IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 12 May 2025, accepted 3 July 2025, date of publication 14 July 2025, date of current version 23 July 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3588009

RESEARCH ARTICLE

TREET: TRansfer Entropy Estimation via Transformers

OMER LUXEMBOURG, (Student Member, IEEE), DOR TSUR, (Student Member, IEEE), AND HAIM PERMUTER, (Senior Member, IEEE)

Electrical and Computer Engineering Department, Ben-Gurion University of the Negev, Be'er-Sheva 8410501, Israel

Corresponding author: Omer Luxembourg (omerlux@post.bgu.ac.il)

This work was supported in part by Israel Science Foundation (ISF) under Grant 899/21 and Grant 3211/23, and in part by the NSF-(Israel-USA) Binational Science Foundation (BSF) Grant.

ABSTRACT Transfer entropy (TE) is an information theoretic measure that reveals the directional flow of information between processes, providing valuable insights for a wide range of real-world applications. This work proposes Transfer Entropy Estimation via Transformers (TREET), a novel attention-based approach for estimating TE for stationary processes. The proposed approach employs Donsker-Varadhan representation to TE and leverages the attention mechanism for the task of neural estimation. We propose a detailed theoretical and empirical study of the TREET, comparing it to existing methods on a dedicated estimation benchmark. To increase its applicability, we design an estimated TE optimization scheme that is motivated by the functional representation lemma, and use it to estimate the capacity of communication channels with memory, which is a canonical optimization problem in information theory. We further demonstrate how an optimized TREET can be used to estimate underlying densities, providing experimental results. Finally, we apply TREET to feature analysis of patients with Apnea, demonstrating its applicability to real-world physiological data. Our work, applied with state-of-the-art deep learning methods, opens a new door for communication problems which are yet to be solved.

INDEX TERMS Deep learning, information theory, transfer entropy, transformers, neural estimation, communication channels.

I. INTRODUCTION

Transfer entropy (TE), introduced by Schreiber [1], stands as a pivotal information-theoretic measurement that captures the coupling dynamics within temporally evolving systems [2]. Derived as an extension of mutual information (MI), TE is distinctive for its inherent asymmetry, strategically employed in diverse applications for causal analysis [3]. TE serves as a robust measure of the directed, asymmetric information flow between two stochastic processes. Specifically, TE quantifies the reduction in uncertainty about a process observations, incorporating the past values of another process to predict the future values of the first [4].

TE has found applications in various domains. In neuroscience, it has proven to be effective in deciphering functional connectivity among and between neurons to various physical tasks [5], [6], [7]. Moreover, analysis between visual sensors and movement actuators in embodied cognitive systems can be one via TE [8]. TE is instrumental in social network analysis, as it quantifies causal relationships between users by measuring the directed flow of information. This enables the detection of influential individuals and the examination of how misinformation propagates through the network [9]. [10] utilizes a variant of TE to predict the direction of the US stock market, incorporating TE as input feature. Lately, [11] suggested a greedy algorithm for feature selection while leveraging the connection between each feature and the target with TE. Estimating TE is crucial for distinguishing between correlation and causation, as it quantifies the direction and magnitude of information flow between variables, thereby uncovering the underlying causal structures in complex systems

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

© 2025 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License.
For more information, see https://creativecommons.org/licenses/by/4.0/

A. ESTIMATION AND OPTIMIZATION OF TRANSFER ENTROPY

Estimating information theoretic measures such as TE is challenging, especially when the underlying data distribution is unkown [4]. Many methods drows inspiration from MI estimators. Among the estimation methods, Kernel Density Estimation (KDE) [12], and k-nearest neighbors (KNN) based methods such as Kraskov-Stögbauer-Grassberger (KSG) [13], [14] are noteworthy, despite their struggle with the bias-variance trade-off. Granger causality [15], a statistical concept introduced by Clive W. J. Granger, assesses whether one time series can predict another. Specifically, if the inclusion of past values of a time series X enhances the prediction of another time series Y beyond the information provided by past values of Y alone, then X is said to ''Granger-cause'' Y. This is typically evaluated using vector autoregressive (VAR) models, where the predictive power of past values of X on Y is tested for statistical significance. Notably, Granger causality [15] also serves as a foundational method for TE estimation, particularly in Gaussian joint processes where it is equivalent to TE, highlighting its effectiveness in quantifying information flow between sequences [16], [17].

Copnet [18], a copula entropy-based method, builds on empirical density estimation similar to KNN-based approaches and successfully quantifies TE. By establishing the relationship between copula entropy and MI, [18] proposed a formulation of TE using four copula entropy terms.

Recent advancements have embraced neural estimation approaches like the Donsker-Varadhan (DV) and Nguyen-Nowozin-Jordan variational formulas for Kullback-Leibler (KL) and f-divergence to accurately estimate MI, directed information (DI) rate, and TE [19], [20], [21], [22]. These methods include MI neural estimation (MINE) for MI, DI neural estimation (DINE) for DI rate, and TE neural estimation (TENE) for TE, solving the optimization problems posed by variational formulas using NNs.

TENE [22] uses the classifier-based conditional MI estimator [23], which estimates two MI terms that are subtracted to obtain TE. Besides using NNs for optimization instead of a non-parametric method, TENE reduce the number of terms in the TE formula to two MI values, unlike Copnet. However, it suffers from input dimensionality issues when estimating higher-order TE, since increasing the sequence length to estimate larger-order TE introduces input dimensionality challenges in the DV-based variational formulation. In contrast, [21] estimate DI rate, a closely related measure to TE, using RNNs, enabling estimation for theoretically infinite memory sequences. Nevertheless, DINE is designed to estimate information flow over infinite sequences, which corresponds to the DI rate and is not well-suited for finite-order TE estimation.

NNs have revolutionized fields such as computer vision and natural language processing, with transformers now dominating time-series analysis, overtaking RNNs due to their superior performance [24], [25], [26], [27]. Despite the widespread adoption of MINE for information-theoretic estimation, it has been subject to scrutiny due to its limitations, such as high variance in gradient estimates, bias in MI estimation, and training instability under high-dimensional settings [28], [29], [30].

To address these challenges, we introduce TREET, a transformer-based neural estimator of transfer entropy for high-dimensional continuous data.

B. CONTRIBUTIONS

In this work, we introduce TREET, a novel TE estimator that leverages the attnetion mechanism. Our method is grounded in the DV representation, leveraging attention-based NNs adapted to meet the structural constraints of DV optimization. Theoretically, we establish the consistency of TREET and develop an auxiliary neural distribution generator (NDG), an input distribution generative module to facilitate TE optimization, utilizing transformers. Empirically, we showcase the versatility of TREET across various applications.

- Show that TREET outperforms TENE and Copnet by extending the benchmark presented in [22] to address longer order TE, particulary in scenarios of high valued TE and long temporal contexts.
- Demonstrate the joint optimization of TREET and the NDG for the estimation of the capacities of channels with memory, supported by theoretical validations. Experiments on long-memory channels emphasize TREET's ability to capture and utilize extensive historical dependencies while accurate estimating their channel capacities.
- Derive a density estimator of the underlying process, which emerges as a byproduct of the TREET optimization.
- Apply TREET to the Apnea dataset [31], [32] for feature analysis, illustrating its utility in uncovering causal relationships within real-world data, paving the way for broader applications in future research.

C. ORGANIZATIONS

The reminder of the paper is organized as follows, Section II provides background on information theory and NNs. Section III presents the TREET, its theoretical guarantees and practical implementations, whereas the optimization of TREET is described in Section IV. Experimental results on TE dedicated benchmark, channel capacity estimation and it's memory analysis, probability density estimation and features analysis on real-world data are shown in Section V. Section VI concludes the paper and discuss future research potentials.

II. BACKGROUND

In this section, we elaborate on the preliminaries necessary to present our method. We familiarize the reader with our notation and provide the formal definition of TE, then relate it to DI. Subsequently, we introduce the concept of transformer

NN, define them, and discuss the theorem of universal approximation. Lastly, we present the use of NN as estimators for information theoretic measures.

A. NOTATION

Calligraphic letters, such as $\mathcal{X}$, denote subsets of the d-dimensional Euclidean space, $\mathbb{R}^d$. Expectations are represented by $\mathbb{E}$, with all random variables defined in the probability space $(\Omega, \mathcal{F}, \mathbb{P})$. The collection of Borel probability measures on $\mathcal{X}$ is indicated as $\mathcal{P}(\mathcal{X})$, and $\mathcal{P}_{\mathsf{ac}}(\mathcal{X})$ specifically refers to those measures that are absolutely continuous with respect to Lebesgue measure, with their densities denoted by lowercase p. Random variables and vectors are uppercase, e.g., X, and stochastic processes are in blackboard bold, e.g., $\mathbb{X} := (X_t)_{t\in\mathbb{Z}}$ for discrete time t.

The sequence of l samples from time t in process $\mathbb{X}$ is $X_t^{t+l} := [X_t, \ldots, X_{t+l}]^\top$, and for stationary processes, $X^l := [X_0, X_1, \ldots, X_l]^\top$. For $\mathcal{Q} \in \mathcal{P}_{\mathsf{ac}}(\mathcal{X})$ with PDF q, the cross entropy between P and Q is $\mathsf{h}_{\mathsf{CE}}(P, Q) := -\mathbb{E}_P[\log q]$. Differential entropy of $X \sim P$ is $\mathsf{h}(X) := \mathsf{h}_{\mathsf{CE}}(P, P)$. If $Q \ll P$, the KL divergence $\mathsf{D}_{\mathsf{KL}}(P\|Q) := \mathbb{E}_P[\log \frac{\mathrm{d}P}{\mathrm{d}Q}]$. MI between $(X, Y) \sim P_{XY}$ is $\mathsf{I}(X; Y) := \mathsf{D}_{\mathsf{KL}}(P_{XY}\|P_X \otimes P_Y)$. Conditional KL divergence for $P_{Y|X}, Q_{Y|X}$ given $X \sim P_X$ is $\mathsf{D}_{\mathsf{KL}}(P_{Y|X}\|Q_{Y|X}|P_X)$, and conditional MI for $(X, Y, Z) \sim P_{XYZ}$ is $\mathsf{I}(X; Y|Z) := \mathsf{D}_{\mathsf{KL}}(P_{XY|Z}\|P_{X|Z} \otimes P_{Y|Z}|P_Z)$.

For a comprehensive list of all symbols and their definitions, please refer to Appendix E.

B. TRANSFER ENTROPY

TE quantifies causal influence from the past of one sequence on the present of another, formally given as follows

Definition 1 (Transfer Entropy): For jointly distributed processes $\mathbb{X}$ and $\mathbb{Y}$ and $k, l < \infty$, the TE, with parameters (k, l), is given by

$$\mathsf{TE}_{X\to Y}(t; k, l) := \mathsf{I}\left(X_{t-l}^t; Y_t|Y_{t-k}^{t-1}\right). \tag{1}$$

When the considered processes are jointly stationary, the temporal index t in (1) can be omitted, and TE is written as

$$\mathsf{TE}_{X\to Y}(k, l) := \mathsf{I}\left(X^l; Y_l|Y_{l-k}^{l-1}\right). \tag{2}$$

Note that our definition of TE includes X_t, whereas other definitions [1], [16], [22], [33] define TE as $\mathsf{I}(X^{l-1}; Y_l|Y_{l-k}^{l-1})$. The main difference introduced by this change is that the definition reduces to MI when the processes are jointly independent and identically distributed (i.i.d.), i.e., $\mathsf{TE}_{X\to Y}(0, 0) = I(X; Y)$. When $k = l$, the abbreviated notation $\mathsf{TE}_{X\to Y}(l)$ is used.

C. RELATION TO DIRECTED INFORMATION

DI [34], [35] is given by

$$\mathsf{I}\left(X^n \to Y^n\right) := \sum_{i=0}^{n-1} \mathsf{I}\left(X^i; Y_i|Y^{i-1}\right) \tag{3}$$

$$\mathsf{I}\left(X^n \to Y^n\right) = \sum_{i=0}^{n-1} \mathsf{TE}_{X\to Y}(i). \tag{4}$$

When (X^n, Y^n) are the n-fold projections of stochastic processes $\mathbb{X}$ and $\mathbb{Y}$, which (2) shows that DI is the sum of TE terms. The DI *rate* can be defined as

$$\begin{aligned}\mathsf{I}(\mathbb{X} \to \mathbb{Y}) &:= \lim_{n\to\infty} \frac{1}{n} \mathsf{I}\left(X^n \to Y^n\right) \\ &\overset{(a)}{=} \lim_{n\to\infty} \mathsf{I}(X^n; Y_n|Y^{n-1}),\end{aligned} \tag{5}$$

and (a) holds due to stationarity [21]. The DI rate can be therefore observed as a limit case of TE, where the limit of TE exists since $\mathsf{TE}_{X\to Y}(l) = \mathsf{h}(Y_l|Y^{l-1}) - \mathsf{h}(Y_l|X^l, Y^{l-1})$ and each conditional entropy is non-negative, decreasing for increasing l (conditioning reduces entropy). Moreover, under appropriate Markov assumptions, $Y_l - Y^{l-1} - Y_{-\infty}^{-1}, Y_l - (X^l, Y^{l-1}) - (X_{-\infty}^{-1}, Y_{-\infty}^{-1})$, an equality between TE and DI can be established,

Lemma 1: Define markov property $Z_n - Z_{n-1} - Z^{n-2}$ as $P_{Z_n|Z^{n-1}} = P_{Z_n|Z_{n-1}}$. Let $\mathbb{X}$ and $\mathbb{Y}$ be two jointly stationary processes such that the following markov property holds,

$$\begin{aligned}&Y_l - Y^{l-1} - Y_{-\infty}^{-1}, \\ &Y_l - (X^l, Y^{l-1}) - (X_{-\infty}^{-1}, Y_{-\infty}^{-1}),\end{aligned}$$

for $l \in \mathbb{N}$. Then, for $m \in \mathbb{N}, m \geq l$,

$$\mathsf{TE}_{X\to Y}(m) = \mathsf{TE}_{X\to Y}(l), \tag{6}$$

and

$$\mathsf{TE}_{X\to Y}(m) = \mathsf{I}\left(\mathbb{X} \to \mathbb{Y}\right). \tag{7}$$

Refer to Appendix A for detailed proof.

D. ATTENTION MECHANISM AND TRANSFORMERS

NNs capabilities can be enhanced by the attention[1] mechanism [24], which selects various combinations of inputs according to their significance to the predictions. The attention can address time series datasets, while weighting over each time input. Attention comprises *queries*, which represent the current temporal focus, *keys*, which match against the queries to determine relevance, and *values*, that contain the NN inputs to be weighted and act as a memory function in time-series.

