metrics include Correlation Alignment (CORAL) (Sun & Saenko, 2016) and Contrastive Domain Discrepancy (CDD) (Kang et al., 2019).

Adversarial and discrepancy-based strategies are the two main approaches to the challenge of domain shift. While adversarial methods are often plagued by unstable training dynamics, discrepancy-based methods such as MMD offer a more stable alternative. MMD is characterized by its direct estimation of distributional variances, which streamlines both the implementation and training processes. Given these advantages, our MDLR uses MMD as the key metric for assessing and mitigating distributional discrepancies.

2.2. Time series domain adaptation

The special characteristics of time series data have led to a relatively limited exploration of self-supervised domain adaptation for such data types (He, Queen, et al., 2023). Drawing inspiration from established approaches in visual unsupervised domain adaptation, these methods focus on training models to create general and strong features suited for the complex nature of time series data (He, Queen, et al., 2023; Jin et al., 2022). One notable method, CoDATS (Wilson, Doppa, & Cook, 2020), employs adversarial training and utilizes a Convolutional Neural Network (CNN) for feature extraction, which offers flexibility in handling diverse complexities of the data. VRADA (Purushotham, Carvalho, Nilanon, & Liu, 2016) took the route of a variational recurrent neural network for feature extraction, demonstrating the potential of recurrent networks in this domain. Another noteworthy approach is AdvSKM (Liu & Xue, 2021), which focuses on minimizing statistical divergence and argues that MMD may not be robustly applied to UTSDA. As a solution, it proposes an Adversarial Spectral Kernel Matching (AdvSKM) method for UTSDA. Furthermore, there are methodologies like SASA (Cai et al., 2021) which have worked on aligning both intra- and inter-variable attention using MMD. And SEA (Wang et al., 2023) efficiently tackles MTS-UDA challenges by aligning sensor features at local and global levels using a multi-branch self-attention mechanism for domain adaptation.

The existing methods in UTSDA primarily emphasize aligning features at the level of their characteristics, without fully exploring the potential of disentangled learning representations for facilitating easier alignment across varying domain spaces. However, our MDLR approach sets itself apart by paying attention to the trend and seasonality elements of time series data. This strategy allows for a more effective disentanglement of the intrinsic mechanisms involved in domain adaptation, leading to potentially more robust and accurate alignment across different domains.

2.3. Disentangled learning representations

Disentangled learning representations aim to separate out the underlying causal factors of the data (Li et al., 2023; Wang, Chen, Zhou, Ma, & Zhu, 2022), which can facilitate more effective transfer and generalization of learning (Khemakhem, Kingma, Monti, & Hyvarinen, 2020). A wide range of techniques has been developed and proposed specifically to attain the objective of disentangled representations in various contexts. For example, Variational Autoencoders (VAEs) are usually used to employ a latent variable model for the disentanglement of features (Khemakhem et al., 2020; Liu, Wang, & Li, 2024). Another notable method is InfoGAN, which leveraging the maximization of mutual information between certain latent variables and observations to enhance its effectiveness (Chen et al., 2016). Due to the specific nature of time series, some work has also explored disentangled learning representations in this context. For instance, DTS (Li et al., 2021) is a novel framework for learning interpretable and disentangled time-series representations by addressing complex temporal correlations, KL vanishing, and multi-level disentanglement challenges, leading to superior performance in downstream applications with high interpretability. CoST (Woo, Liu, Sahoo, Kumar, & Hoi, 2022) is a work closely related to this paper. It utilizes contrastive learning for superior time-series forecasting via disentangled seasonal-trend representations. These methods demonstrate the promising potential of disentangled representations in time series analysis.

While methods using disentangled learning representations are frequently utilized for forecasting in time series, a notable oversight in many of these approaches, especially in the UTSDA task, is their neglect of inherent trend and seasonal patterns. These elements are critical yet are often overlooked despite their separability. To address this gap, our study presents the MDLR approach. This method specifically acknowledges the distinct features pertaining to trends and seasonality inherent in time series data. By strategically decomposing the time series into trend-based and senson-based components, MDLR simplifies the architecture of the representation layers. This simplification not only streamlines the design but also enhances the efficiency and effectiveness of information extraction from time series data.

3. Preliminaries

This section aims to establish the foundation for understanding domain adaptation in time-series data. Firstly, we will present the problem and define necessary terminology. Furthermore, we will explore the causal aspect of domain adaptation to elucidate our approach.

3.1. Problem formulation

In this section, we offer an in-depth exploration of fundamental domain adaptation concepts. We have organized key terminologies in Table 1, providing their definitions to aid understanding throughout this paper. Consider a dataset $D_s = \left[X_s, Y_s\right]$ from a

Table 1
Notations and explanations.

Notations	Explanations
D_s	Source domain's labeled data
X_s	Source domain time series
$P_s(X_s)/P_s$	Source data's marginal distribution
D_t	Target domain's unlabeled data
X_t	Target domain time series
$P_t(X_t)/P_t$	Target data's marginal distribution
Y_s	Labels in the source domain
K	Number of labels
$\mathbf{C}$	Underlying causal factors
S	Season-related causal factors
T	Trend-related causal factors
X_{trend}	Time series' trend component
X_{season}	Time series' seasonal component
g_1	Season feature extractor
g_2	Trend feature extractor
$\mathcal{L}_{CLS}$	Classification loss
$\mathcal{L}_{DOM}$	Domain classification loss
$\mathcal{L}_{T_{MMD}}$	MMD loss for trend
$\mathcal{L}_{S_{MMD}}$	MMD loss for season
h	Model function for label prediction
w_{season}, w_{trend}	Weights for season/trend factors in the classifier
α, β, γ	Parameters to balance loss components
κ	Kernel size in time series decomposition

source domain, with $X_s \in \mathbb{R}^{n_s \times d}$ representing a multivariate time series and $Y_s = \left(y_1, \ldots, y_{n_s}\right)^T \in \mathbb{R}^{n_s \times 1}$ as its corresponding labels. In contrast, the target domain has an unlabeled dataset, $D_t = [X_t]$, where X_t spans a $n_t \times d$ dimensional space. Both domains share an identical label space $Y = \{1, 2, \ldots, K\}$, with K representing the total number of distinct labels. Notably, the marginal distributions of the data from these domains differ, as signified by $P_s(X_s) \neq P_t(X_t)$ (Ragab, Eldele, Tan, et al., 2023). The objective of domain adaptation is to develop a resilient predictive model capable of accurately assigning labels to the target domain data, leveraging the invariant knowledge extracted from a labeled source domain (He, Queen, et al., 2023). Essentially, domain adaptation focuses on capturing the statistical correlation between observed inputs and their associated labels, symbolized by $P(Y|X)$. This relationship is typically assumed to change between different domains.