Definition 2 (Attention): Let $W_Q, W_K, W_V \in \mathbb{R}^{x\times d}$ and let $Q = XW_Q, K = XW_K$ and $V = XW_V$ be the queries, keys and values, $X = (x_1, x_2, \ldots, x_n), \forall i : x_i \in \mathbb{R}^{d_x}$. Then the attention is given by,

$$\mathsf{Attn}(X) = \mathsf{softmax}\left(QK^\top\right)V,$$

where $\mathsf{softmax}(Z)_j = \exp[Z_j]/\sum_{m=1}^n \exp[Z_m]$ where $Z \in \mathbb{R}^n$, is performed for each column of the dot-product between queries and keys separately.

Transformers utilize positional encoding (PE), which maps each input to the attention according to its ordinal index, and is applied before the attention for time series applications to deformation of the sequence structure. Attention can be

[1]This paper considers the dot-product attention.

generalized by considering several heads which operate in parallel. Multi-head attention is given by:

$$\text{MultiHead-Attn}(X) = [H_1, \ldots, H_h]W_0, \tag{8}$$

where

$$H_i = \text{softmax}\left(Q^{(i)}K^{(i)\top}\right)V^{(i)}, \quad i = 1, \ldots, n, \tag{9}$$

$W_0 \in \mathbb{R}^{dh\times d}$ is learnable parameter, and $Q^{(i)}, K^{(i)}, V^{(i)}$ are the i^{th} learnable projections of queries keys and values [24]. A transformer block is a sequence-to-sequence function, which maps input sequence to an output sequence. Each transformer block consists of attention layer and time-wise feed-forward layer. Formally, a transformer function class is given as follows [36]

Definition 3 (Transformer Function Class): Let $d_i, d_o, l, v \in \mathbb{N}$. The class of transformers with v neurons, denoted $\mathcal{G}_{\text{tf}}^{(d_i,d_o,l,v)} : \mathbb{R}^{d_i\times l} \to \mathbb{R}^{d_o\times l}$, is the set of discrete-time with the following structure:

$$X_{\text{pe}} = W_1 \cdot X + E, \tag{10a}$$

$$\text{Attn}(X_{\text{pe}}) = X_{\text{pe}} + \sum_{i=1}^{h} W_O^i W_V^i X_{\text{pe}} \cdot \text{softmax}\left[(W_K^i X_{\text{pe}})^\top (W_Q^i X_{\text{pe}})\right], \tag{10b}$$

$$Y = \text{FF}(X_{\text{pe}}) = \text{Attn}(X_{\text{pe}}) + W_3 \cdot \sigma_{\text{R}}\left(W_2 \cdot \text{Attn}(X_{\text{pe}}) + b_1\mathbf{1}_l^\top\right) + b_2\mathbf{1}_l^\top, \tag{10c}$$

where $X \in \mathbb{R}^{d_i\times l}$ is the input sequence of l samples, $Y \in \mathbb{R}^{d_o\times l}$ is the transformer output, $W_1 \in \mathbb{R}^{d_e\times d_i}$, $W_O^i \in \mathbb{R}^{d_e\times d_m}$, $W_Q^i, W_K^i, W_V^i \in \mathbb{R}^{d_m\times d_e}$, $W_2 \in \mathbb{R}^{d_r\times d_e}$, $W_3 \in \mathbb{R}^{d_e\times d_r}$, $b_1 \in \mathbb{R}^{d_r}$, $b_2 \in \mathbb{R}^{d_e}$ are the weights and biases of the network, $E \in \mathbb{R}^{d_e\times l}$ is the PE of the input. The number of heads h and the head size d_m are the parameters of the multi-head attention and d_r is the hidden dimension of the feed-forward (FF) layer. Assuming that the input is an additive product with it's positional encoding product, before the transformer blocks. The class of transformers with dimensions (d_i, d_o, l) is thus given by

$$\mathcal{G}_{\text{tf}}^{(d_i,d_o,l)} := \bigcup_{v\in\mathbb{N}} \mathcal{G}_{\text{tf}}^{(d_i,d_o,l,v)}. \tag{11}$$

Transformers are a universal approximation class of sequence to sequence mappings [36]:

Theorem 1 (Universal Approx. for Transformers): Let $\epsilon > 0$, $l \in \mathbb{N}$. $\mathcal{U} \subset \mathbb{R}^{T\times d_i}$, $\mathcal{Z} \subset \mathbb{R}^{l\times d_o}$ be open sets, and $f : \mathcal{U} \to \mathcal{Z}$ be a continuous vector-valued function. Then, there exist $v \in \mathbb{N}$ and a v-neuron transformer $g \in \mathcal{G}_{\text{tf}}^{(d_i,d_o,l,v)}$ (as in Definition 3, such that for any sequence of inputs $\{u^l\} \in \mathcal{U}$ and sequence of outputs $\{z^l\} \in \mathcal{Z}$, we have

$$\left\|f(u^l) - g(u^l)\right\|_1 \leq \epsilon, \tag{12}$$

Theorem 1 establishes the universal approximation capabilities of transformers for sequence-to-sequence mappings, whereas many real-world applications, such as time-series forecasting and information flow estimation, require models that respect the inherent causal structure of sequential data. To address this, the class of *causal transformers* is utilized, $\mathcal{G}_{\text{ctf}}$, which is built upon $\mathcal{G}_{\text{tf}}$ and enforces a strict temporal dependency, ensuring that each output at certain time step is computed only from past and present inputs. This causal structure is crucial for tasks such as TE estimation, where information flow must be inferred without accessing future observations. To this end, the notation of causal functions is used

Definition 4 (Causal Function): Let $\mathcal{F} : \mathbb{R}^{d_u\times L} \to \mathbb{R}^{d_z\times L}$ be a function for $d_u, d_z, L \in \mathbb{N}$. For a series of inputs $U = \{u_t\}_{t=1}^L$ and function outputs $Z = \{z_t\}_{t=1}^L$, the function $\mathcal{F}$ is as a causal function if, for any t_0 the output z_{t_0} depends only on the inputs $\{u_t\}_{t\leq t_0}$.

To enforce causality in the mapping function, a *causal mask* is applied to the attention scores [24], ensuring that each output only depends on past and present inputs, as defined in Definition 4. Notably, only the attention mechanism performs time mixing, so applying the causal mask does not alter the rest of the transformer architecture. The causal mask, denoted as $M \in \mathbb{R}^{l\times l}$, is applied element-wise to the dot-product of keys and queries before the softmax operation and is given by

$$M_{[i,j]} = \begin{cases} 1 & \text{if} j \leq i \\ -\infty & \text{otherwise} \end{cases} \tag{13}$$

where $-\infty$ nullifies the corresponding entries after the softmax operation. Hence, (10b) is written as,

$$\text{Attn}(X_{\text{pe}}) = X_{\text{pe}} + \sum_{i=1}^{h} W_O^i W_V^i X_{\text{pe}} \cdot \text{softmax}\left[(W_K^i X_{\text{pe}})^\top (W_Q^i X_{\text{pe}}) \odot M\right], \tag{14}$$

where $\odot$ is the Hadamard (element-wise) product, meaning that the multiplication is performed entry-wise between matrices of the same dimensions. Although changing the dot-product operation of the attention, the model remains consistent with the universality framework, which is grounded in the architecture's capacity for pair-wise operations rather than being constrained by the specifics of the attention mechanism [36].

E. NEURAL ESTIMATION

Neural estimation leverages NNs to approximate and optimize statistical functionals, such as mutual information and statistical divergences. Neural estimators often utilize a variational formula, such as the DV representation [19, Theorem 3.2].

Theorem 2 (DV Representation): For any, $P, Q \in \mathcal{P}(\mathcal{X})$, we have

$$\text{D}_{\text{KL}}(P\|Q) = \sup_{f:\mathcal{X}\to\mathbb{R}} \mathbb{E}_P[f] - \log\left(\mathbb{E}_Q[e^f]\right), \tag{15}$$

where the supremum is taken over all measurable functions f with finite expectations.

To obtain an estimate from the DV representation, the class of functions is approximated with the class of NNs and

expectations are replaced with sample means [20], [21], [22], [37]. A provably consistent neural estimator of TE is developed, that utilizes the power of the attention mechanism.

III. ESTIMATION OF TRANSFER ENTROPY

Transformers excel at capturing long-range dependencies in time series, offering superior scalability and training stability compared to RNNs [24]. While DINE [21], built on RNNs, helped mitigate the high variance and bias of neural information-theoretic estimators, self-attention's global receptive field and parallelism allow it to further improve neural estimation in high-dimensional settings.

In this paper we harness to computational power of transformers architecture and modify the attention mechanism to result with TREET, a new estimator of TE for high dimensional continuous data. We begin by deriving the estimator, then account for its theoretical guarantees. Finally, implementation details are provided, outlining the modifications of attention to neural estimation.

A. ESTIMATOR DERIVATION

The TREET provides an estimate of the TE (1) from a set of samples $D_{n,l} = (X^{n+l}, Y^{n+l}) \sim P_{X^{n+l},Y^{n+l}}$, where l is the memory order of the TE. Assuming that $l \ll n$ and thus omitting the dependence on l in the dataset notation, i.e. $D_n := D_{n,l}$. To derive TREET, first we represent TE as subtraction of KL divergences, w.r.t. some absolutely continuous reference distribution $\widetilde{P}_Y$ over the alphabet $\mathcal{Y}$.[2] We propose the following.

Lemma 2 (TE as KL Divergences): TE decomposes as

$$\mathsf{TE}_{X\to Y}(l) = \mathsf{D}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l} - \mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}, \tag{16}$$

where

$$\mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l} := \mathsf{D}_{\mathsf{KL}}\left(P_{Y_l|Y^{l-1}}\|\widetilde{P}_Y\middle|P_{Y^{l-1}}\right), \tag{17a}$$

$$\mathsf{D}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l} := \mathsf{D}_{\mathsf{KL}}\left(P_{Y_l|Y^{l-1}X^l}\|\widetilde{P}_Y\middle|P_{Y^{l-1}X^l}\right), \tag{17b}$$

and the conditional KL divergence is $\mathsf{D}_{\mathsf{KL}}(P_{X|Z}\|P_{Y|Z}|P_Z) := \mathbb{E}_Z[\mathsf{D}_{\mathsf{KL}}(P_{X|Z}\|P_{Y|Z})]$.

Lemma 2 is proved in Section VI-C, and follows basic information theoretic properties of TE and KL divergences. Let $\mathcal{G}^Y_{\mathsf{ctf}} := \mathcal{G}^{(d_y,1,l,v_y)}_{\mathsf{ctf}}, \mathcal{G}^{XY}_{\mathsf{ctf}} := \mathcal{G}^{(d_x+d_y,1,l,v_{xy})}_{\mathsf{ctf}}$ be sets of causal transformer architectures for $l, d_y, d_x, v_y, v_{xy} \in \mathbb{N}$. Each KL term can be approximated using the DV representation (15) as follows:

$$\mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l} = \sup_{g_y\in\mathcal{G}^Y_{\mathsf{ctf}}} \mathbb{E}\left[g_y\left(Y^l\right)\right] - \log\left(\mathbb{E}\left[e^{g_y(\widetilde{Y}_l,Y^{l-1})}\right]\right), \tag{18a}$$

$$\mathsf{D}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l} = \sup_{g_{xy}\in\mathcal{G}^{XY}_{\mathsf{ctf}}} \mathbb{E}\left[g_{xy}\left(Y^l,X^l\right)\right] - \log\left(\mathbb{E}\left[e^{g_{xy}(\widetilde{Y}_l,Y^{l-1},X^l)}\right]\right). \tag{18b}$$

[2]If $\mathcal{Y}$ is not bounded, the maximal bounds can be set, regarding to the dataset properties.

where $g_y \in \mathcal{G}^Y_{\mathsf{ctf}}, g_{xy} \in \mathcal{G}^{XY}_{\mathsf{ctf}}$. Finally, replacing expectations with sample means in (18a), yields the TREET, given by

$$\widehat{\mathsf{TE}}_{X\to Y}(D_n; l) = \sup_{g_{xy}\in\mathcal{G}^{XY}_{\mathsf{ctf}}} \widehat{\mathsf{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}(D_n, g_{xy}) - \sup_{g_y\in\mathcal{G}^Y_{\mathsf{ctf}}} \widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(D_n, g_y), \tag{19a}$$

where

$$\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(D_n, g_y) := \frac{1}{n}\sum_{i=1}^n g_y\left(Y_i^{i+l}\right) - \log\left(\frac{1}{n}\sum_{i=1}^n e^{g_y\left(\widetilde{Y}_{i+l},Y_i^{i+l-1}\right)}\right), \tag{20a}$$

$$\widehat{\mathsf{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}(D_n, g_{xy}) := \frac{1}{n}\sum_{i=1}^n g_{xy}\left(Y_i^{i+l},X_i^{i+l}\right) - \log\left(\frac{1}{n}\sum_{i=1}^n e^{g_{xy}\left(\widetilde{Y}_{i+l},Y_i^{i+l-1},X_i^{i+l}\right)}\right). \tag{20b}$$

where $\widetilde{Y}$ is i.i.d. under absolutely continuous reference measure under the alphabet $\mathcal{Y}$, and

$$\sup_{g_{xy}\in\mathcal{G}^{XY}_{\mathsf{ctf}}} \widehat{\mathsf{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}(D_n, g_{xy}) = \mathsf{D}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l},$$
$$\sup_{g_y\in\mathcal{G}^Y_{\mathsf{ctf}}} \widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(D_n, g_y) = \mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}.$$

The TREET (19) is consequently given as the subtraction of the solutions of two optimization problems (20a), (20b), each optimizing its own NN. The optimization is performed via mini-batch gradient ascent with the corresponding model g_y, g_{xy} and the dataset D_n. Next, we prove that the TREET implemented with causal transformers is a consistent estimator of TE.

B. THEORETICAL GUARANTEES

The TREET consistency is established when the joint process $(\mathbb{X}, \mathbb{Y})$ is stationary and the TREET networks are implemented with the class of causal transformers. We have the following:

Theorem 3 (TREET Consistency): Let $\mathbb{X}$ and $\mathbb{Y}$ be jointly stationary, ergodic stochastic processes. The TREET is a strongly consistent estimator of $\mathsf{TE}_{X\to Y}(l)$ for $l \in \mathbb{N}$, i.e. $\mathbb{P}-a.s.$ for every $\epsilon > 0$ there exists and $N \in \mathbb{N}$ such that for every $n > N$ we have

$$\left|\widehat{\mathsf{TE}}_{X\to Y}(D_n; l) - \mathsf{TE}_{X\to Y}(l)\right| \le \epsilon \tag{21}$$

where l is the memory parameter of the TE.