3.2. Overview of domain adaptation from causal view

Domain adaptation (DA) focuses on modeling the statistical association between observed inputs X and their respective labels Y, particularly $P(Y|X)$, which is subject to variation across different domains (Cai et al., 2021; Javidian, Pandey, & Jamshidi, 2021; Kisamori, Kanagawa, & Yamazaki, 2020). In the context of causal inference, domain adaptation predominantly addresses the issue of covariate shift, a scenario where the feature distribution varies among domains (Schölkopf et al., 2012). To address this challenge, we seek causally invariant features that remain consistent across domains. While we represent the underlying causal factors as $\mathbf{C}$, we lack prior knowledge about them. Instead, we are provided with unstructured time series data X. The Independent Causal Mechanisms (ICM) principle (Lv et al., 2022; Peters, Janzing, & Schölkopf, 2017) proposes that causal factors should be jointly independent. This principle enables the factorization of the joint distribution $p_{\tilde{\mathbf{C}}}\left(c_1, \ldots, c_D\right)$ of causal factors into conditionals:

$$p_{\tilde{\mathbf{C}}}\left(c_1, \ldots, c_D\right) = \prod_{i=1}^{D} p_i\left(c_i \mid \mathbf{c}_{pa(i)}\right), \tag{1}$$

where $p_{\tilde{\mathbf{C}}}$ represents the combined distribution of the causal elements $\tilde{\mathbf{c}}$, which includes various factors such as $\{c_1, c_2, \ldots, c_D\}$. Additionally, p_i indicates the specific distribution of each element c_i, conditioned upon its causal parents, which are represented by $\mathbf{c}_{pa(i)}$. It should be noted, however, that identifying all causal factors in time-series data presents significant challenges, even under the assumption that these factors operate independently. The task of domain adaptation can be simplified as a feature selection problem (He, Queen, et al., 2023). The objective is to find the best subset of predictors that keeps the conditional distribution of the target variable consistent across different domains.

In time-series analysis, decomposing data into seasonal and trend components helps distinguish short-term changes from long-term patterns, improving forecasting and data interpretation (Meng et al., 2023; Wen et al., 2022). In time series domain adaptation, the trend component provides us with long-term directional information, while the seasonal component helps us identify and understand short-term seasonalchanges and non-linear dependencies. Modeling these two components can capture different characteristics of time series data. In this paper, inspired by tasks related to time-series prediction (Chen et al., 2022; Woo et al., 2022) and considering the separability of seasonal and trend components in time series, we classify causal factors into two subsets. The first subset consists of season-related causal factors denoted by $S = \{c_1^s, c_2^s, \ldots, c_{h_s}^s\}$, and the second subset consists of trend-related causal factors denoted by $T = \{c_1^t, c_2^t, \ldots, c_{h_t}^t\}$. To represent the joint distribution of causal factors in a time series, considering

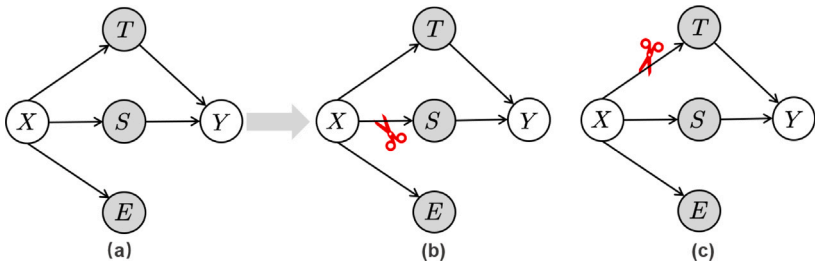

Fig. 2. Causal graph of the proposed model. White circles represent observed variables, while gray circles represent latent (unobserved) variables. In (a), the output Y is directly affected by the trend-related variable T and season-related variable S. The observed time series data X is derived from the interactions of S (season-related factors), T (trend-related factors), and E (error variable). The inference stage of the "do" operation is illustrated in (b) and (c).

the distinct nature of its seasonal and trend components, we can reformulate it as follows:

$$p_{\tilde{\mathbf{C}}}\left(c_1,\ldots,c_D\right)=\prod_{i=1}^{h_s}p_i\left(c_i^s\mid\mathbf{c}_{pa(c_i^s)}\right)\prod_{j=1}^{h_t}p_j\left(c_j^t\mid\mathbf{c}_{pa(c_j^t)}\right), \tag{2}$$

where pa$(\cdot)$ indicates the causal parents for a specified causal component. Instead of trying to rebuild all causal factors directly, our approach involves breaking down the time series data into components of seasonality and trend. We aim to efficiently train models focused on these trends and seasons to learn representations effectively. This approach will better capture the causal factors in X and help in generalizing across different domains when predicting Y.

In order to model the conditional probabilities $p\left(c^s\mid\mathbf{c}_{pa(c^s)}\right)$ and $p\left(c^t\mid\mathbf{c}_{pa(c^t)}\right)$, we draw inspiration from Bayesian Structural Time Series models (Qiu, Jammalamadaka, & Ning, 2018; Scott & Varian, 2015) and illustrate the generative process of the time series data through a causal graph, as depicted in Fig. 2(a). The time series data X that has been observed can be attributed to three distinct variables (Woo et al., 2022): the error variable E, the trend-related variable T, and the season-related variable S. As E is unpredictable noise, the primary prediction task Y depends only on T and S. Based on the causal graph, the following Structural Causal Model (SCM) is formulated to represent the process of generating time series data:

$$\begin{aligned}X&:=f\left(S,T,E\right),\\ Y&:=h\left(S,T\right)=h\left(g_1(X),g_2(X)\right),\end{aligned} \tag{3}$$

where f, g, and h are unknown structural functions. Structural Causal Model (SCM) (Schölkopf, 2022) provide a framework for describing how variables interact and influence each other, making them highly relevant in the context of domain adaptation. In this approach, we critically analyze and model the components of SCM, namely T, S, and E, to understand the causal relationships that help identify and adapt to the causal variations that exist across different domains. Furthermore, we formulate the optimization problem of UTSDA as follows:

$$\begin{aligned}h^*&=\arg\min_h\mathbb{E}_P[\ell(h(g_1(X),g_2(X)),Y)]\\&=\arg\min_h\mathbb{E}_P[\ell(h(S,T),Y)],\end{aligned} \tag{4}$$

where the function $\ell(\cdot,\cdot)$ symbolizes the cross-entropy loss. The focal point of this approach is to identify the function h that is most effective in reducing the expected loss, represented as $\mathbb{E}_P[\ell(h(g_1(X),g_2(X)),Y)]$. By identifying the stable causal factors T and S for different domain data or finding a function g that effectively maps the transformed inputs X to S and T, we can potentially enhance the model's proficiency in performing the UTSDA task.

4. Methodology

In this section, we explore the details of our proposed MDLR method for unsupervised time series domain adaptation (UTSDA). Specifically, we will first present an overview of MDLR, including its overall architecture and loss function. Subsequently, we will delve into the fundamental design details of MDLR.

4.1. Overview of MDLR

In this paper, we consider the separability of trend and seasonal components in time series data, which represent unique aspects in the process of generating time series data. For example, as shown in Fig. 1, the long-term trend can reflect macroeconomic conditions such as economic policies, while seasonal components can reflect the sequence interaction mechanisms among multidimensional time series. This determines that we can decompose domain-invariant causal factors into two subsets: season-related and trend-related factors. Therefore, guided by structural causal model (SCM) (Schölkopf, 2022), we assume that the trend-related variable T and the seasonal-related variable S causally affect the category label Y, as depicted in Fig. 2. In UTSDA, Our objective is to