The proof following the steps of 1) representation step - represents TE as a subtraction of two DV potentials, 2) estimation step - proves that the DV potentials is achievable by empirical mean of a given set of samples, and 3) approximation step - shows that the estimator built upon causal transformers converges to TE with the corresponding memory parameter. The proof is given in the Appendix B.

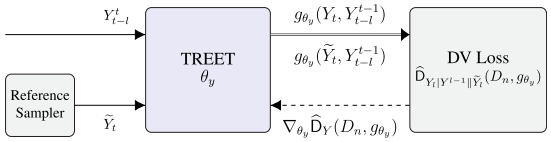


FIGURE 1. The estimator architecture for the calculation of $\widehat{\mathrm{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(D_n, g_{\theta_Y})$.

Algorithm 1 TREET

Input: Joint process samples D_n; Observation length $l \in \mathbb{N}$.
Output: $\widehat{\mathrm{TE}}_{X\to Y}(D_n; l)$ - TE estimation.

1: NNs initialization g_{θ_y}, $g_{\theta_{xy}}$ with corresponding parameters θ_y, θ_{xy}.
2: **Step 1 – Optimization:**
3: **repeat**
4: Draw a batch B_m: $m < n$ sub-sequences, length $L > l$ from D_n, with reference samples $P_{\widetilde{Y}}$ for each.
5: Compute both potentials $\widehat{\mathrm{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}(B_m, g_{\theta_{xy}})$, $\widehat{\mathrm{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(B_m, g_{\theta_y})$ via (20).
6: Update parameters:
7: $\theta_{xy} \leftarrow \theta_{xy} + \nabla_{\theta_{xy}}\widehat{\mathrm{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}(B_m, g_{\theta_{xy}})$
8: $\theta_y \leftarrow \theta_y + \nabla_{\theta_y}\widehat{\mathrm{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(B_m, g_{\theta_y})$
9: **until** convergence criteria.
10: **Step 2 – Evaluation:** Evaluate for a sub-sequence (20) and (19a) to obtain $\widehat{\mathrm{TE}}_{X\to Y}(D_n; l)$.

C. ALGORITHM AND IMPLEMENTATION

This section describes the TREET implementation. We present an overview scheme for estimation of TE and describe the algorithm. Then the TREET architecture is outlined. In practice, the TREET optimization boils down to gradient-based optimization of its parameters. Therefore, denote the TREET networks with g_{θ_y}, $g_{\theta_{xy}}$ with corresponding parameters θ_y and θ_{xy} respectively. For simplicity, we address mainly g_{θ_y} and $\mathrm{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}$, and afterwards explain how to extend all to $g_{\theta_{xy}}$ and $\mathrm{D}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}$.

1) OVERVIEW AND ALGORITHM

The TREET algorithm follows an iterative joint optimization of $\widehat{\mathrm{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}$ and $\widehat{\mathrm{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}$ through iterative mini-batch gradient optimization. The algorithm inputs are l, D_n, which are the TE parameter and the dataset, respectively. Every iteration begins with feeding mini-batch sized $m < n$ with sequences length l in each model, followed by the calculation of both DV potentials (20), that construct $\widehat{\mathrm{TE}}_{X\to Y}(l)$ (19). The calculated objective is then used for gradient-based optimization of the NN parameters. The iterative process continues until a stopping criteria is met, typically defined as the convergence of $\widehat{\mathrm{TE}}_{X\to Y}(D_n; l)$ within a specified tolerance parameter $\epsilon > 0$. For evaluation, the sequence-wise TE is estimated over as many samples and then averaged to obtain the estimated TE. The full pipeline for estimating $\widehat{\mathrm{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(D_n, g_{\theta_y})$ is presented in Fig. 1, and the complete list of steps is given in Algorithm 1.

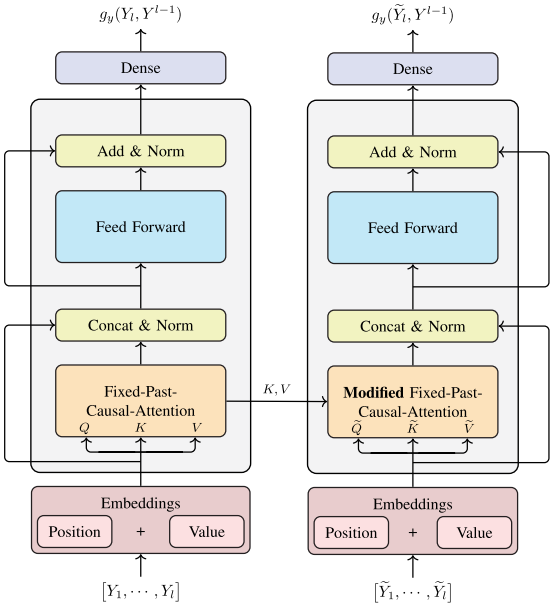


FIGURE 2. The TREET architecture for g_y, with memory parameter l. It illustrated using a single sequence as an example. However, it is capable of parallel processing for sequences of length $L > l$, and in such cases, the number of outputs for the function will be $L - l + 1$. Both transformer share the same weights. The FPCA and the modified FPCA are as elaborated in Section III-C2.

The DV representation, (15), suggest that the function f is the same function for both terms that construct it, i.e. the weights are shared for both network propagation. Hence, our transformer in TREET, which constructed by (20), is the same for both terms, for both DV representation. Exemplifying with $\widehat{\mathrm{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}$, both $g_y(Y^l)$ and $g_y(\widetilde{Y}_l, Y^{l-1})$ use the same learning parameters, from positional encoding layers and attention to FF layers. The only difference between the two terms, is in how the attention mechanism operates, which essentially re-uses keys and values generated for the first term, to generate the second in (18b). This model for g_y is visualized in Fig. 2. Next, we elaborate about the proposed *fixed past causal attention* that constructs the TREET and its variation, *modified fixed past causal attention* which is required for the reference measurement distribution.

2) FIXED PAST CAUSAL ATTENTION

To comply with (20) in Section III-A, the model must avoid using future inputs relative to the prediction point and operate with a fixed sequence length of l. To enforce this constraint within the attention mechanism, a masking strategy introduced, termed as *fixed past causal attention* (FPCA). Unlike the standard causal mask $M \in \mathbb{R}^{L\times L}$ that simply prevents access to future values, FPCA employs a Toeplitz-like banded mask $M' \in \mathbb{R}^{L\times L}$ defined as:

$$M'_{[i,j]} = \begin{cases} 1 & \text{if } j - i < l \text{ and } j \geq i \\ -\infty & \text{otherwise} \end{cases} \tag{24}$$

This ensures that each attention query attends only to the current and previous $l - 1$ inputs, maintaining a fixed-length

historical context. The FPCA is given by

$$\mathsf{FPCA} := \mathsf{softmax}(QK^\top \odot M')V \tag{25}$$

FPCA matrix is visualized in (22), as shown at the bottom of the page. Note that only the last $L - l + 1$ results from the transformer outputs sequence are used, in order to keep the past information fix to length l. Hence, the complexity of the proposed attention operation is $\mathcal{O}(Lld_o)$.

3) REFERENCE SAMPLING WITH FPCA

This section describes the calculation of $g_y(\widetilde{Y}_l, Y^{l-1})$ (20a). Denote the scoring value after the softmax operation of FPCA for query q_i with key k_j as

$$\mathcal{R}_{(i,j)} := \frac{e^{q_i \cdot k_j}}{\sum_{m=i-l}^{i} e^{q_i \cdot k_m}}, \quad j \le i. \tag{26}$$

Since FPCA with memory parameter l is used, the attention output at time t can be written as $\mathsf{FPCA}_t = \mathcal{R}_{(t,t-l)}v_{t-l} + \mathcal{R}_{(t,t-l+1)}v_{t-l+1} + \ldots + \mathcal{R}_{(t,t)}v_t$. In order to calculate $g_y(\widetilde{Y}_t, Y_{t-l}^{t-1})$, FPCA should use information from the keys and values that were used to calculate $g_y(Y_{t-l}^{t})$. To that end, we modify the FPCA mechanism as follows:

$$\widetilde{\mathcal{R}}_{(i,j)} := \begin{cases} \dfrac{e^{\widetilde{q}_i \cdot k_j}}{e^{\widetilde{q}_i \cdot \widetilde{k}_i} + \sum_{m=i-l}^{i-1} e^{\widetilde{q}_i \cdot k_m}} & , \quad j < i \\ \dfrac{e^{\widetilde{q}_i \cdot \widetilde{k}_i}}{e^{\widetilde{q}_i \cdot \widetilde{k}_i} + \sum_{m=i-l}^{i-1} e^{\widetilde{q}_i \cdot k_m}} & , \quad i = j. \end{cases} \tag{27}$$

In this case, the modified FPCA for time t can be written as $\mathsf{Modified\text{-}FPCA}_t = \widetilde{\mathcal{R}}_{(t,t-l)}v_{t-l} + \widetilde{\mathcal{R}}_{(t,t-l+1)}v_{t-l+1} + \ldots + \widetilde{\mathcal{R}}_{(t,t)}\widetilde{v}_t$. Summarizing, the second term (20a) generated by the modified FPCA, contains all keys and values of the relative past and query, key and value for the current present, which are a function of the reference distribution. The dot-product matrix between queries and keys is visualized in (23), as shown at the bottom of the page, for $l \le L$, and modified FPCA is written as $\mathsf{Modified\text{-}FPCA} = \mathsf{softmax}(\widehat{QK^\top} \odot M')V$. FPCA and modified FPCA immediately extend to multi-head setting.

In our implementation, the reference input $\widetilde{Y}$, for the second term in (20a) are drawn from the uniform measure on the bounding box of the current batch of Y samples, while theoretically, our method allows to draw from any positive continuous distribution measure; for further details check Appendix B. The implementation of $\widehat{\mathsf{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}(D_n, g_{xy})$ (20b) is obtained by concatenating the X^l values with the corresponding Y^l or $(\widetilde{Y}_l, Y^{l-1})$ values for both the FPCA and the modified FPCA, respectively.

IV. OPTIMIZATION OF ESTIMATED TRANSFER ENTROPY

Many data-driven problems reduce to steering an input distribution to maximize information flow, analogous to reinforcement learning, where a policy is learned to maximize expected reward. Thus, applications can leverage TE optimization by controlling the distributions of processes $\mathbb{X}$ and $\mathbb{Y}$, either independently or jointly. Communication channel capacity offers a canonical example: it equals the DI rate and, by Section II-C, can be estimated via TE. Because capacity embodies the fundamental limit of information flow, demonstrating TREET optimization with on this task provides a proof-of-concept readily generalizable to other TE-based control problems.

Remark 1 (Channel capacity): Consider channels with and without feedback links from the channel output back to the encoder. The feedforward capacity of a channel sequence $\{P_{Y^n\|X^n}\}$, *for* $n \in \mathbb{N}$ *is*

$$C_{\mathsf{FF}} = \lim_{n\to\infty} \sup_{P_{X^n}} \frac{1}{n} \mathsf{I}(X^n; Y^n), \tag{28}$$

and the feedback capacity is

$$C_{\mathsf{FB}} = \lim_{n\to\infty} \sup_{P_{X^n\|Y^{n-1}}} \frac{1}{n} \mathsf{I}(X^n \to Y^n). \tag{29}$$

The achievability of the capacities is further discussed in [38], [39], and [34] showed that for non-feedback scenario, the optimization problem over $P_{X^n\|Y^n}$ *can be translated to* P_{X^n}, *which support the use of DI rate for both optimization problems [21]. Under some conditions TE can estimate the DI rate, as presented in Section II-C.*

Focusing on channel capacity, and assuming that the input distribution sampling mechanism is controllable, we propose an algorithm for the optimization of estimated TE with respect to the input generator. We refer to this model as the *neural distribution generator* (NDG). The estimated

$$QK^\top = \begin{bmatrix} q_{(t+l)} \cdot k_{(t)} & \ldots & q_{(t+l)} \cdot k_{(t+l)} & -\infty & \ldots & -\infty \\ -\infty & q_{(t+l+1)} \cdot k_{(t+1)} & \ldots & q_{(t+l+1)} \cdot k_{(t+l+1)} & -\infty & \vdots \\ \vdots & -\infty & \ddots & & \ddots & -\infty \\ -\infty & \ldots & -\infty & q_{(t+L)} \cdot k_{(t+L-l)} & \ldots & q_{(t+L)} \cdot k_{(t+L)} \end{bmatrix} \tag{22}$$

$$\widehat{QK^\top} = \begin{bmatrix} \widetilde{q}_{(t+l)} \cdot k_{(t)} & \ldots & \widetilde{q}_{(t+l)} \cdot \widetilde{k}_{(t+l)} & -\infty & \ldots & -\infty \\ -\infty & \widetilde{q}_{(t+l+1)} \cdot k_{(t+1)} & \ldots & \widetilde{q}_{(t+l+1)} \cdot \widetilde{k}_{(t+l+1)} & -\infty & \vdots \\ \vdots & -\infty & \ddots & & \ddots & -\infty \\ -\infty & \ldots & -\infty & \widetilde{q}_{(t+L)} \cdot k_{(t+L-l)} & \ldots & \widetilde{q}_{(t+L)} \cdot \widetilde{k}_{(t+L)} \end{bmatrix}. \tag{23}$$

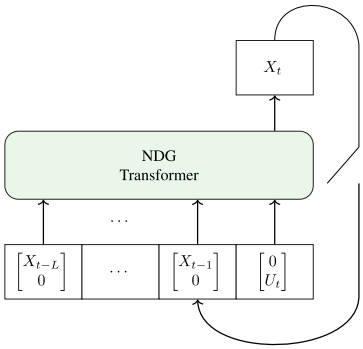


FIGURE 3. The recursive process for the NDG with transformers. If feedback presents, the input includes the past channel output, concatenated with the corresponding value X_i at time i.

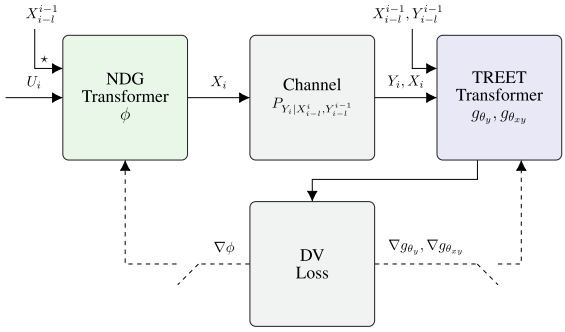


FIGURE 4. Complete system for estimating and optimizing TREET with NDG while Altering between the models to train on. ($\star$) If feedback presents, past channel output realizations included.