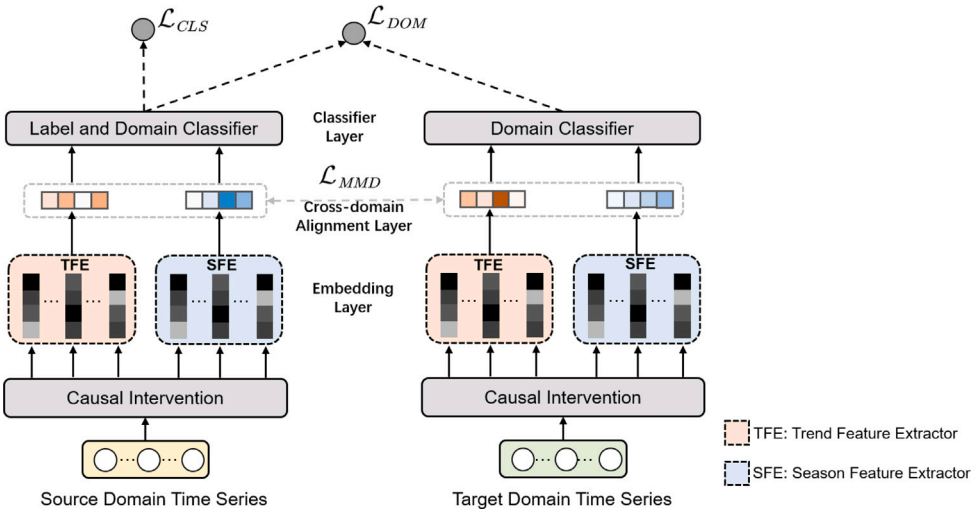

Fig. 3. Overall architecture of MDLR. The architecture features a dual-tower structure, each dedicated to the trend and seasonal components of the time series. Both towers process data through causal intervention, embedding, cross-domain alignment, and classifier layers.

identify and extract the latent causal elements represented by S and T from the raw input X. These two causal factors capture different aspects of the underlying domain-invariant representations. To achieve this, we adopt a dual-tower architecture inspired by multi-task learning to effectively handle complex time series data, as illustrated in Fig. 3.

The MDLR framework comprises two separate towers, each dedicated to extracting trend-related and season-related domain-invariant representations. This design choice is motivated by the need to handle the complexities and distinct nature of trend and seasonal data in a more focused and effective manner. Additionally, both towers are structured with four distinct layers. In the causal intervention layer, we use the Moving Average (MA) algorithm to decompose the input time series data $X \in \mathbb{R}^{n\times d}$ into trend component $X_{\text{trend}} \in \mathbb{R}^{n\times d}$ and season component $X_{\text{season}} \in \mathbb{R}^{n\times d}$. In the embedding layer, a season feature extractor (SFE), $g_1 : \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times d_S}$, extracts season-related patterns from X_{season}, while a trend feature extractor (TFE), $g_2 : \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times d_T}$, extracts trend-related patterns using an encoder backbone to transform the trend components data into a latent space. The separation of these extractors within the dual-tower architecture is a strategic decision to optimize the individual processing of each component, thereby enhancing the overall accuracy and robustness of the model. In the cross-domain alignment layer, the alignment of trend-related and season-related representations across various domains is achieved using the Maximum Mean Discrepancy (MMD) method. Finally, the classifier layer outputs the results for both the label and domain classification tasks by concatenating the separately trained outputs of the classifier module on the trend-related and season-related representations. The overall loss function for the MDLR framework is given by

$$\mathcal{L} = \alpha\mathcal{L}_{CLS} + \beta\mathcal{L}_{DOM} + \frac{\gamma}{2}(\mathcal{L}_{T_{MMD}} + \mathcal{L}_{S_{MMD}}), \tag{5}$$

where α, β, and γ are hyperparameters that balance the trade-off between all loss factors.

Next, we detail the core layers of the MDLR architecture.

4.2. The causal intervention and embedding layers

Existing domain adaptation methods model the probability $P(Y|X)$ to adapt to different domains. As discussed in the preliminaries section, we can divide the numerous domain-invariant factors into trend-related causal factors S and season-related causal factors T, which directly influence the output variable Y. Therefore, in the UTSDA task, the conditional distribution of input time series X and output Y can be further represented as:

$$P(Y|X) = P(Y|S,T). \tag{6}$$

As described in Eq. (3), the functions g_1 and g_2 serve the purpose of extracting the season-related and trend-related mechanistic information from the original time series X. Furthermore, according to the independent mechanisms assumption, it is clear that the modules representing seasonal and trend factors operate independently, without influencing each other or sharing information. In practical applications, deep neural network models are typically used to learn and represent these causal factors indirectly, rather than reconstruct them directly.

To proceed, we need to design separate models for extracting the invariant season-related mechanisms and trend-related mechanisms among different domains. As shown in Fig. 2(b), to effectively model the season-related factors S, we need to eliminate the bias introduced by the trend-related information T during the training period. Inspired by the use of causal intervention (Lv

et al., 2022; Woo et al., 2022), which have been extensively utilized to enhance the robustness of deep learning models across various tasks, we use the "do" operation to exclude the influence of the confounder T on the impact of the seasonal factors S on the model output Y. Specifically, by using the "do" operation to prevent the introduction of causal effects of T on Y, we can derive the causal effects of the target feature S on the model output Y as follows:

$$P(Y \mid do(S)) = \sum P(Y \mid S, t)\, P(t), \tag{7}$$

where $t \in T$ corresponds to a trend-related factor. Since T cannot be directly observed, and directly computing the above expression would incur a large sampling cost, we employ the Normalized Weighted Geometric Mean (NWGM) (Xu et al., 2015; Zhang, Sang, Wang, Jiang, & Wang, 2023) as an approximation technique. NWGM has been widely utilized in various causal inference intervention calculations. To infer $P(Y \mid do(S))$, we utilize the intervention feature $\mathbb{E}[t]$ to substitute specific trend features t. The expression can be rewritten as follows:

$$P(Y \mid do(S)) = P(Y \mid S, \mathbb{E}[t])), \tag{8}$$

where $\mathbb{E}[t]$ is the mathematical expectation of t. Similarly, the trend-related factor intervention $P(Y \mid do(T))$ can be calculated.

Inspired by the work (Wang, Feng, He, Zhang, & Chua, 2021), we implement the intervention in our model by excluding the trend-related factor t from the model's input. This is equivalent to setting $\mathbb{E}[t]$ to 0. As a result, the resulting model is specifically trained to predict $P(Y \mid do(S))$, relying exclusively on season components of time series. This allows us to eliminate any potential bias introduced by the trend-related features. In the causal intervention layer, we employ a moving average (MA) strategy (Hyndman, 2011) to mitigate seasonal variations and emphasize long-term patterns in time series data inspired by the idea of time series decomposition (Cleveland, Cleveland, McRae, & Terpenning, 1990; Dudek, 2023; Woo et al., 2022). MA is a statistical way to smooth data by calculating the average of a set of data points over a specific time period (window) and repeating the calculation as the window slides along the data. Specifically, given an input time series of length n, $X \in \mathbb{R}^{n \times d}$, the process is as follows:

$$\begin{aligned} X_{\text{trend}} &= \text{AvgPool}(\text{ Padding }(X)), \\ X_{\text{season}} &= X - X_{\text{trend}}, \end{aligned} \tag{9}$$

where X_{season} and X_{trend} are matrices in $\mathbb{R}^{n \times d}$, representing the season and trend components. The $\text{AvgPool}(\cdot)$ function with padding is used to smooth the data and maintain the original length of the time series. To capture information from different time periods as much as possible, different kernel sizes can be used. We introduce the $\text{SeriesDecomp}(\cdot)$ function to summarize the above equations, and the results of different kernel sizes κ are given by:

$$X^{\kappa}_{season}, X^{\kappa}_{trend} = \text{SeriesDecomp}(X, \kappa). \tag{10}$$

To better represent the seasonal and trend components extracted from the time series, we utilize the capabilities of Convolutional Neural Networks (CNNs) in the embedding layer. CNNs are adept at capturing local patterns in data, which is particularly beneficial for time series analysis (Ragab, Eldele, Tan, et al., 2023). Given the seasonal component X^{κ}_{season} and the trend component X^{κ}_{trend} for each kernel size κ, we utilize a CNN backbone to extract features:

$$\begin{aligned} S^{\kappa} &= \text{CNN}(X^{\kappa}_{season}), \\ T^{\kappa} &= \text{CNN}(X^{\kappa}_{trend}). \end{aligned} \tag{11}$$

Here, the features extracted from the seasonal and trend components for a kernel size of κ are represented as S^{κ} and T^{κ}, respectively. These feature extractors form the basis of the trend feature extractor (TFE) and the season feature extractor (SFE). The primary difference between TFE and SFE lies in their algorithmic implementation, which is primarily determined by their hyperparameters. The TFE is designed to identify and capture long-term patterns and shifts in the time series data, and the SFE is tailored to detect and analyze seasonalpatterns and cyclic behaviors inherent in the data. By jointly considering both seasonal and trend components from different scales, the model becomes better equipped to learn a domain-invariant feature space.