TE optimization methodology is inspired by the proposed methods from [21]. However, the adaptation to transformer architectures considers a different implementation of the proposed scheme. As the TREET estimates TE from samples, the NDG is defined as a generative model of the input distribution samples, and is optimized with the goal of maximizing the downstream estimated TE.

Lemma 3 (Optimal TE): Let $(\mathbb{X}, \mathbb{Y})$ *be jointly stationary processes, and the TE with memory parameter* $l \in \mathbb{N}$. *Then, the maximal TE,* $\mathsf{TE}^{\star}_{X\to Y}(l)$, *by* $\mathbb{X}$ *is*

$$\mathsf{TE}^{\star}_{X\to Y}(l) := \sup_{P_{X^l}} \mathsf{I}(X^l; Y_l|Y^{l-1}). \tag{30}$$

The proof of the lemma is given in Appendix D. The lemma suggests that NDG with input sequence length l is enough to achieve maximum TE with memory parameter l, for independently controlling the distribution of $\mathbb{X}$.

The NDG calculates a sequence of channel input X^l through the mapping

$$h_\phi : (U_i, Z_{i-l}^{i-1}) \to X_i^\phi, \qquad i = 1, \ldots, n, \tag{31}$$

where U_i is the random noise drawn from $P_U \in \mathcal{P}_{\mathsf{ac}}(\mathcal{U}), \mathcal{U} \subset \mathbb{R}^{d_x}$, which cause the stochasticity, Z_{i-l}^{i-1} are the past observation of the generated process created by the model, and the channel corresponding outputs if feedback exists. h_ϕ is the parametric NDG mapping with parameters $\phi \in \Phi$. By the functional representation lemma [40] and the restated lemma in [21], the distribution of X is achievable from an NN function. After l iterations, with Z_{i-l}^{i-1} storing the information of previous iterations, the NDG generates the whole sequence.

Transformers need access to the whole sequence at once, in contrast to RNNs where a single state can theoretically represent the past sequence. Thus, the input sequence to the NDG with transformer is created via past outputs of the transformer itself (and corresponding channel outputs if feedback exists) as depicted in Fig. 3, and the past observations are taken without gradients to prevent back-propagation through iterations. In our implementation, the input convention for each time step is a concatenated vector of $[X_i, U_i]^\top$ to maintain the input structure of samples and random noise for every projection in the network. In relative history time steps, the noise is replaced with a zero vector, and for the present time, the input X_i is replaced with a zero vector.

Algorithm 2 Continuous TREET Optimization

Input: Continuous sequence-to-sequence system $\mathcal{S}$; Observation length $l \in \mathbb{N}$.
Output: $\widehat{\mathsf{TE}}^{\star}_{X\to Y}(U^n; l)$, optimized NDG.

NNs initialization g_{θ_y}, $g_{\theta_{xy}}$ and h_ϕ with corresponding parameters $\theta_y, \theta_{xy}, \phi$.
repeat
  Draw noise U^m, $m < n$.
  Compute batch B_m^ϕ sized m using NDG, $\mathcal{S}$
  if training TREET **then**
    Perform TREET optimization - Step 1 in Algorithm 1.
  else Train NDG
    Compute $\widehat{\mathsf{TE}}_{X\to Y}(B_m^\phi, g_{\theta_y}, g_{\theta_{xy}}, h_\phi; l)$ using (19a).
    Update NDG parameters:
    $\phi \leftarrow \phi + \nabla_\phi \widehat{\mathsf{TE}}_{X\to Y}(B_m^\phi, g_{\theta_y}, g_{\theta_{xy}}, h_\phi; l)$
until convergence criteria.
Draw U^m to produce l length sequence and evaluate $\widehat{\mathsf{TE}}_{X\to Y}(D_n^\phi; l)$.
return $\widehat{\mathsf{TE}}^{\star}_{X\to Y}(U^n; l)$, optimized NDG.

To estimate and optimize the TE at once, we jointly training the TREET and the NDG. As described in Algorithm 2 and Fig. 4, in each iteration only one of those model is updated by maximizing each DV potentials, $\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}$, $\widehat{\mathsf{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}$, for TREET model and by maximizing $\widehat{\mathsf{TE}}_{X\to Y}$ for NDG model. The entire pipeline of one iteration creates a sequence length l from the NDG, by iterating it l times with some initiated zero values - which creates the dataset, $D_n^\phi = (X^{\phi,n}, Y^{\phi,n})$. Afterwards, feeding each sample sequence from the dataset to TREET for achieving the corresponding networks' outputs as mentioned back in Algorithm 1, which constructs the loss that generate gradients backward according to each model. Next, we demonstrate the power of the proposed method for estimating the channel capacity.

V. EXPERIMENTAL RESULTS

Evaluation of TREET comprises four interconnected experiments that build upon one another to tell a unified narrative: first, assessing core TE estimation accuracy; next, harnessing TREET for optimization of information flow; then demonstrating TREET's built-in distribution-estimation utility; and finally applying it as a feature-analysis tool on real-world clinical data, illustrating how foundational performance extends seamlessly to practical applications.

Section V-A benchmarks TE estimation on synthetic long-memory data, extending [22] and comparing TREET against TENE and Copnet. Section V-B integrates TREET with NDG to optimize communication channel capacity, compared with DINE, the RNN-based DI estimator [21], and Section V-C further analyses the optimization and estimation memory performance. While estimation experiments are about continuous spaces of distributions, discrete spaces can easily be applied to the algorithm of TREET. Section V-D applies TREET to PDF estimation of memory processes, demonstrating adaptability to continuous-space distribution tasks. Section V-E presents a real-world case study using TREET for feature analysis on an Apnea patients dataset.

All experiments consider a TREET network with a single FPCA layer follows by a single FF layer, and for the optimization procedure the NDG consists of transformer with a single causal-attention layer and a single FF layer. Further details about the implementation[3] are provided in Appendix F.

A. TRANSFER ENTROPY BENCHMARK

In order to present TREET as robust estimator, we extend the benchmark from [22] to handle higher orders of TE. Considering the following system:

$$Y_t = \begin{cases} Z_t, & \text{if } Y_{t-1} < \lambda, \\ \rho X_{t-1} + \sqrt{1-\rho^2} Z_t, & \text{else}, \end{cases} \tag{32}$$

where $\lambda \in \mathbb{R}$ is the process threshold, $\rho \in [0,1]$ determines the dependency between X_{t-1} and Z_t, for X_t, Z_t i.i.d. $\mathcal{N}(0,1)$. The following equality holds - $\mathsf{TE}_{X\to Y}(l) = \mathsf{TE}_{X\to Y}(1), \quad \forall l \geq 1$ [22]. Comparing TREET with Copnet and TENE.

Copnet [18] is empirical-density based non-parametric method that estimates the TE by four different values of copula entropy, which each one equals to a different MI value. We edited their version of estimator to handle conditioning on longer context length than one. Another estimator is TENE [22], parametric estimator, which utilizes the DV representation for estimation of two MI terms which are non-conditional, thus is prone to break with higher orders of TE since the input dimension to the DV optimization function will increase. For a fair comparison, the experiment was recreated using NN architectures with similarly sized parameters for both TENE and TREET and a training batch size of 1024.

[3]The code implementation can be found at our Github repository https://github.com/omerlux/TREET

TABLE 1. TE estimation extended benchmark.

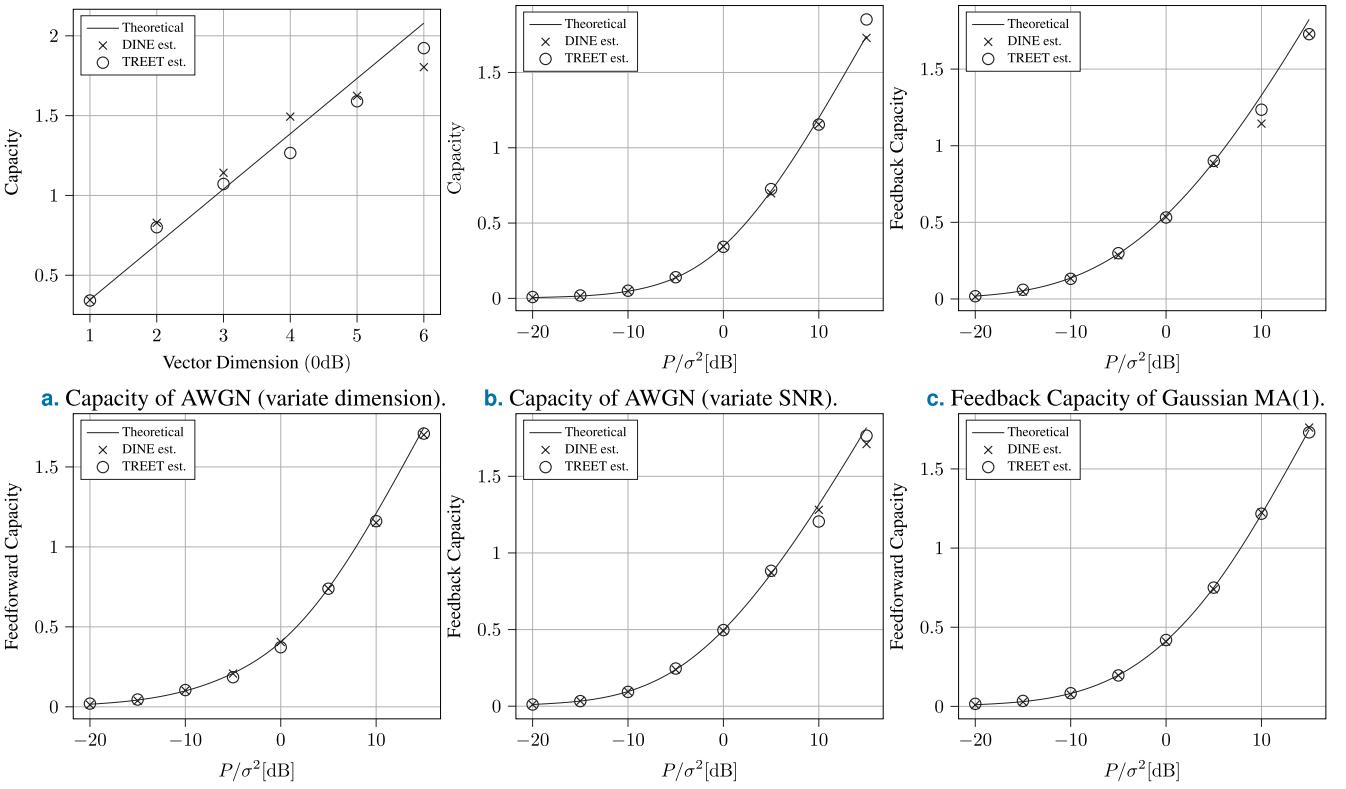
	λ	-3	-2	-1	0	1	2	3
Model, l \ Ground Truth		0.829	0.811	0.699	0.415	0.132	0.019	0.001
TREET	1	0.812	0.792	0.667	0.395	0.126	0.016	0.0
	2	0.826	0.796	0.681	0.392	0.117	0.014	0
	4	0.825	0.798	0.682	0.393	0.115	0.013	0
	7	0.82	0.799	0.679	0.405	0.121	0.015	0.001
	9	0.815	0.802	0.686	0.403	0.126	0.017	0.001
	19	0.824	0.805	0.69	0.405	0.128	0.017	0.001
	49	0.829	0.811	0.694	0.409	0.128	0.018	0.001
	99	0.829	0.81	0.693	0.41	0.115	0.017	0.001
TENE	1	0.823	0.807	0.696	0.416	0.126	0.014	0
	2	0.814	0.782	0.688	0.39	0.115	0.013	0
	4	0.43	0.76	0.602	0.382	0.119	0.013	-0.001
	7	>10	>10	0.698	0.354	0.09	0.013	0.0
	9	>10	>10	>10	0.359	0.091	0.014	0.0
	19	>10	>10	>10	1.796	0.038	0.021	0
	49	<-10.0	<-10.0	<-10.0	>10	0.027	0.01	-0.002
	99	>10	>10	>10	>10	0.102	-0.002	-0.002
Copnet	1	0.835	0.81	0.688	0.397	0.111	0.0	-0.022
	2	0.819	0.786	0.676	0.377	0.106	-0.004	-0.017
	4	0.9	0.864	0.747	0.469	0.193	0.087	0.079
	7	1.297	1.268	1.135	0.903	0.649	0.571	0.575
	9	1.703	1.679	1.558	1.357	1.106	1.054	1.053
	19	4.862	4.834	4.75	4.574	4.431	4.428	4.432
	49	>10	>10	>10	>10	>10	>10	>10
	99	>10	>10	>10	>10	>10	>10	>10

Comparing TREET, TENE and Copnet. The process is given by (32) and the true TE can be calculated for varying λ [22]. The color intensity signifies the extent of deviation from the ground truth. The value is constant for any order - $l \geq 1$ since $\mathsf{TE}_{X\to Y}(l) = \mathsf{TE}_{X\to Y}(1)$. Our proposed benchmark is an extended version of the one presented in [22] to include variable length input to calculate different orders of TE. In the given benchmark, ρ is set to 0.9, while the λ and l parameters are varied.

As shown in Table 1, TREET achieves performance comparable to TENE and Copnet for TE parameter $l = 1$, and process parameter $\rho = 0.9$. However, as l increases, both TENE and Copnet struggle to estimate TE accurately due to the increasing input dimensionality associated with longer context windows. In contrast, TREET effectively handles the additional temporal information, even for $l = 99$. This highlights the advantages of TREET's architecture in accommodating time dependencies.

B. CAPACITY ESTIMATION FOR FINITE MEMORY PROCESSES

As shown in Section II-C, under some conditions TE is equal to the DI rate and converges to it. DINE [21] proved that it can estimate a capacity of stationary channels by optimizing the DI rate estimator and the input distribution of the channel P_X, as Remark 1 mentions about channel capacities. Our



a. Capacity of AWGN (variate dimension). **b.** Capacity of AWGN (variate SNR). **c.** Feedback Capacity of Gaussian MA(1).

d. Feedforward Capacity of Gaussian MA(1). **e.** Feedback Capacity of Gaussian AR(1). **f.** Feedforward Capacity of Gaussian AR(1).