4.3. The cross-domain alignment layer

Unsupervised domain adaptation aims to create a representation space where data from various domains can be consistently and comparably represented. One common approach to achieve this goal is to reduce the divergence in distance between data originating from the source domain and that from the target domain. The Maximum Mean Discrepancy (MMD) metric (Gretton, Borgwardt, Rasch, Schölkopf, & Smola, 2006) provides an effective tool for quantifying the difference between two data distributions. This measure fundamentally contrasts the average values of samples within a reproducing kernel Hilbert space. By minimizing this measure, it ensures that the two distributions are made as similar as possible within this space. In this paper, we also introduce MMD metric to balance the distributions between different domains. From a causal inference perspective described above, domain-invariant factors across domains can be dissected into two main factors: trend-related and season-related causal factors. While season-related features in time series data often exhibit more microscopic invariant patterns across various domains, such as the nonlinear dependencies between different time sequences, these dependencies might not be evident on larger time scales but can be very pronounced on smaller scales. On the other hand, trend-related features reflect more macroscopic domain-variant factors. For instance, when people engage in physical activities, vital signs like heart rate and respiration rate might indeed show an increasing

Table 2
Overview of utilized datasets detailing number of domains, sensor channels, classes, and total samples in training and test portions.

Dataset	Domains	Classes	Channels	Training set	Testing set
UCIHAR (Anguita, Ghio, Oneto, Parra Perez, & Reyes Ortiz, 2013)	30	6	9	2300	990
WISDM (Kwapisz, Weiss, & Moore, 2011)	36	6	3	1350	720
HHAR_SA (Stisen et al., 2015)	9	6	3	12,716	5218

trend until they stabilize. Considering this observation, we propose a dual-alignment approach, where trend-related and season-related representations are aligned separately across the source and target domains. This dual alignment is encapsulated in our loss function, which comprises two distinct MMD terms: $\mathcal{L}_{T_{MMD}}$ for trend-related alignment and $\mathcal{L}_{S_{MMD}}$ for season-related alignment.

$$\mathcal{L}_{\text{MMD}} = \mathcal{L}_{T_{MMD}} + \mathcal{L}_{S_{MMD}}. \tag{12}$$

4.4. The label and domain classifier layer

Unsupervised domain adaptation is a powerful approach that leverages knowledge derived from a known domain to enhance performance on a new, previously unseen domain. In most domain adaptation tasks, there is some supervision information available, including partial labels of the target domain, domain information, or other forms of partial supervision information. Among these, label and domain information are two important types of information that can be utilized, and they are widely used in transfer learning (Woo et al., 2022). Therefore, we design a label and domain classifier layer to train the UTSDA model effectively. In our label classifier, a deep neural network architecture is employed. This network accepts a related representation as its input and yields a prediction for the label linked to the source data. As previously discussed, the prediction label can be formulated as:

$$Y = w_{trend}\text{soft}(\text{MLP}(T^{\kappa})) + w_{season}\text{soft}(\text{MLP}(S^{\kappa})), \tag{13}$$

Where soft$(\cdot)$ represents the softmax function applied to the task classifier's output. w_{trend} and w_{season} are the weights assigned to trend-related and season-related factors. The definition of the cross-entropy loss for the label classifier module is as follows:

$$\mathcal{L}_{CLS} = -\mathbb{E}_{X_s \sim P_s}[Y_s \log(h(g_1(X_s), g_2(X_s)))] \tag{14}$$

where notation P_s symbolizes the distribution of data X_s from the source domain, while Y_s signifies the real labels associated with the source domain. In this paper, the domain classifier operates as an adversarial component during the training process and consists of a multilayer perceptron (MLP) architecture. Specifically, it is trained to determine from which domain the input data originated. Trend and seasonal features, as important components within time series data, play a significant role in enhancing the performance of the domain classifier. Trend features, in particular, exhibit lower levels of noise. In our practice, we have observed that utilizing trend features leads to greater stability and more accurate domain classification. The loss of the domain classifier is also calculated using the cross-entropy loss, and the predicted domain result Y_{dom} can be formulated as:

$$Y_{dom} = \text{soft}(\text{MLP}(g_2(X))). \tag{15}$$

5. Experiment

To assess the MDLR model's performance in real-world scenarios, we conducted a series of experiments in this section. We first introduce the experimental setup, which includes datasets, baseline methods, and evaluation metrics used to assess the effectiveness of the UTSDA methods. Subsequently, we present the specific experimental results, including comparisons with baseline methods, ablation visualization results, and hyperparameter κ analysis results.

5.1. Experimental setup

5.1.1. Dataset overview

In this section, our method's performance is evaluated using popular time series datasets, chosen from real-world scenarios as noted by ADATIME (Ragab, Eldele, Tan, et al., 2023). These datasets play a crucial role in evaluating models within Unsupervised Time Series Domain Adaptation (UTSDA), offering diverse challenges and real-world applicability. Detailed summaries of these datasets can be found in Table 2, and their descriptions are as follows:

- **UCIHAR** (Anguita et al., 2013) gathers data from accelerometers, gyroscopes, and body sensors across 30 subjects performing six distinct activities. It uniquely treats each subject as a separate domain, offering a diverse perspective on human activity recognition. Our evaluation focuses on five randomly selected cross-domain scenarios, aiming to assess the model's adaptability in handling activity patterns.
- **WISDM** (Kwapisz et al., 2011) includes accelerometer data from 36 subjects undertaking activities identical to those in UCIHAR. Its significance lies in the inherent class imbalances, offering a rigorous test for models in handling diverse, real-world data distributions.
- **HHAR_SA** (Stisen et al., 2015) or Heterogeneity Human Activity Recognition, involves data gathered from nine individuals using smartphone and smartwatch sensors.

5.1.2. Baselines

In our study, our primary goal is to conduct a thorough evaluation of the effectiveness of our multi-task disentangled learning representations (MDLR) approach. In order to achieve this goal, we have carefully selected a set of baseline methods covering a wide range of strategies in the context of UTSDA. These selected baseline methods play a crucial role in evaluating how well our proposed approach performs in different facets of domain adaptation. Below, we provide a more detailed explanation of the rationale behind the selection of each of these baseline methods:

- **LSTM** (Hochreiter & Schmidhuber, 1997) is a type of network good at understanding long-term dependencies in sequential data. Its primary strategy involves learning from the source domain and directly applying the learned model to the target domain without specific adaptation techniques. We include LSTM as a baseline due to its widespread use and effectiveness in capturing long-term dependencies in time series data. It provides a fundamental benchmark for evaluating more complex adaptation techniques.
- **Deep CORAL** (Sun & Saenko, 2016) primarily focuses on aligning the second-order statistics (covariances) of the source and target domain features in a deep neural network. This alignment is achieved by minimizing a loss function that measures the difference in covariances between the two domains. By doing so, Deep CORAL effectively reduces the domain shift, enabling the neural network to generalize better to the target domain, even in the absence of labeled data in that domain. This method is both efficient and effective, making it a valuable tool in the field of UTSDA.
- **CoDATS** (Wilson et al., 2020) excels in improving accuracy and reducing training time, particularly in scenarios with varying data availability across domains. It leverages data from multiple sources, enhancing its effectiveness for complex datasets with high inter-domain variability. Additionally, the model incorporates the Domain Adaptation with Weak Supervision (DA-WS) method, utilizing weak supervision through target-domain label distributions, setting a new standard in time series domain adaptation.
- **SASA** (Cai et al., 2021; Li et al., 2022) offers a novel solution for domain adaptation in time series data. This method includes adaptive segment summarization to handle offsets, and employs both intra- and inter-variable sparse attention mechanisms to capture associative structures considering time lags. The alignment of these structures facilitates effective knowledge transfer, showcasing SASA's innovative approach to overcoming the complexities in time series domain adaptation.
- **AdvSKM** (Liu & Xue, 2021) employs adversarial spectral kernel matching, a technique that aligns the spectral distributions of source and target domains to facilitate domain adaptation. The approach is particularly effective in scenarios where labeled data is scarce or unavailable in the target domain. By leveraging the spectral properties of time series data and adversarial learning, this method significantly enhances the adaptability and performance of time series models across different domains.
- **CoTMix** (Eldele et al., 2023) addresses the challenge of domain shift between a labeled source domain and an unlabeled target domain. CoTMix uses contrastive learning and a temporal mixing strategy to create intermediate augmented views for both domains, effectively reducing the distribution shift. This approach maximizes the similarity between each domain and its augmented view, guiding both towards a common intermediate space.

5.1.3. Model variants

In order to gain deeper insights into the impact and importance of various components within our proposed framework, we introduce and evaluate several model variants. This involves creating and analyzing various model adaptations, each designed to highlight the individual contribution of a specific component to the overall effectiveness of the model. This subsection details the specific structure and purpose behind each variant:

- **MDLR** is our original model proposed in this paper.
- **MDLR-tr** eliminates the component that extracts season-related features to determine the significance of trend-related feature information.
- **MDLR-sea** only utilizes season-related information instead of both trend-related and season-related information to verify its usefulness.

5.1.4. Evaluation metric

Due to the fact that certain classes may be entirely absent from certain subjects, the accuracy metric alone may not provide a representative evaluation in such cases (Ragab, Eldele, Tan, et al., 2023). As a result, we use the Macro F1 (MF1) score for a more balanced evaluation of our methods. The MF1 score averages the F1 scores across different classes, ensuring a fair assessment even when some classes are absent. The MF1 is calculated as:

$$\mathrm{MF1} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{F1}_i, \tag{16}$$

where N represents the total number of classes, and $\mathrm{F1}_i$ denotes the F1 score calculated for the $i-$th class.

5.1.5. Implementation details

The comparison algorithms in our study primarily rely on two key approaches: ADATIME (Ragab, Eldele, Tan, et al., 2023) and SASA (Cai et al., 2021). We obtain our experimental data from the ADATIME research, which involves splitting each dataset into

Table 3
Comparison of the MF1 score over the UCIHAR dataset.

Method	2→11	12→16	9→18	4→9	27→3	Avg
LSTM (Hochreiter & Schmidhuber, 1997)	44.2652(4.4149)	41.8773(5.8167)	34.1139(2.2722)	38.5430(9.2611)	45.9549(1.8307)	40.9508
Deep Coral (Sun & Saenko, 2016)	99.6258(0.6479)	58.6122(0.4840)	48.0212(12.4747)	43.0574(12.4843)	90.9798(7.8535)	68.0592
AdvSKM (Liu & Xue, 2021)	93.5802(11.1193)	57.3007(1.0508)	42.5054(6.0673)	60.1062(6.2722)	78.7377(6.3465)	66.4460
CoDATS (Wilson et al., 2020)	100(0)	63.1813(1.7432)	76.0955(4.0992)	62.1657(7.7272)	96.2181(4.9818)	79.5321
SASA (Cai et al., 2021; Li et al., 2022)	59.0502(2.3981)	57.3815(0.7630)	53.7608(3.5208)	28.5359(3.2997)	56.7476(4.0086)	51.0950
CoTMix (Eldele et al., 2023)	92.7723(0.6454)	75.8572(3.0900)	85.0630(1.6305)	57.6395(1.4525)	97.9336(0.0507)	81.8531
MDLR	100(0)	65.2370(6.4914)	77.1725(5.0914)	70.3994(14.3148)	98.7233(0.5561)	**82.3064**

Table 4
Comparison of the MF1 score over the WISDM dataset.

Method	7→18	20→30	35→31	6→19	23→31	Avg
LSTM (Hochreiter & Schmidhuber, 1997)	24.4478(0.3814)	24.9676(3.4875)	37.2910(1.0819)	24.2355(0.7308)	20.6698(6.6812)	26.3223
Deep Coral (Sun & Saenko, 2016)	56.0187(2.1786)	63.9740(4.9646)	46.7507(19.1594)	57.7043(5.75693)	49.2834(7.7823)	54.7462
AdvSKM (Liu & Xue, 2021)	45.9945(0.6307)	69.2655(4.0615)	46.3613(3.9359)	58.7460(6.4677)	55.6449(6.1024)	55.2024
CoDATS (Wilson et al., 2020)	45.7402(5.5218)	67.9711(11.1440)	67.7228(0.9210)	54.3348(11.9263)	44.9203(12.4757)	56.1378
SASA (Cai et al., 2021; Li et al., 2022)	57.7813(1.0139)	60.7300(3.2919)	56.6333(1.9611)	52.9275(2.0720)	47.3673(6.2272)	55.0878
CoTMix (Eldele et al., 2023)	50.8580(11.1642)	69.2431(5.5169)	62.7697(2.8024)	52.3571(6.7929)	55.7332(5.4097)	58.1922
MDLR	57.9102(6.0991)	71.8890(6.1517)	62.7448(6.4254)	74.9741(14.5675)	59.5477(3.6414)	**65.4131**

training, validation, and test sets. The experiments are conducted on a robust computational platform featuring a 12th Gen Intel(R) Core(TM) i5-12490F processor, complemented by 16 GB of RAM. The system is further enhanced with a high-performance NVIDIA GeForce RTX 3060 12G GPU, ensuring efficient processing of complex algorithms. We maintain a consistent batch size of 32 across all datasets throughout the experiments. To optimize our model, we employ the Adam optimizer with a consistent weight decay of 0.0001. Regarding the model parameters, we set $w_{trend} = w_{season} = 0.5$, indicating that equal weights are assigned to the trend-related and season-related predictors in the label classifier. We choose $\alpha = 100$, $\beta = 1$, and $\gamma = 1$ to balance the trade-off between all loss factors. Furthermore, we select only one kernel size and maintain a consistent kernel size parameter (κ) 8 for all three datasets. To ensure the reliability and robustness of our results, we randomly select five adaptation problems/tasks. For each task, we compute the average performance on the test set across three separate sets of network weights, all initialized randomly.

5.2. Experimental results and analysis

5.2.1. Overall comparison

Results on UCIHAR. Table 3 illustrates an comprehensive overview of different UTSDA methods on the UCIHAR dataset. The MDLR method significantly outperformed other state-of-the-art models, with a remarkable average MF1 score of 82.3064. Among other methods, CoTMix performed almost as well as MDLR with an average score of 81.8531, demonstrating its ability to adapt during domain shifts. LSTM has the lowest average score of 40.9508. The AdvSKM and Deep Coral methods have intermediate levels of accuracy, while the SASA method has the lowest average accuracy among these intermediate methods. Overall, these findings highlight the effectiveness and adaptability of MDLR and CoTMix, making them top contenders for UCIHAR dataset tasks.

Results on WISDM. The comparison of the MF1 score over the WISDM dataset clearly indicates the strength of our proposed model. Table 4 presents the mean values and standard deviations of the outcomes for various methodologies. Notably, our MDLR model attains superior F1 scores across numerous tasks, distinguishing itself in performance. For example, Task 20 → 30 achieves an F1 score of 71.8890 (with a standard deviation of 6.1517), and Task 6 → 19 attains 74.9741 (with a standard deviation of 14.5675). Although CoDATS obtains the highest score of 67.7228 in Task 35 → 31, MDLR closely follows with a score of 62.7448. Based on the average F1 scores, MDLR outperforms all other methods with an average score of 65.4131. The CotMix method, ranked second, achieved a score of 58.1922, exhibiting a noteworthy difference from the MDLR algorithm's performance. The average F1 scores further prove the robustness of our model in extracting invariant knowledge across diverse domains.

Table 5
Comparison of the MF1 score over the HHAR_SA dataset.

Method	8→1	0→2	5→8	7→6	5→3	Avg
LSTM (Hochreiter & Schmidhuber, 1997)	44.2900(0.3909)	56.3782(2.0068)	62.4528(5.2861)	62.1426(6.7426)	46.6353(2.0185)	54.3797
Deep Coral (Sun & Saenko, 2016)	93.7473(1.0487)	68.7375(2.5435)	96.5117(0.2691)	90.0865(8.1601)	95.2632(0.9637)	88.8692
AdvSKM (Liu & Xue, 2021)	88.5591(1.9946)	66.2225(1.8916)	85.4380(10.8185)	83.6562(0.8402)	74.7243(0.5439)	79.7200
CoDATS (Wilson et al., 2020)	76.5712(15.3432)	68.0878(5.3224)	92.9593(5.9070)	89.0866(2.6734)	83.1453(10.9483)	81.9700
SASA (Cai et al., 2021; Li et al., 2022)	64.8329(2.5699)	55.6107(1.5656)	75.7382(0.8887)	72.1470(0.3227)	69.8535(0.8078)	67.6364
CoTMix (Eldele et al., 2023)	73.1530(1.6602)	72.6165(0.8942)	98.9269(0.6092)	94.6952(0.8325)	67.1293(53.9731)	81.3041
MDLR	94.5543(1.9030)	73.7938(4.1498)	97.1164(1.8901)	89.8126(0.1527)	93.1630(2.3677)	**89.6880**

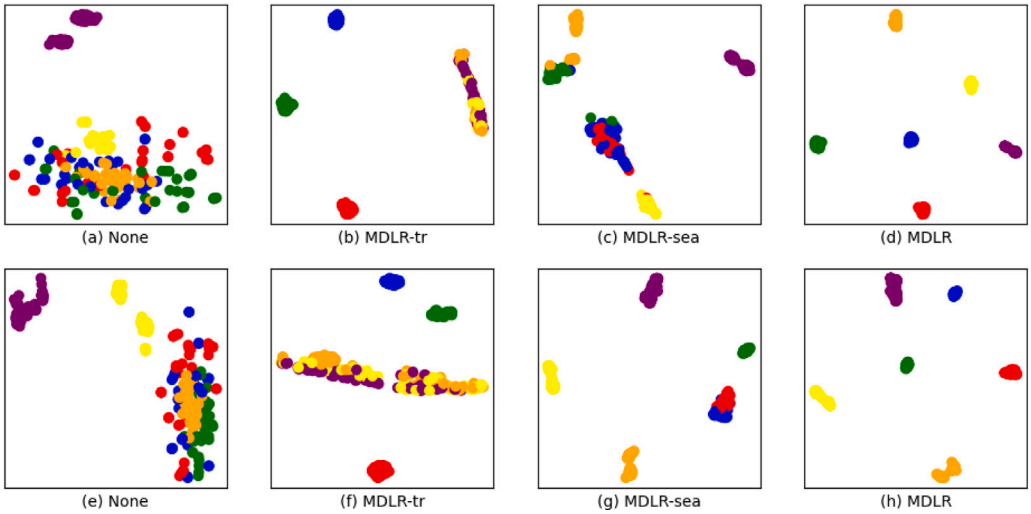

Fig. 4. Feature visualization on the UCIHAR 19 → 15 and 12 → 16 task. The t-SNE embedding features for tasks 12 → 16 and 19 → 15 are displayed in the upper and lower rows, respectively, with each color denoting a different category. The first column shows the original data's embedding features without any processing. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Results on HHAR_SA. The performance outcomes for different models on a time series domain adaptation task, utilizing the HHAR_SA dataset, are detailed in Table 5. It appears that MDLR and CoTMix are the top-performing methods, with relatively low standard deviations and high mean values across most pairs of input and output sequences. For example, MDLR has a mean value of 73.7938 for the 0 → 2 pair, which is the highest among all methods. On the other hand, LSTM appears to be the weakest performer, with mean values consistently lower than the other methods. In the 5 → 3 task, Deep Coral emerges as the leader, but it does not hold a significant advantage over MDLR. The CoTMix algorithm continues to show the best results in the 7 → 6 and 5 → 8 tasks, although the gap between it and MDLR is not very significant. In summary, MDLR and Deep Coral are the most effective methods for this task, with MDLR achieving the best performance.

5.2.2. Ablation study and visualization

We also utilize the t-SNE visualization tool to illustrate the effectiveness of trend-related and season-related information that MDLR has learned in the UTSDA task, thus further proving the rationality of our proposed method for learning disentangled season and trend representations. Specifically, we visualize the final features obtained by the variant models (MDLR, MDLR-tr and MDLR-sea) on the UCIHAR (Anguita et al., 2013) 12 → 16 and 19 → 15 tasks. The results of the t-SNE embedding are presented in Fig. 4, spanning from subfigures (a) through (h). In subfigures (a) and (e), the visualizations reveal that the original time series data exhibit poor separability in the feature space. The season and trend factors of the time series data respectively represent different information from the original data, which can be known by the discrete degree of different color clusters in the illustration. For instance, as illustrated in subfigures (b) and (f), trend-related data effectively differentiates the blue, green, and red categories. Similarly, subfigures (c) and (g) demonstrate that season-related information is key in distinguishing between the purple and yellow categories. This suggests that our proposed method for learning disentangled representations of season and trend in the original time series data is reasonable.

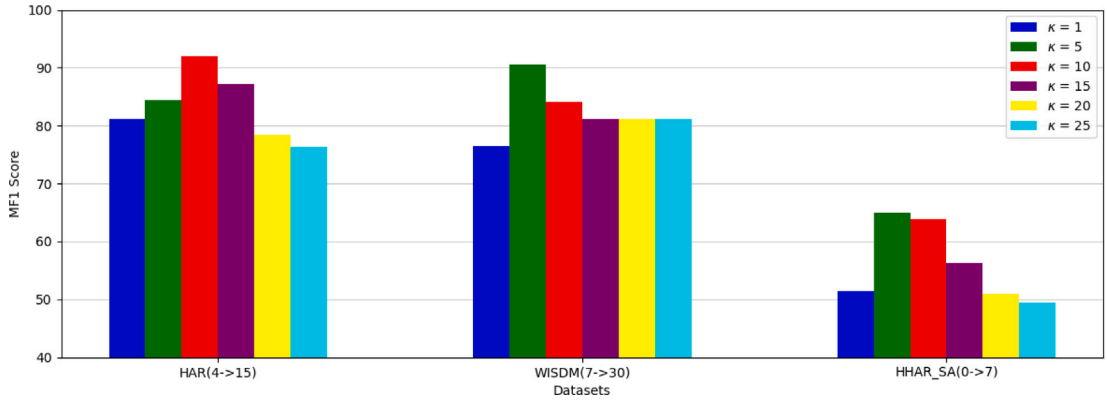

Fig. 5. Sensitivity Analysis of the kernel Sizes Parameter κ. Bars of different colors represent distinct κ values: blue ($\kappa = 1$), green ($\kappa = 5$), red ($\kappa = 10$), purple ($\kappa = 15$), yellow ($\kappa = 20$), and cyan ($\kappa = 25$). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.2.3. Sensitivity analysis of the kernel sizes parameter κ