FIGURE 5. **Channel capacity estimations. (a) Capacity of AWGN channel as a function of vector dimension for the 0 dB case. (b) Capacity of AWGN channel as a function of SNR. (c) Feedback capacity of Gaussian MA(1) channel with variate SNR. (d) Feedforward capacity of Gaussian MA(1) channel with variate SNR. (e) Feedback capacity of Gaussian AR(1) channel with variate SNR. (f) Feedforward capacity of Gaussian AR(1) channel with variate SNR.*P* represents the power constraint and the noise parameter is σ^2.**

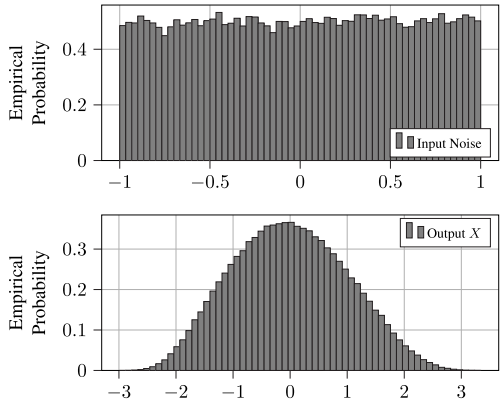


FIGURE 6. **NDG input noise and output *X*, 0 SNR case.**

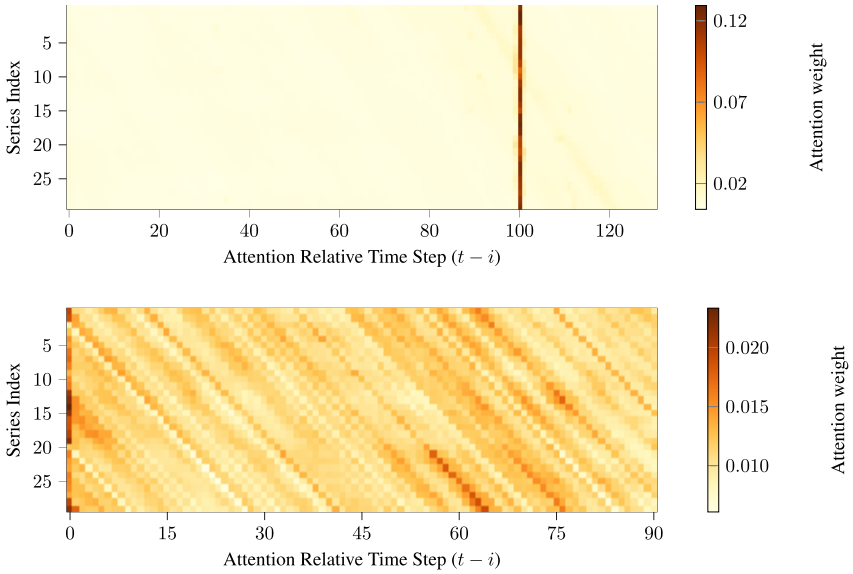
experiments have shown that for memory-less channels and for memory channels with and without feedback, TREET (for an appropriate l order) can approximate the DI rate, which is estimates the capacity, with the joint optimization procedure of the input distribution (NDG) and TE estimation. All of the results are presented in Fig. 5. Important to note that channel input constraints are essential for ensuring that the transmitted signals are well-suited to the channel's characteristics and limitations. In this experiment, the power constraint on the input signal of the channel, which is implemented via normalizing a batch of samples to a certain statistics according to the power constraint. While we present results on continuous-valued processes, TREET can be optimal for discrete data, using methods, such as reinforcement learning optimization [41].

1) AWGN CHANNEL

Consider additive white Gaussian noise (AWGN) channel with i.i.d. noise,

$$Z_t \sim \mathcal{N}(0, \sigma^2),$$
$$Y_t = X_t + Z_t, \qquad t \in \mathbb{Z},$$

X_t is the channel's input sequence, coupled with the average power constraint $\mathbb{E}[X_t^2] \leq P$. The capacity of this channel is simple for analytical calculation, and is given by the following formula $\mathrm{C} = 0.5 \log\left(1 + P/\sigma^2\right)$. Since the process is memoryless, maximized both DI rate and TE (for any $l \geq 0$) coincide with the channel capacity. We estimated and optimized the TREET according to Algorithm 2 and compared the model performance with the DI rate estimation and optimization. Results are presented in Fig. 5a and Fig. 5b. It can be seen that both are estimating the right



a. 130 input length.

b. 90 input length.

FIGURE 7. Attention weights at training convergence of TREET optimized by NDG. Each row in the matrices represent a different input sequence and the columns are the weights of past values i from current prediction t (i.e. $i = 0$ represent the present prediction t). For the GMA(100) process, it can be observed that giving enough time-steps, TREET easily observes at the related time $i = 100$ where the information needed to be gathered from, whereas shorter length lead to instability of training.

TABLE 2. Capacity estimation for the GMA(100) channel at 0 SNR with varying memory lengths. The true capacity is 0.405 [nats]. In this setup, the memory length l for DINE corresponds to the *bptt* length of the LSTM, and for TREET, it represents the sequence input length. DINE estimates the DI rate regardless of input length, whereas TREET estimates the order-l TE. Each value is averaged over ten experiments with different random seeds. Best estimations in bold.

l	TREET		DINE	
	Estimated capacity [nat]	Absolute error (%)	Estimated capacity [nat]	Absolute error (%)
60	0.19	53	**0.30**	**25**
70	0.29	29	**0.35**	**15**
80	0.28	31	**0.34**	**17**
90	0.29	28	**0.30**	**26**
100	**0.37**	**9**	0.36	12
110	**0.38**	**7**	0.33	20
120	**0.36**	**11**	0.33	18
130	**0.36**	**11**	0.34	17
140	**0.35**	**14**	0.26	35

capacities which are the MI, although their access to multiple past observations. Moreover, changing the input dimension changes the estimated capacities, as expected. Note that larger dimensions cause error of in estimation and still is an open academic research [28], [29], [30]. To further analyze the learned distribution, we visualize the optimized NDG mapping in Fig. 6. It can be seen that the optimized NDG maps the uniform ($U \sim [-1, 1]$) inputs into Gaussian samples. This observation meets our expectations, as the distribution that achieve the channel capacity in AWGN is the Gaussian distribution [42].

2) GAUSSIAN MA(1) CHANNEL

Given a Moving Average (MA) Gaussian noise channel with order 1,

$$Z_t = N_t + \alpha N_{t-1},$$
$$Y_t = X_t + Z_t, \qquad t \in \mathbb{Z},$$

where $N_t \sim \mathcal{N}(0, \sigma^2)$ are i.i.d. and X_t is the input to the channel with power constraint $\mathbb{E}[X_t^2] \leq P$. We apply Algorithm 2 to both feedforward and feedback settings, comparing with ground truth solutions and the DI-based scheme. The feedforward capacity can be calculated with the water-filling algorithm [43], whereas the feedback capacity can be calculated by the root of forth order polynomial equation [44]. As seen in Fig. 5c and Fig. 5d, our method successfully estimated the capacity for a wide range of SNR values, compared to previous method.

3) GAUSSIAN AR(1) CHANNEL

The case of autoregressive (AR) Gaussian noise channel of order 1 is similar,

$$Z_t = N_t + \alpha Z_{t-1},$$
$$Y_t = X_t + Z_t, \qquad t \in \mathbb{Z},$$

where $N_t \sim \mathcal{N}(0, \sigma^2)$ are i.i.d. and X_t is the input to the channel with power constraint $\mathbb{E}[X_t^2] \leq P$. The capacity is also affected by the existence of feedback. Feedforward capacity can be solved with the water-filling

TABLE 3. Comparison of density estimation models on the HMM process without state delay ($\beta = 0$), with Gaussian and uniform noise. Results show KL divergence and TV distance relative to the analytical conditional density, averaged over ten trials. Best in **bold**; second-best underlined.

Model	Gaussian		Uniform	
	KLD	TV	KLD	TV
Kalman	1.076	0.467	1.304	0.408
KDE	1.135	0.608	0.847	0.417
MDN	1.098	0.632	0.667	0.460
DINE	**0.795**	0.525	**0.475**	**0.291**
TREET	0.797	**0.524**	0.482	0.296

TABLE 4. Memory-delay analysis of density estimation on the HMM process ($\beta \neq 0$) with varying state delay k. TREET and DINE are compared in terms of mean KL divergence and TV distance relative to the analytical density, averaged over ten trials. Best in **bold**; second-best underlined.

k	TREET		DINE		MDN		Kalman	
	KL	TV	KL	TV	KL	TV	KL	TV
2	0.933	0.569	0.934	0.570	1.460	0.708	**0.636**	**0.405**
5	1.036	0.601	**0.875**	**0.556**	1.454	0.707	1.284	0.599
10	**0.984**	**0.585**	1.299	0.662	1.450	0.706	1.239	0.603
15	**0.992**	**0.587**	1.441	0.704	1.448	0.707	1.232	0.602
25	**0.993**	**0.587**	1.451	0.706	1.443	0.704	1.196	0.598
50	**0.993**	**0.589**	1.452	0.707	1.453	0.707	1.193	0.597

algorithm [43] and [45] prove how to achieve the feedback capacity analytically. Fig. 5e and Fig. 5f compare the results TREET estimator with the previous one, showcasing it's successful estimation on variated SNR values.

C. CAPACITY ESTIMATION - MEMORY ANALYSIS

Previous experiments showed the capability of correctly estimating TE. Delving deeper into the memory effectiveness of TREET, we tested Gaussian MA channel, as presented before, but with time delay of 100 steps.

$$Z_t = N_t + \alpha N_{t-100},$$
$$Y_t = X_t + Z_t, \qquad t \in \mathbb{Z}.$$

It is notable that to achieve the correct capacity, the information from $t - 100$ steps must be an input to the TREET, hence any TE estimation with order $l < 100$ will result a false estimation value. The estimation optimization algorithm is tested with shorter input length and longer input length, than the demanded one. Fig. 7a shows the analysis of the attention weights related to the $\widehat{\mathsf{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}$ model with $l = 130$ input steps (i.e. $\mathsf{TE}_{X\to Y}(130)$), at the convergence stages of training, and Fig. 7b considers the same for $l = 90$ input steps ($\mathsf{TE}_{X\to Y}(90)$). It is evident from Fig. 7 that TREET effectively captures relevant information from long time series when provided with a sufficient input length. However, when the input length is too short, critical information is lost, leading to a complete noisy attention weights.

Additionally, we compare TREET with DINE for different input lengths on the GMA(100) channel. It is important to note that DINE can theoretically deal with long sequences even if given a shorter backpropagation through time (*bptt*) input length than the process memory due to state propagation. However, as seen in Table 2, DINE struggles to estimate the correct capacity under long memory. In contrast, TREET achieves its best estimation for memory lengths larger than 100, with an absolute error rate of less than 14%. When the TREET memory is shorter than the channel memory, its performance significantly degrades, with absolute error rate no less than 28%.

D. TREET-BASED DENSITY ESTIMATION

In this section, we demonstrate how optimized TREET networks can be used for density estimation - a byproduct of the TREET optimization procedure that requires no additional training. Specifically, TREET enables the estimation of the conditional density function of the observed continuous process $\mathbb{Y}$, namely $P_{Y_t|Y^{t-1}}$, directly from the learned network $\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}$. Furthermore, since TREET optimizes both $\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}$ and $\widehat{\mathsf{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}$, it also provides access to the more informative conditional density $P_{Y_t|Y^{t-1},X^t}$, enabling flexible estimation depending on the available context.

Given the optimization of $\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}$, the network outputs the log-likelihood ratio between the conditional density $P_{Y_t|Y^{t-1}}$ and an explicitly known reference density $P_{\widetilde{Y}_t}$. Thus, the true conditional density is recovered via:

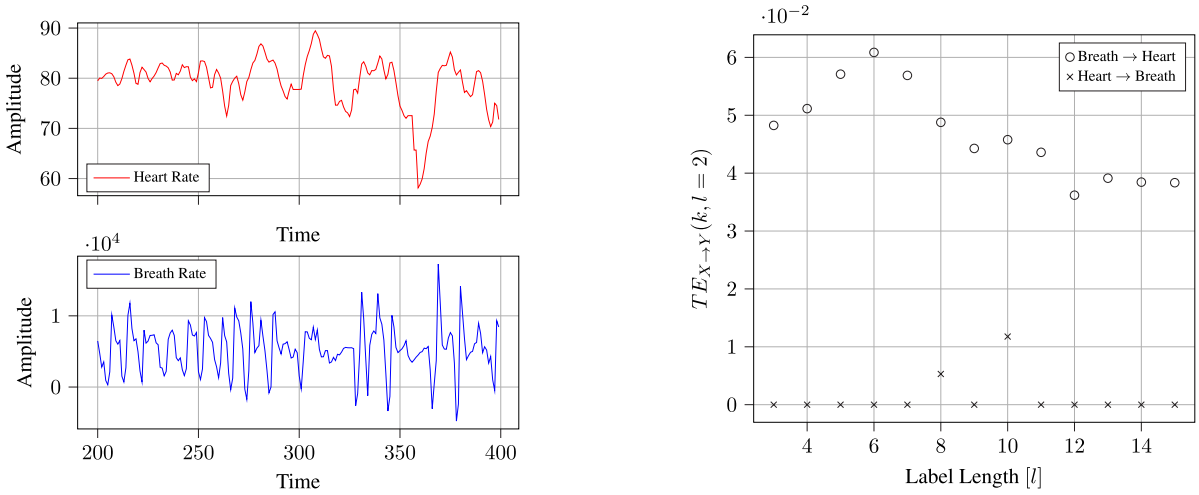
$$P_{Y_t|Y^{t-1}}(y_t|y^{t-1}) \approx \exp\left(\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(D_n, g_y)\right) \cdot \widetilde{P}_{Y_t}(y_t). \tag{33}$$

This expression defines an unnormalized density function, since the optimized $\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}$ approximates the log-likelihood ratio (up to an additive constant) between the true conditional density and the reference density, as guaranteed by the DV representation. Consequently, the probability given by this equation is not directly normalized. To obtain a properly normalized conditional density, the continuous domain of Y_t is discretized into a sufficiently fine grid, evaluate the unnormalized density at each grid point, and numerically normalize by summing over all points. In practice, this numerical integration approach provides a valid approximation of the conditional density.

Our method builds upon previous work [46], which addressed non-time-related density estimation, and [47], who proposed a time-related approach based on the DINE framework. While DINE is designed to capture limiting (stationary) densities, TREET leverages a fixed memory length, aligning naturally with memory-aware estimation.

E. FEATURES ANALYSIS IN PHYSIOLOGICAL DATA

Motivated by the results in [4, Chapter 7.1], we tested the TREET on the Apnea dataset from Santa Fe Time Series



a. Sampled sequence of Apnea dataset.

b. TE estimation for variable length history of Y.

FIGURE 8. Transfer Entropy estimation on physiological data. The Apnea dataset consists of heart rate and breathing rate measurements. The information flow for patient diagnosis is diagnosed. Patients with Apnea experience breathing cessation, leading to alterations in heart rate during sleep.