In this subsection, our objective was to investigate the influence of kernel sizes (κ) on MDLR model's performance across diverse transfer tasks within datasets, specifically tasks randomly selected as HAR(4→15), WISDM(7→30), and HHAR_SA(0→7). As visualized in Fig. 5, the x-axis represents categorizes of the datasets, and the y-axis quantifies the performance using the MF1 score. From the chart, HAR(4→15) achieves its highest MF1 score with $\kappa = 10$, WISDM(7→30) peaks at $\kappa = 5$, and HHAR_SA(0-> 7) reaches its optimal performance at $\kappa = 5$. Overall, in all transfer tasks, there is a common pattern where the MF1 score initially increases as κ increases and then starts to weaken. The optimal κ value appears to be around 8 for these tasks. Importantly, it is clear that too high values of κ may result in a decrease in performance. This underscores the importance of carefully tuning the κ parameter for each dataset when using the MDLR model.

6. Discussion and analysis

6.1. Results analysis

In response to the challenge of adapting deep learning models to diverse time series domains, MDLR is designed as a dual-tower architecture capable of disentangled learning of trend-related and season-related features to better extract invariant factors inherent in various domains of time series data. Our experimental results reveal that our newly developed MDLR approach is highly effective in addressing the UTSDA challenge. Specifically, when compared to existing methods such as LSTM, Deep CORAL, CoDATS, SASA, AdvSKM, and CoTMix on three real-world time series datasets, MDLR consistently outperforms them, as indicated by higher MF1 mean scores. Ablation studies and sensitivity analysis further emphasize the significance of individual components in the overall model architecture. The MDLR-tr and MDLR-sea variants in the ablation experiments reveal the distinct contributions of trend and season-related information to the model's performance. In this research, the Moving Average (MA) method is applied to smooth out seasonalfluctuations, with kernel size (κ) influencing data smoothness and sensitivity. Our experiments explore different kernel sizes, finding the optimal range to be 5–10. However, this varies with the dataset and transfer task, as indicated by sensitivity analysis.

Overall, combining all experimental results and comparing them with existing baseline methods highlights the advantages of our proposed MDLR method, which are primarily focused on the following points: (1) In complex time series data domation adaptation tasks, this method demonstrates good performance. (2) It emphasizes the importance of trend and season components in time series data, especially in transfer tasks, where the performance of domain-invariant factors in different domains exhibits significant differences in both the season and trend scales. The MDLR method effectively identifies and utilizes these differences to enhance the model's transfer capability. (3) By separately extracting trend-related and season-related domain-invariant information from decomposed trend and seasonalcomponents of time series data, it effectively decouples representations of trend and periodic-related invariant factors. While our MDLR method achieves good performance, its limitations mainly lie in the manual setting of the hyperparameter κ for different datasets, which not only increases the preprocessing workload but may also affect the final performance.

6.2. Theoretical and practical implications

From a theoretical perspective, we introduce the MDLR framework with a causal perspective, aiming to enhance the learning of domain-invariant mechanisms across various domains. Specifically, we analyze the Time Series Domain Adaptation (TSDA) task using causal inference and construct a causal graph based on the inherent decomposability features of trend and season components in time-series data. Using a Structural Causal Model (SCM) and do-intervention operations, we design a dual-tower architecture

comprising a Trend Feature Extractor (TFE) and a Season Feature Extractor (SFE) to extract trend-related and season-related information. The causal theory provides a solid theoretical foundation for our model, offering a valuable theoretical framework for addressing more complex domain-invariant factors in future research. In practical terms, our proposed multi-task disentangled representation learning algorithm takes into consideration the separability of trend and season components in real-world time-series data. By separately extracting features from these components to obtain domain-invariant factors, it significantly reduces the complexity of training a single model to achieve disentangled representations in practice. Therefore, this work holds great promise for practical applications.

7. Conclusion

This paper introduces a new method for unsupervised time series domain adaptation (UTSDA), focusing on a multi-task approach that leverages disentangled learning representations. Specifically, the causal inference theory provides a theoretical foundation for applying the disentangled learning representation method in UTSDA. Simultaneously, a multi-task framework is employed to extract trend-related and season-related mechanisms separately. By using a cross-domain alignment loss to facilitate cross-domain feature alignment, the method effectively learns invariant information across various domains. The proposed MDLR method has been tested and validated for its effectiveness through experiments on three different real-world datasets, demonstrating its practical utility in a variety of scenarios. Despite these promising results, our approach is not without limitations. A primary concern is that our method focuses predominantly on trend and seasonality aspects in time series data. This focus potentially overlooks other critical factors, which might be crucial in certain domain adaptation tasks. In response to this limitation, future work will explore the integration of additional mechanisms within the UTSDA framework to capture a wider array of domain-specific features.

CRediT authorship contribution statement

Yu Liu: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **Duantengchuan Li:** Conceptualization, Methodology, Software, Supervision, Writing – original draft, Writing – review & editing. **Jian Wang:** Conceptualization, Funding acquisition, Supervision, Writing – review & editing. **Bing Li:** Conceptualization, Funding acquisition, Supervision, Writing – review & editing. **Bo Hang:** Visualization, Writing – review & editing.

Data availability

Data will be made available on request.

Acknowledgments

This research is supported by the National Key Research and Development Program of China (Grant No. 2022YFF0902701), the National Natural Science Foundation of China (Grant No. 62032016), and the Key Research and Development Program of Hubei Province (No. 2021BAA031).