Competition[4] [31], [32]. This is a multivariate data that have been recorded from a diseased patient of Apnea in a sleep laboratory, which is a condition characterized by brief, involuntary pauses in breathing, particularly during sleep. Each sample consists of three different variables from a specific time, with a sample rate of 2 Hz. The three features are heart rate, chest volume (which is the respiration force) and blood oxygen concentration. A sampled sequence of heart rate and breath rate (chest volume) is presented in Fig. 8a.

The hidden and observed processes are modeled as

$$\begin{aligned} X_t &= \alpha X_{t-1} + \beta X_{t-k} + W_t, \\ Y_t &= \gamma X_t + V_t, \end{aligned} \tag{34}$$

where W_t, V_t are i.i.d. innovations (either Gaussian, zero mean and $\sigma^2 = 0.5$, or uniform on $[-1, 1]$). The process is stationary (and ergodic) iff every root z of the characteristic polynomial $1 - \alpha z - \beta z^k = 0$ lies outside the unit circle, $|z| > 1$ [48]. We conduct two experiments - the first is classic HMM without any state delay ($k = 0$), with parameters $\alpha = 0.9$, $\beta = 0$, $\gamma = 0.5$, $\sigma_W^2 = \sigma_V^2 = 0.5$, and fix the memory length of all models to $l = 3$. TREET employs this $l = 3$ memory parameter for TE estimation, and DINE's *bptt* is likewise truncated to three steps. The second experiment is memory analysis by applying state-delay, with $\beta \neq 0$ and variate delay k. Set $\alpha = 0.001$, $\beta = 0.9$, $\gamma = 0.9$, $\sigma_W^2 = \sigma_V^2 = 0.5$, and TREET's memory $l = k + 5$ and compared all but KDE due to computational complexity of the algorithm for larger ks.

For the classic HMM, TREET compared against several baselines: DINE; mixture density networks (MDN) [49], which output a Gaussian mixture model for conditional densities; conditional KDE [50], which performs non-parametric smoothing over the conditioning variables; and the Kalman filter [51], the optimal solution in the Gaussian HMM (without delay) setting. To evaluate TREET's density-estimation performance, we compare the estimated conditional density of observations $P_{Y_t|Y^{t-1}}$ to the analytical conditional density $P_{Y_t|X_t}$, which assumes full access to the latent state. As metrics, KL divergence is computed and the total variation (TV) distance between each model's estimate and the analytical density.

Table 3 reports these distances for the classic HMM experiment averaged over ten trials. Under both Gaussian and uniform noise, TREET matches or exceeds the performance of DINE, mixture density networks, conditional kernel density estimation, and the Kalman filter. Table 4 summarizes the memory—delay analysis. For each state delay $k \in \{2, 5, 10, 15, 25, 50\}$ we set the memory for each algorithm to $l = k + 5$, e.g. TE parameter for TREET and *bptt* for DINE. and report mean KL divergence and TV distance over ten trials. TREET matches or improves upon DINE's KL and TV scores for every state delay, and begins to pull away from both MDN and the Kalman filter once $k \geq 10$, demonstrating its ability to capture long—range dependencies. At very short delays, however, the Kalman filter and DINE remains competitive – reflecting their explicit state-update schemes suited to near-Markovian dynamics.

Overall, these two experiments confirm that TREET provides accurate, memory—aware conditional density estimates with no additional training beyond TE optimization, outperforming existing methods even as the required memory length grows.

Determining the interaction between different physiological features is crucial for diagnosing diseases by revealing causal connections within the human body, enabling targeted

[4]https://physionet.org/content/santa-fe/1.0.0/

diagnostics and personalized treatments according to identified risk factors. Therefore, TREET was applied to determine the magnitude and direction of information transfer in the given setting. Both heart rate and breath features were used to compare the TE measurements $\mathsf{TE}_{\mathsf{Breath}\to\mathsf{Heart}}(k,2)$ and $\mathsf{TE}_{\mathsf{Heart}\to\mathsf{Breath}}(k,2)$ for variable length k that is the Y process's history observations length. The results are presented in Fig. 8b and are aligned with the results of [4] (for $k,l=2$) and extend it with tests on variate history length of Y process.

Notably, for every considered k, the TE from the breath process to the heart process is consistently higher, aligning with the diagnosis of Apnea disorder (abrupt cessation of breathing during sleep). In terms of information flow, our results indicate that the breathing process transfers more information regarding the behavior of the heart rate process, than the opposite direction, since the value of the estimated TE is consistently higher in comparison, whereas in information theory it essentially reflect that the X process has more to reveal about Y's current value than Y's past itself.

Furthermore, in the influence direction $\mathsf{TE}_{\mathsf{Breath}\to\mathsf{Heart}}$, increasing the number of visible heart rate history samples decreases the information transfer, since a longer history of heart rate provides more insight into future outcomes than the instantaneous breath value. However, $\mathsf{TE}_{\mathsf{Heart}\to\mathsf{Breath}}$ shows that for variate k values, the majority of TE results indicate that the heart process barely affect the breathing process in Apnea patients.

The conclusion of this experiment provides valuable insights and validates established scientific knowledge. While it is generally understood that an increase in heart rate corresponds with an increase in breathing rate, patients with Apnea exhibit a distinct phenomenon: a sudden cessation of muscle activity during inhalation (i.e., breathing cessation) significantly affects the heart rate, contrary to the behavior observed in healthy individuals. This experiment corroborates the findings presented in [4] and [52] and further introduces a novel insight - namely, that the estimated TE decreases with increasing context (history length) of the process Y. This observation aligns with the theoretical expectation that incorporating a longer historical context of Y reduces uncertainty, thus lowering the resulting TE values.

VI. CONCLUSION AND FUTURE WORK

This work presented an attention-based architecture for estimating TE in ergodic and stationary processes. We developed a DV-based neural TE estimator, established its consistency, and introduced a novel modified attention mechanism tailored for the task. Our extended TE benchmark demonstrates its superior performance relative to existing methods. Furthermore, we designed an optimizer for the estimated TE, which was subsequently applied to channel capacity estimation, and proved to perform greatly for higher order TE estimation. In addition, showcased its inherent probability density estimation functionality, which emerges directly from the TE estimation training and finally evaluated TREET's capability in causal feature analysis on the Apnea dataset.

With the increasing popularity of sequential data in contemporary machine learning, TREET will be leveraged for information theoretic analysis and architecture design through the lens of causal information transfer. Potential applications include enhancing predictive probabilistic models, refining feature selection processes, reconstructing complex networks, improving anomaly detection capabilities, and optimizing decision-making in dynamic environments. Moreover, building on the work in [53], the TREET optimization scheme will be extended to sequential data compression tasks.

APPENDIX

A. PROOF OF LEMMA 1

Let $\mathbb{X}$, $\mathbb{Y}$ be two jointly stationary processes that pose the markov property with $l\in\mathbb{N}$ and $m\geq l$, then

$$\begin{aligned}&\mathsf{TE}_{X\to Y}(l)\\&=\mathsf{I}\left(X^{l};Y_l|Y^{l-1}\right)\\&\overset{(a)}{=}\mathsf{I}\left(X_{l-m}^{l};Y_l|Y_{l-m}^{l-1}\right)\\&\overset{(b)}{=}\mathsf{I}\left(X^{m};Y_m|Y^{m-1}\right)\\&=\mathsf{TE}_{X\to Y}(m),\end{aligned}\tag{35}$$

where (a) is true from the markovity, and (b) transition is index shift which valid due to process stationarity. Observing the DI rate,

$$\begin{aligned}&\mathsf{I}(\mathbb{X}\to\mathbb{Y})\\&\overset{(a)}{=}\lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\mathsf{I}\left(X^{i};Y_i|Y^{i-1}\right)\\&=\lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\left[\mathsf{h}\left(Y_i|Y^{i-1}\right)-\mathsf{h}\left(Y_i|X^{i},Y^{i-1}\right)\right]\\&\overset{(b)}{=}\lim_{n\to\infty}\mathsf{h}\left(Y_n|Y^{n-1}\right)-\mathsf{h}\left(Y_n|X^{n},Y^{n-1}\right)\\&=\lim_{n\to\infty}\mathsf{I}\left(X^{n};Y_n|Y^{n-1}\right)\\&\overset{(c)}{=}\lim_{n\to\infty}\mathsf{TE}_{X\to Y}(n)\\&\overset{(d)}{=}\mathsf{TE}_{X\to Y}(m),\end{aligned}\tag{36}$$

where the limit in (a) exists whenever the joint process is stationary, transition (b) is valid because the limit exists for each series of conditional entropies, and since conditioning reduces entropy, the limit for each normalized sum of series is $\lim_{n\to\infty}\mathsf{h}\left(Y_n|Y^{n-1}\right)$, $\lim_{n\to\infty}\mathsf{h}\left(Y_n|X^{n},Y^{n-1}\right)$, respectively [42, Theorem 4.2.1]. Since the limit exists for the conditional MI, transition (c) is valid by definition of TE, and the TE with limit of parameter m exists, and (d) is from (35) for $n\geq m$. Concluding the proof. □

B. PROOF OF THEOREM 3

The proof following the steps of representation step - represents TE as a subtraction of two DV potentials, estimation step - proves that the DV potentials is achievable by empirical mean of a given set of samples, and approximation step - shows that the estimator converges to TE with the corresponding memory parameter l. Thus concluding that our estimator is a consistent estimator for the TE.

Let $\{X_i, Y_i\}_{i\in\mathbb{Z}}$ be the two values of a process $\mathbb{X}, \mathbb{Y}$ respectively, and $\mathbb{P}$ be the stationary ergodic measure over $\sigma(\mathbb{X}, \mathbb{Y})$. Define $P_{X^n,Y^n} := \mathbb{P}|_{\sigma(X^n,Y^n)}$ as the n-coordinate projection of $\mathbb{P}$, where $\sigma(X^n, Y^n)$ is the σ-algebra generated by (X^n, Y^n). Let $D_n = (X^n, Y^n) \sim P_{X^n,Y^n}$. Let $\widetilde{Y} \sim Q_Y$ be an absolutely continuous PDF, independent of $\{(X_i, Y_i)\}_{i\in\mathbb{Z}}$ and its distribution noted as $\widetilde{P}_Y$. The proof is divided to three steps - variational representation, estimation from samples and functional approximation.

1) REPRESENTATION OF TE

In order to write TE as a difference between two KL divergences, recall Lemma 2 - let,

$$\mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l} := \mathsf{D}_{\mathsf{KL}}\left(P_{Y_l|Y^{l-1}}\|\widetilde{P}_{Y_l}\middle|P_{Y^{l-1}}\right), \tag{37a}$$

$$\mathsf{D}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l} := \mathsf{D}_{\mathsf{KL}}\left(P_{Y_l|Y^{l-1}X^l}\|\widetilde{P}_{Y_l}\middle|P_{Y^{l-1}X^l}\right). \tag{37b}$$

Then

$$\mathsf{TE}_{X\to Y}(l) = \mathsf{D}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l} - \mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}. \tag{38}$$

This lemma is proved in Appendix C. Estimating the KL divergence is applicable with the DV representation 2

$$\begin{aligned}\mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l} = &\sup_{f_y:\Omega_{\mathcal{Y}}\to\mathbb{R}} \mathbb{E}\left[f_y\left(Y^l\right)\right]\\ &- \log\mathbb{E}\left[e^{f_y(Y^{l-1},\widetilde{Y}_l)}\right],\end{aligned} \tag{39a}$$

where $\Omega_{\mathcal{Y}} = \mathcal{Y}^l$. For the other term, the DV representation is,

$$\begin{aligned}\mathsf{D}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l} = &\sup_{f_{xy}:\Omega_{\mathcal{X}\times\mathcal{Y}}\to\mathbb{R}} \mathbb{E}\left[f_{xy}\left(X^l, Y^l\right)\right]\\ &- \log\mathbb{E}\left[e^{f_{xy}(X^l,Y^{l-1},\widetilde{Y}_l)}\right],\end{aligned} \tag{39b}$$

where $\Omega_{\mathcal{X}\times\mathcal{Y}} = \mathcal{X}^l \times \mathcal{Y}^l$. The next section refers to (39a) but the claims are the same for (39b).

2) ESTIMATION

By the DV representation, the supremum in (39a) achieved for

$$\begin{aligned}f^\star_{y,l} &:= \log\left(\frac{dP_{Y^l}}{d(P_{Y^{l-1}}\otimes\widetilde{P}_{Y_l})}\right)\\ &= \log p_{Y_l|Y^{l-1}} - \log\widetilde{p}_{Y_l},\end{aligned} \tag{40}$$

where the last equality holds due to $P_{Y^l} \ll P_{Y^{l-1}}\otimes\widetilde{P}_{Y_l}$, both measures have Lebesgue densities. Although it is not mandatory to select the reference measurement as uniform, choosing a uniform reference measurement can result in a constant that can be neutralized to obtain the likelihood function of $Y_l|Y^{l-1}$. This approach allows for simplification and facilitates the estimation process. Empirical means can estimate the expectations in (39a), while applying the generalized Birkhoff theorem [54], stated next:

Theorem 4 (The Generalized Birkhoff Theorem): Let T be a metrically transitive one-to-one measure preserving transformation of the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ onto itself. Let $g_0(\omega), g_1(\omega), \ldots$ be a sequence of measurable functions on Ω converging a.s. to the function $g(\omega)$ such that $\mathbb{E}[\sup_i |g_i|] \le \infty$. Then,

$$\frac{1}{n}\sum_{i=1}^{n} g_i(T^i\omega) \overset{n\to\infty}{\longrightarrow} \mathbb{E}[g], \quad \mathbb{P}-a.s. \tag{41}$$

By applying Theorem 4 and for any $\epsilon > 0$ and sufficiently large n, we have

$$\left|\mathbb{E}\left[f^\star_{y,l}\left(Y^l\right)\right] - \frac{1}{n}\sum_{i=0}^{n-1} f^\star_{y,l}\left(Y_i^{i+l}\right)\right| < \frac{\epsilon}{8},$$

$$\left|\log\left(\mathbb{E}\left[e^{f^\star_{y,l}(Y^{l-1},\widetilde{Y}_l)}\right]\right)\right. \tag{42a}$$

$$\left. - \log\left(\frac{1}{n}\sum_{i=0}^{n-1} e^{f^\star_{y,l}\left(Y_i^{i+l-1},\widetilde{Y}_{i+l}\right)}\right)\right| < \frac{\epsilon}{8}, \tag{42b}$$

where $\{f^\star_{y,l}\}$ is the function of l time steps, that achieves the supremum of $\mathsf{D}_{\mathsf{KL}}\left(P_{Y_l|Y^{l-1}}\|\widetilde{P}_{Y_l}\middle|P_{Y^{l-1}}\right)$. Convergence achieved from the generalized Brikhoff theorem, where the series of functions is the fixed function $f^\star_{y,l}$,

$$\mathbb{E}_n\left[f^\star_{y,l}\left(Y^l\right)\right] := \frac{1}{n}\sum_{i=0}^{n-1} f^\star_{y,l}\left(Y_i^{i+l}\right), \tag{43a}$$

$$\mathbb{E}_n\left[e^{f^\star_{y,l}(Y^{l-1},\widetilde{Y}_l)}\right] := \frac{1}{n}\sum_{i=0}^{n-1} e^{f^\star_{y,l}\left(Y_i^{i+l-1},\widetilde{Y}_{i+l}\right)}. \tag{43b}$$