References

Alqahtani, Y., Al-Twairesh, N., & Alsanad, A. (2023). *Information Processing & Management*, *60*(3), Article 103338.
Anguita, D., Ghio, A., Oneto, L., Parra Perez, X., & Reyes Ortiz, J. L. (2013). *Proceedings of the 21th international european symposium on artificial neural networks, computational intelligence and machine learning* (pp. 437–442).
Cai, R., Chen, J., Li, Z., Chen, W., Zhang, K., Ye, J., et al. (2021). *Proceedings of the AAAI conference on artificial intelligence, vol. 35, no. 8* (pp. 6859–6867).
Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., & Abbeel, P. (2016). *Advances in neural information processing systems: vol. 29*.
Chen, M.-H., Kira, Z., AlRegib, G., Yoo, J., Chen, R., & Zheng, J. (2019). *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 6321–6330).
Chen, W., Wang, W., Peng, B., Wen, Q., Zhou, T., & Sun, L. (2022). *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining* (pp. 146–156).
Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. (1990). *The Journal of Official Statistics*, *6*(1), 3–73.
Dudek, G. (2023). *IEEE Transactions on Knowledge and Data Engineering*.
Eldele, E., Ragab, M., Chen, Z., Wu, M., Kwoh, C.-K., & Li, X. (2023). *IEEE Transactions on Artificial Intelligence*.
Fu, T.-c. (2011). *Engineering Applications of Artificial Intelligence*, *24*(1), 164–181.
Gretton, A., Borgwardt, K., Rasch, M., Schölkopf, B., & Smola, A. (2006). *vol. 19, Advances in neural information processing systems*.
He, H., Queen, O., Koker, T., Cuevas, C., Tsiligkaridis, T., & Zitnik, M. (2023). *International conference on machine learning*.
He, T., Xia, Z., Chen, J., Li, H., & Chan, S. H. G. (2023). Target-agnostic source-free domain adaptation for regression tasks.
Hochreiter, S., & Schmidhuber, J. (1997). *Neural Computation*, *9*, 1735–1780. http://dx.doi.org/10.1162/neco.1997.9.8.1735.
Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., et al. (2018). *International conference on machine learning* (pp. 1989–1998). Pmlr.
Hussein, A., & Hajj, H. (2022). *ACM Transactions on Internet of Things*, *3*(2), 1–26.
Hyndman, R. J. (2011). Moving averages.
Javidian, M. A., Pandey, O., & Jamshidi, P. (2021). Scalable causal domain adaptation. arXiv preprint arXiv:2103.00139.
Jin, X., Park, Y., Maddix, D., Wang, H., & Wang, Y. (2022). *International conference on machine learning* (pp. 10280–10297). PMLR.
Kang, G., Jiang, L., Yang, Y., & Hauptmann, A. G. (2019). *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4893–4902).
Khemakhem, I., Kingma, D., Monti, R., & Hyvarinen, A. (2020). *International conference on artificial intelligence and statistics* (pp. 2207–2217). PMLR.
Kisamori, K., Kanagawa, M., & Yamazaki, K. (2020). *International conference on artificial intelligence and statistics* (pp. 1244–1253). PMLR.
Kwapisz, J. R., Weiss, G. M., & Moore, S. A. (2011). *ACM SigKDD Explorations Newsletter*, *12*(2), 74–82.

Li, Z., Cai, R., Chen, J., Yan, Y., Chen, W., Zhang, K., et al. (2022). Time-series domain adaptation via sparse associative structure alignment: Learning invariance and variance. arXiv preprint arXiv:2205.03554.
Li, Y., Chen, Z., Zha, D., Du, M., Zhang, D., Chen, H., et al. (2021). Learning disentangled representations for time series. arXiv preprint arXiv:2105.08179.
Li, Z., Zhang, Q., Zhu, F., Li, D., Zheng, C., & Zhang, Y. (2023). *Information Processing & Management*, *60*(4), Article 103348.
Liu, Z., Lin, Y., & Sun, M. (2023). *Representation learning for natural language processing*. Springer Nature.
Liu, Y., Wang, J., & Li, B. (2024). *Information Sciences*, *652*, Article 119578.
Liu, Q., & Xue, H. (2021). *IJCAI* (pp. 2744–2750).
Lv, F., Liang, J., Li, S., Zang, B., Liu, C. H., Wang, Z., et al. (2022). *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 8046–8056).
Meng, Q., Qian, H., Liu, Y., Xu, Y., Shen, Z., & Cui, L. (2023). Unsupervised representation learning for time series: A review. arXiv preprint arXiv:2308.01578.
Nozza, D., Manchanda, P., Fersini, E., Palmonari, M., & Messina, E. (2021). *Information Processing & Management*, *58*(3), Article 102537.
Oza, P., Sindagi, V. A., Sharmini, V. V., & Patel, V. M. (2023). *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
Ozyurt, Y., Feuerriegel, S., & Zhang, C. (2023). *ICLR*.
Peters, J., Janzing, D., & Schölkopf, B. (2017). *Elements of causal inference: foundations and learning algorithms*. The MIT Press.
Purushotham, S., Carvalho, W., Nilanon, T., & Liu, Y. (2016). *International conference on learning representations*.
Qiu, J., Jammalamadaka, S. R., & Ning, N. (2018). *Journal of Machine Learning Research*, *19*(1), 2744–2776.
Ragab, M., Eldele, E., Tan, W. L., Foo, C.-S., Chen, Z., Wu, M., et al. (2023). *ACM Transactions on Knowledge Discovery from Data*, http://dx.doi.org/10.1145/3587937.
Ragab, M., Eldele, E., Wu, M., Foo, C.-S., Li, X., & Chen, Z. (2023). *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining* (pp. 1989–1998).
Ramponi, A., & Plank, B. (2020). In D. Scott, N. Bel, & C. Zong (Eds.), *Proceedings of the 28th international conference on computational linguistics* (pp. 6838–6855). Barcelona, Spain (Online): International Committee on Computational Linguistics, http://dx.doi.org/10.18653/v1/2020.coling-main.603, Retrieved from https://aclanthology.org/2020.coling-main.603.
Rozantsev, A., Salzmann, M., & Fua, P. (2018). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *41*(4), 801–814.
Schölkopf, B. (2022). *Probabilistic and causal inference: the works of Judea Pearl* (pp. 765–804).
Schölkopf, B., Janzing, D., Peters, J., Sgouritsa, E., Zhang, K., & Mooij, J. (2012). On causal and anticausal learning. arXiv preprint arXiv:1206.6471.
Scott, S. L., & Varian, H. R. (2015). *Economic analysis of the digital economy* (pp. 119–135). University of Chicago Press.
Stisen, A., Blunck, H., Bhattacharya, S., Prentow, T. S., Kjærgaard, M. B., Dey, A., et al. (2015). *Proceedings of the 13th ACM conference on embedded networked sensor systems* (pp. 127–140).
Sun, B., Feng, J., & Saenko, K. (2016). *Proceedings of the AAAI conference on artificial intelligence, vol. 30, no. 1*.
Sun, B., & Saenko, K. (2016). *European conference on computer vision* (pp. 443–450). Springer.
Trung, N. N., Van, L. N., & Nguyen, T. H. (2022). *Proceedings of the 29th international conference on computational linguistics* (pp. 4741–4752).
Tzeng, E., Hoffman, J., Darrell, T., & Saenko, K. (2015). *Proceedings of the IEEE international conference on computer vision* (pp. 4068–4076).
Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., & Darrell, T. (2014). Deep domain confusion: Maximizing for domain invariance. arXiv preprint arXiv:1412.3474.
Wang, X., Chen, H., Zhou, Y., Ma, J., & Zhu, W. (2022). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(1), 408–424.
Wang, W., Feng, F., He, X., Zhang, H., & Chua, T.-S. (2021). *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval* (pp. 1288–1297).
Wang, Y., Xu, Y., Yang, J., Chen, Z., Wu, M., Li, X., et al. (2023). *Proceedings of the AAAI conference on artificial intelligence, vol. 37, no. 8* (pp. 10253–10261).
Wen, Q., Zhou, T., Zhang, C., Chen, W., Ma, Z., Yan, J., et al. (2022). Transformers in time series: A survey. arXiv preprint arXiv:2202.07125.
Wilson, G., Doppa, J. R., & Cook, D. J. (2020). *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 1768–1778).
Woo, G., Liu, C., Sahoo, D., Kumar, A., & Hoi, S. (2022). *International conference on learning representations*. Retrieved from https://openreview.net/forum?id=PilZY3omXV2.
Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., et al. (2015). *International conference on machine learning* (pp. 2048–2057). PMLR.
Yang, J., Ding, X., Zheng, Z., Xu, X., & Li, X. (2023). *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 11878–11887).
Yang, M., Liu, F., Chen, Z., Shen, X., Hao, J., & Wang, J. (2021). *2021 IEEE/CVF conference on computer vision and pattern recognition* (pp. 9588–9597). http://dx.doi.org/10.1109/CVPR46437.2021.00947.
Zhang, Y., Sang, J., Wang, J., Jiang, D., & Wang, Y. (2023). *Proceedings of the 31st ACM international conference on multimedia* (pp. 8860–8868). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/3581783.3612317.