3) APPROXIMATION

Last step is to approximate the functional space with the space of transformers. Recall that set of causal transformer architectures $\mathcal{G}^Y_{\mathsf{ctf}} := \mathcal{G}^{(d_y,1,l,v_y)}_{\mathsf{ctf}}$, $\mathcal{G}^{XY}_{\mathsf{ctf}} := \mathcal{G}^{(d_x+d_y,1,l,v_{xy})}_{\mathsf{ctf}}$ for given $l, d_y, d_x, v_y, v_{xy} \in \mathbb{N}$. Define

$$\begin{aligned}\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(D_n) := &\sup_{g_y\in\mathcal{G}^Y_{\mathsf{tf}}} \frac{1}{n}\sum_{i=0}^{n-1} g_y\left(Y_i^{i+l}\right)\\ &- \log\left(\frac{1}{n}\sum_{i=0}^{n-1} e^{g_y\left(Y_i^{i+l-1},\widetilde{Y}_{i+l}\right)}\right),\end{aligned} \tag{44}$$

where the DV functions are transformers $g_y \in \mathcal{G}^Y_{\mathsf{ctf}}, g_{xy} \in \mathcal{G}^{XY}_{\mathsf{ctf}}$. We want to prove that for a given $\epsilon > 0$

$$\left|\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(D_n) - \mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}\right| \le \frac{\epsilon}{2}. \tag{45}$$

From Theorem 2, we obtain

$$\mathbb{E}\left[f^\star_{y,l}\left(Y^l\right)\right] = \mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}, \tag{46a}$$

$$\mathbb{E}\left[f^{\star}_{y,l}\left(Y^{l-1},\widetilde{Y}_l\right)\right]=1. \tag{46b}$$

Thus, we seek to bound the following subtraction $\left|\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(D_n)-\mathbb{E}\left[f^{\star}_{y,l}\left(Y_i^{i+l}\right)\right]\right|$. By the given identity $\log(x)\le x-1,\forall x\in\mathbb{R}_{\ge 0}$,

$$\begin{aligned}
&\left|\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(D_n)-\mathbb{E}\left[f^{\star}_{y,l}\left(Y^{l}\right)\right]\right|\\
&=\left|-\mathbb{E}\left[f^{\star}_{y,l}\left(Y^{l}\right)\right]+\sup_{g_y\in\mathcal{G}^{Y}_{\mathsf{tf}}}\left\{\frac{1}{n}\sum_{i=0}^{n-1}g_y\left(Y_i^{i+l}\right)\right.\right.\\
&\quad\left.\left.-log\left(\frac{1}{n}\sum_{i=0}^{n-1}e^{g_y\left(Y_i^{i+l-1},\widetilde{Y}_{i+l}\right)}\right)\right\}\right|\\
&\le\left|1-\mathbb{E}\left[f^{\star}_{y,l}\left(Y^{l}\right)\right]+\sup_{g_y\in\mathcal{G}^{Y}_{\mathsf{tf}}}\left\{\frac{1}{n}\sum_{i=0}^{n-1}g_y\left(Y_i^{i+l}\right)\right.\right.\\
&\quad\left.\left.-\left(\frac{1}{n}\sum_{i=0}^{n-1}e^{g_y\left(Y_i^{i+l-1},\widetilde{Y}_{i+l}\right)}\right)\right\}\right|\\
&\le\left|+\mathbb{E}\left[f^{\star}_{y,l}\left(Y^{l-1},\widetilde{Y}_l\right)\right]-\mathbb{E}\left[f^{\star}_{y,l}\left(Y^{l}\right)\right]\right.\\
&\quad+\sup_{g_y\in\mathcal{G}^{Y}_{\mathsf{tf}}}\left\{\frac{1}{n}\sum_{i=0}^{n-1}g_y\left(Y_i^{i+l}\right)\right.\\
&\quad\left.\left.-\left(\frac{1}{n}\sum_{i=0}^{n-1}e^{g_y\left(Y_i^{i+l-1},\widetilde{Y}_{i+l}\right)}\right)\right\}\right|.
\end{aligned}\tag{47}$$

Due to (42), there exists $N\in\mathbb{N}$ such that $\forall n>\mathbb{N}$

$$\left|\mathbb{E}\left[f^{\star}_{y,l}\left(Y^{l}\right)\right]-\mathbb{E}_n\left[f^{\star}_{y,l}\left(Y^{l}\right)\right]\right|<\frac{\epsilon}{8}, \tag{48a}$$

$$\left|\left(\mathbb{E}\left[e^{f^{\star}_{y,l}\left(Y^{l-1},\widetilde{Y}_l\right)}\right]\right)-\mathbb{E}_n\left[e^{f^{\star}_{y,l}\left(Y^{l-1},\widetilde{Y}_l\right)}\right]\right|<\frac{\epsilon}{8}. \tag{48b}$$

Plugging (48) to (47) gives

$$\begin{aligned}
&\left|\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(D_n)-\mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}\right|\\
&\le\frac{\epsilon}{4}+\left|-\mathbb{E}_n\left[e^{f^{\star}_{y,l}\left(Y^{l-1},\widetilde{Y}_l\right)}\right]-\mathbb{E}_n\left[f^{\star}_{y,l}\left(Y^{l}\right)\right]\right.\\
&\quad+\sup_{g_y\in\mathcal{G}^{Y}_{\mathsf{tf}}}\left\{\frac{1}{n}\sum_{i=0}^{n-1}g_y\left(Y_i^{i+l}\right)\right.\\
&\quad\left.\left.-\left(\frac{1}{n}\sum_{i=0}^{n-1}e^{g_y\left(Y_i^{i+l-1},\widetilde{Y}_{i+l}\right)}\right)\right\}\right|.
\end{aligned}\tag{49}$$

Since the empirical mean of $f^{\star}_{y,l}$ is converging to the expected mean, it is uniformly bounded by some $M\in\mathbb{R}_{\ge 0}$. Since the exponent function is Lipschitz continuous with Lipschitz constant $\exp[M]$ on the interval $(-\infty,M]$, we obtain

$$\begin{aligned}
&\frac{1}{n}\sum_{i=1}^{n}e^{f^{\star}_{y,l}\left(\widetilde{Y}_{i+l},Y_i^{i+l-1}\right)}-e^{g_y\left(\widetilde{Y}_l,Y_i^{i+l-1}\right)}\\
&\le e^{M}\frac{1}{n}\sum_{i=1}^{n}\left|f^{\star}_{y,l}\left(\widetilde{Y}_{i+l},Y_i^{i+l-1}\right)\right.\\
&\quad\left.-g_y\left(\widetilde{Y}_{i+l},Y_i^{i+l-1}\right)\right|.
\end{aligned}\tag{50}$$

Definition 4 is a sub-functions class of the continuous sequence to sequence functions class, and applies to the causal transformer. Thus, concludes that the causal transformer $g\in\mathcal{G}^{(d_i,d_o,l,v)}_{\mathsf{ctf}}$ also applies for the universal approximation theorem, for continuous vector-values functions. Moreover, for our case the output sequence is a scalar value, $\mathcal{U}\subset\mathbb{R}^{l\times d_i}$, $\mathcal{Z}\subset\mathbb{R}^{1\times d_o}$. For given ϵ, M, l and n, denote $g^{\star}_y\in\mathcal{G}^{(d_y,1,l,v)}_{\mathsf{ctf}}$ the causal transformer, such that the approximation error is uniformly bounded $\exp[-M]\times\epsilon/4$ for the final time prediction out from the model. Finally, combining Theorem 1 we have

$$\begin{aligned}
&\left|\widehat{\mathsf{D}}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}(D_n)-\mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}\right|\\
&\le\left(1+e^{M}\right)\frac{1}{n}\sum_{i=1}^{n}\left|f^{\star}_{y,l}\left(\widetilde{Y}_{i+l},Y_i^{i+l-1}\right)\right.\\
&\quad\left.-g^{\star}_y\left(\widetilde{Y}_{i+l},Y_i^{i+l-1}\right)\right|+\frac{\epsilon}{4}\\
&\le\frac{\epsilon}{2}.
\end{aligned}\tag{51}$$

This concludes the proof of (39a). For (39b), note that

$$\begin{aligned}
f^{\star}_{y,l}&:=\log\left(\frac{dP_{Y^l}}{d(P_{X^lY^{l-1}}\otimes\widetilde{P}_{Y_l})}\right)\\
&=\log p_{Y_l|X^lY^{l-1}}-\log\widetilde{p}_{Y_l},
\end{aligned}\tag{52}$$

achieves the supremum. Following the same claims for $\widehat{\mathsf{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}(D_n)$,

$$\left|\widehat{\mathsf{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}(D_n)-\mathsf{D}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}\right|\le\frac{\epsilon}{2}, \tag{53}$$

where

$$\begin{aligned}
&\widehat{\mathsf{D}}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}(D_n)\\
&:=\sup_{g_{xy}\in\mathcal{G}^{XY}_{\mathsf{tf}}}\frac{1}{n}\sum_{i=0}^{n-1}g_{xy}\left(Y_i^{i+l},X_i^{i+l}\right)\\
&\quad-\log\left(\frac{1}{n}\sum_{i=0}^{n-1}e^{g_{xy}\left(Y_i^{i+l-1},X_i^{i+l},\widetilde{Y}_{i+l}\right)}\right).
\end{aligned}\tag{54}$$

With (53) and (51) we end the proof. ■

C. PROOF OF LEMMA 2

Recall

$$\mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}:=\mathsf{D}_{\mathsf{KL}}\left(P_{Y_l|Y^{l-1}}\|\widetilde{P}_{Y_l}\middle|P_{Y^{l-1}}\right), \tag{55a}$$

$$\mathsf{D}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l}:=\mathsf{D}_{\mathsf{KL}}\left(P_{Y_l|Y^{l-1}X^l}\|\widetilde{P}_{Y_l}\middle|P_{Y^{l-1}X^l}\right). \tag{55b}$$

By expanding the first term we obtain,

$$\begin{aligned}
&\mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l}\\
&=\mathbb{E}_{P_{Y^{l-1}}}\left[\mathsf{D}_{\mathsf{KL}}(P_{Y_l|Y^{l-1}}\|P_{\widetilde{Y}_l})\right]\\
&=\int_{\mathcal{Y}^{l-1}}\left[\int_{\mathcal{Y}_l}\log\left(\frac{P(y_l|y^{l-1})}{\widetilde{P}(y^l)}\right)\right.\\
&\quad\left.P(y_l|y^{l-1})d(y_l)\right]P(y^{l-1})d(y^{l-1})
\end{aligned}$$

$$= \int_{\mathcal{Y}^l} \log\left(\frac{P(y_l|y^{l-1})}{\widetilde{P}(y_l)}\right) P(y^l) d(y^l)$$
$$= \int_{\mathcal{X}^l\mathcal{Y}^l} \log\left(\frac{P(y_l|y^{l-1})}{\widetilde{P}(y_l)}\right) P(x^l y^l) d(x^l y^l), \quad (56)$$

and the second term,

$$\begin{aligned}
&\mathsf{D}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l} \\
&= \mathbb{E}_{P_{X^lY^{l-1}}}\left[\mathsf{D}_{\mathsf{KL}}(P_{Y_l|X^lY^{l-1}}\|P_{\widetilde{Y}_l})\right] \\
&= \int_{\mathcal{X}^l\mathcal{Y}^{l-1}} \left[\int_{\mathcal{Y}_l} \log\left(\frac{P(y_l|x^l y^{l-1})}{\widetilde{P}(y_l)}\right)\right. \\
&\quad \left. P(y_l|x^l y^{l-1}) d(y_l)\right] P(x^l y^{l-1}) d(x^l y^{l-1}) \\
&= \int_{\mathcal{X}^l\mathcal{Y}^l} \log\left(\frac{P(y_l|x^l y^{l-1}}{\widetilde{P}(y_l)}\right) P(x^l y^l) d(x^l y^l), \quad (57)
\end{aligned}$$

where $\widetilde{P}_Y$ is a reference density function and is absolutely continuous on $\mathcal{Y}$. Subtructing the two terms resulting,

$$\begin{aligned}
&\mathsf{D}_{Y_l|Y^{l-1}X^l\|\widetilde{Y}_l} - \mathsf{D}_{Y_l|Y^{l-1}\|\widetilde{Y}_l} \\
&= \int_{\mathcal{X}^l\mathcal{Y}^l} \log\left(\frac{P(y_l|x^l y^{l-1})}{P(y_l|y^{l-1}}\right) P(x^l y^l) d(x^l y^l) \\
&= h(Y_l|Y^{l-1}) - h(Y_l|X^l Y^{l-1}) \\
&= \mathsf{TE}_{X\to Y}(l). \quad \square \quad (58)
\end{aligned}$$

D. PROOF OF LEMMA 3

The optimal TE, $\mathsf{TE}^\star_{X\to Y}(l)$, by non-finite memory process $\mathbb{X}$, is given by

$$\mathsf{TE}^\star_{X\to Y}(l) := \lim_{n\to\infty} \sup_{P_{X^l_{-n}}} \mathsf{TE}_{X\to Y}(l). \quad (59)$$

For any $n > 0$ we have,

$$\begin{aligned}
&\sup_{P_{X^l_{-n}}} \mathsf{TE}_{X\to Y}(l) \\
&= \sup_{P_{X^0_{-n}} P_{X^l|X^0_{-n}}} \mathsf{I}\left(X^l; Y_l|Y^{l-1}\right) \\
&= \sup_{P_{X^0_{-n}} P_{X^l|X^0_{-n}}} \int_{\mathcal{X}^l_{-n}\mathcal{Y}^l} P(x^0_{-n}) P(x^l, y^l|x^0_{-n}) \\
&\quad \underbrace{\mathsf{I}\left(X^l; Y_l|Y^{l-1}\right)}_{\text{independent of } x^0_{-n}} d(x^l x^0_{-n} y^l) \\
&\overset{(a)}{=} \sup_{P_{X^l}} \int_{\mathcal{X}^l\mathcal{Y}^l} P(x^l, y^l) \mathsf{I}\left(X^l; Y_l|Y^{l-1}\right) d(x^l y^l) \\
&= \sup_{P_{X^l}} \mathsf{TE}_{X\to Y}(l), \quad (60)
\end{aligned}$$

where (a) follows from the fact that the conditional MI does not depend on x^0_{-n}, thus we can eliminate the integral of $\mathcal{X}^0_{-n}$ that does not affect the conditional MI, as for the supremum. Since (60) is true for any $n \in \mathbb{N}$, with (59) we get,

$$\mathsf{TE}^\star_{X\to Y}(l) = \sup_{P_{X^l}} \mathsf{TE}_{X\to Y}(l). \quad \square \quad (61)$$

E. LIST OF SYMBOLS

Table 5 summarizes all primary symbols for quick reference.

TABLE 5. Comprehensive list of symbols used throughout the manuscript.

Symbol	Definition
$\mathbb{R}, \mathbb{Z}, \mathbb{N}$	Sets of real numbers, integers, and natural numbers, respectively.
$\mathcal{X}$	Subset of $\mathbb{R}^d$, the observation space.
$\mathbb{X}$	Stochastic process $(X_t)_{t\in\mathbb{Z}}$.
$\mathbb{P}$	Probability measure on $(\Omega, \mathcal{F})$.
$\mathcal{P}(\mathcal{X})$	Set of Borel probability measures on $\mathcal{X}$.
$\mathcal{P}_{\mathsf{ac}}(\mathcal{X})$	Subset of $\mathcal{P}(\mathcal{X})$ absolutely continuous w.r.t. Lebesgue measure.
$\mathbb{E}$	Expectation operator under $\mathbb{P}$.
$\mathsf{h}_{\mathsf{CE}}(P, Q)$	Cross-entropy between P and Q.
$\mathsf{h}(X)$	Differential entropy of $X \sim P$, i.e. $\mathsf{h}_{\mathsf{CE}}(P, P)$.
$\mathsf{D}_{\mathsf{KL}}(P\|\|Q)$	Kullback–Leibler divergence between P and Q.
$\mathsf{I}(X;Y)$	Mutual information between random variables X, Y.
$\mathsf{TE}_{X\to Y}(l)$	Transfer entropy from process X to Y with memory parameter l.
$\mathsf{I}(\mathbb{X}\to\mathbb{Y})$	Directed information from $\mathbb{X}$ to $\mathbb{Y}$.
W	Weight matrices in neural network layers.
b	Bias vectors in neural network layers.
α_{ij}	Attention weight from position i to j.
Q, K, V	Query, key, and value matrices in self-attention.
$\mathsf{Attn}(\cdot)$	Dot-product attention function (Definition 2).
$\mathsf{softmax}(\cdot)$	Softmax activation applied columnwise.
X_{pe}	Positional-encoded for input X.
$M_{[i,j]}$	Causal mask entry: can be 1 or $-\infty$ to include or ignore positions in the softmax operation.
$\mathcal{R}_{(i,j)}$	Attention coefficient weight from query at time i to key at time j.
$\mathcal{G}^{(d_i,d_o,l,v)}_{\mathsf{tf}}$	Class of transformers with input dim d_i, output dim d_o, length l, and width v (Definition 3).
$\mathcal{G}^{(d_x,1,l,v_y)}_{\mathsf{ctf}}$	Class of causal transformer architectures for single-process and joint-process estimation (Definition 3).
$\mathsf{D}_{Y_l\|Y^{l-1}\|\widetilde{Y}_l}$	
$\mathsf{D}_{Y_l\|Y^{l-1}X^l\|\widetilde{Y}_l}$	Variational KL objectives ((17)).
$\widehat{\mathsf{TE}}_{X\to Y}(D_n; l)$	TREET estimator objective for sample set D_n and memory l.
$\widehat{\mathsf{D}}_{Y_l\|Y^{l-1}\|\widetilde{Y}_l}$	
$\widehat{\mathsf{D}}_{Y_l\|Y^{l-1}X^l\|\widetilde{Y}_l}$	Empirical estimators of KL objectives ((19)).
C_{FF}	Feedforward channel capacity.
C_{FB}	Feedback channel capacity.

F. IMPLEMENTATION PARAMETERS

For the TE benchmark, channel capacity estimation model, and density estimation, we trained the optimization procedure with a limit of 200 epochs. We utilized a batch size of 1024, a learning rate of 8×10^{-3}, and the Adam optimizer [55]. The transformer architecture comprises one attention layer followed by a FF layer. The attention mechanism is

implemented with a single head, having a dimension of 32 neurons. The dimension of the FF layer is 64, featuring ELU activation [56]. Typically the order of TE l, was set to 30. The inputs sequence length was set to $l + 30$ to calculate 30 series in parallel. Each epoch generates 100K samples of $\mathbb{X}, \mathbb{Y}$ with corresponding random noise (default is uniform). For the cases of optimization, the NDG also employs a transformer structure with one attention layer and one FF layer, maintaining the same dimensional specifications. The learning rate for the NDG is set to 8×10^{-4}. It is trained at a rate of once every four epochs. For the Apnea dataset feature analysis, the learning rate was adjusted to 1×10^{-4}. Additionally, we configured the model to use 1 head with a dimension of 16 for the attention mechanism, and the FF layer dimension was set to 32, with ELU activation. All experiments were conducted on *Nvidia RTX 3090 and RTX Ada 6000 GPUs*.

REFERENCES

[1] T. Schreiber, "Measuring information transfer," *Phys. Rev. Lett.*, vol. 85, no. 2, pp. 461–464, Jul. 2000.
[2] J. M. Nichols, "Examining structural dynamics using information flow," *Probabilistic Eng. Mech.*, vol. 21, no. 4, pp. 420–433, Oct. 2006.
[3] P. Duan, F. Yang, T. Chen, and S. L. Shah, "Direct causality detection via the transfer entropy approach," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 6, pp. 2052–2066, Nov. 2013.
[4] T. Bossomaier, L. Barnett, M. Harré, and J. T. Lizier, "An introduction to transfer entropy," *Cambridge Int. Law J.*, pp. 65–95, 2016. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-43222-9_4
[5] B. Gourévitch and J. J. Eggermont, "Evaluating information transfer between auditory cortical neurons," *J. Neurophysiol.*, vol. 97, no. 3, pp. 2533–2543, Mar. 2007, doi: 10.1152/jn.01106.2006.
[6] S. Ito, M. E. Hansen, R. Heiland, A. Lumsdaine, A. M. Litke, and J. M. Beggs, "Extending transfer entropy improves identification of effective connectivity in a spiking cortical network model," *PLoS ONE*, vol. 6, no. 11, Nov. 2011, Art. no. e27431.
[7] M. D. Madulara, P. A. B. Francisco, S. Nawang, D. C. Arogancia, C. J. Cellucci, P. E. Rapp, and A. M. Albano, "EEG transfer entropy tracks changes in information transfer on the onset of vision," *Int. J. Mod. Phys., Conf. Ser.*, vol. 17, pp. 9–18, Jan. 2012, doi: 10.1142/s201019451200788x.
[8] M. Lungarella and O. Sporns, "Mapping information flow in sensorimotor networks," *PLoS Comput. Biol.*, vol. 2, no. 10, p. e144, Oct. 2006.
[9] G. Ver Steeg and A. Galstyan, "Information transfer in social media," in *Proc. 21st Int. Conf. World Wide Web*, Apr. 2012, pp. 509–518.
[10] S. Kim, S. Ku, W. Chang, and J. W. Song, "Predicting the direction of U.S. stock prices using effective transfer entropy and machine learning techniques," *IEEE Access*, vol. 8, pp. 111660–111682, 2020.
[11] P. Bonetti, A. Maria Metelli, and M. Restelli, "Causal feature selection via transfer entropy," 2023, *arXiv:2310.11059*.
[12] M. Rosenblatt, "Remarks on some nonparametric estimates of a density function," *Ann. Math. Statist.*, vol. 27, no. 3, pp. 832–837, Sep. 1956.
[13] L. F. Kozachenko and N. N. Leonenko, "Sample estimate of the entropy of a random vector," *Problemy Peredachi Informatsii*, vol. 23, no. 2, pp. 9–16, 1987.
[14] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 69, no. 6, Jun. 2004, Art. no. 066138.
[15] C. W. J. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica*, vol. 37, no. 3, pp. 424–438, Aug. 1969.
[16] L. Barnett and T. Bossomaier, "Transfer entropy as a log-likelihood ratio," *Phys. Rev. Lett.*, vol. 109, no. 13, Sep. 2012, Art. no. 138105.
[17] L. Barnett, A. B. Barrett, and A. K. Seth, "Granger causality and transfer entropy are equivalent for Gaussian variables," *Phys. Rev. Lett.*, vol. 103, no. 23, Dec. 2009, Art. no. 238701.
[18] J. Ma, "Estimating transfer entropy via copula entropy," 2019, *arXiv:1910.04375*.
[19] M. D. Donsker and S. R. S. Varadhan, "Asymptotic evaluation of certain Markov process expectations for large time. IV," *Commun. Pure Appl. Math.*, vol. 36, no. 2, pp. 183–212, Mar. 1983.
[20] M. I. Belghazi, A. Baratin, S. Rajeshwar, S. Ozair, Y. Bengio, A. Courville, and D. Hjelm, "Mutual information neural estimation," in *Proc. Int. Conf. Mach. Learn.*, Jul. 2018, pp. 531–540.
[21] D. Tsur, Z. Aharoni, Z. Goldfeld, and H. Permuter, "Neural estimation and optimization of directed information over continuous spaces," *IEEE Trans. Inf. Theory*, vol. 69, no. 8, pp. 4777–4798, Aug. 2023.
[22] J. Zhang, O. Simeone, Z. Cvetkovic, E. Abela, and M. Richardson, "ITENE: Intrinsic transfer entropy neural estimator," 2019, *arXiv:1912.07277*.
[23] S. Mukherjee, H. Asnani, and S. Kannan, "CCMI: Classifier based conditional mutual information estimation," in *Proc. Uncertainty Artif. Intell.*, 2020, pp. 1083–1093.
[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, Jun. 2017, pp. 5998–6008.
[25] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, May 2021, pp. 11106–11115.
[26] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2021, pp. 22419–22430.
[27] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin, "FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting," in *Proc. Int. Conf. Mach. Learn.*, Jan. 2022, pp. 27268–27286.
[28] J. Song and S. Ermon, "Understanding the limitations of variational mutual information estimators," 2019, *arXiv:1910.06222*.
[29] D. McAllester and K. Stratos, "Formal limitations on the measurement of mutual information," in *Proc. Int. Conf. Artif. Intell. Statist.*, Sep. 2018, pp. 875–884.
[30] K. Choi and S. Lee, "Combating the instability of mutual information-based losses via regularization," in *Proc. Uncertainty Artif. Intell.*, Jan. 2020, pp. 411–421.
[31] D. R. Rigney, A. L. Goldberger, W. C. Ocasio, Y. Ichimaru, G. B. Moody, and R. G. Mark, "Multi-channel physiological data: Description and analysis," in *Time Series Prediction: Forecasting the Future and Understanding the Past*, A. S. Weigend and N. A. Gershenfeld, Eds., Reading, MA, USA: Addison-Wesley, 1993, pp. 105–129.
[32] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, Jun. 2000.
[33] Y. Liu and S. Aviyente, "The relationship between transfer entropy and directed information," in *Proc. IEEE Stat. Signal Process. Workshop (SSP)*, Aug. 2012, pp. 73–76.
[34] J. L. Massey, "Causality, feedback and directed information," in *Proc. Int. Symp. Inf. Theory Applications*, Jan. 1990, pp. 303–305.
[35] G. Kramer, *Directed Information for Channels With Feedback*, vol. 11. Konstanz, Germany: Citeseer, 1998.
[36] C. Yun, S. Bhojanapalli, A. Singh Rawat, S. J. Reddi, and S. Kumar, "Are transformers universal approximators of sequence-to-sequence functions?" 2019, *arXiv:1912.10077*.
[37] S. Molavipour, H. Ghourchian, G. Bassi, and M. Skoglund, "Neural estimator of information for time-series data with dependency," *Entropy*, vol. 23, no. 6, p. 641, May 2021.
[38] L. D. Roland, "General formulation of Shannon's main theorem in information theory," *Amer. Math. Soc. Translations*, vol. 33, pp. 323–438, Jan. 1963.
[39] S. Tatikonda and S. K. Mitter, "The capacity of channels with feedback," *IEEE Trans. Inf. Theory*, vol. 55, no. 1, pp. 323–349, Jan. 2008.
[40] A. E. Gamal and Y. H. Kim, *Network Information Theory*. Cambridge, U.K.: Cambridge Univ. Press, 2011.
[41] D. Tsur, Z. Aharoni, Z. Goldfeld, and H. Permuter, "Data-driven optimization of directed information over discrete alphabets," *IEEE Trans. Inf. Theory*, vol. 70, no. 3, pp. 1652–1670, Mar. 2024.
[42] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed., Hoboken, NJ, USA: Wiley, 2006.

[43] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, Jan. 1989.
[44] S. Yang, A. Kavcic, and S. Tatikonda, "On the feedback capacity of power-constrained Gaussian noise channels with memory," *IEEE Trans. Inf. Theory*, vol. 53, no. 3, pp. 929–954, Mar. 2007.
[45] O. Sabag, V. Kostina, and B. Hassibi, "Feedback capacity of MIMO Gaussian channels," 2021, *arXiv:2106.01994*.
[46] S. Park and P. M. Pardalos, "Deep data density estimation through Donsker–Varadhan representation," *Ann. Math. Artif. Intell.*, vol. 93, no. 1, pp. 1–11, Feb. 2025.
[47] Z. Aharoni, D. Tsur, and H. H. Permuter, "Density estimation of processes with memory via Donsker vardhan," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2022, pp. 330–335.
[48] P. J. Brockwell and R. A. Davis, *Time Series: Theory and Methods*. Cham, Switzerland: Springer, 1991.
[49] C. M. Bishop, "Mixture density networks," Neural Comput. Res. Group, Aston University, Birmingham, U.K., Tech. Rep. NCRG/94/004, 1994.
[50] T. A. O'Brien, K. Kashinath, N. R. Cavanaugh, W. D. Collins, and J. P. O'Brien, "A fast and objective multidimensional kernel density estimation method: fastKDE," *Comput. Statist. Data Anal.*, vol. 101, pp. 148–160, Sep. 2016.
[51] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, Mar. 1960.
[52] D. Tsur and H. Permuter, "InfoMat: A tool for the analysis and visualization sequential information transfer," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2024, pp. 2098–2103.
[53] D. Tsur, B. Huleihel, and H. H. Permuter, "On rate distortion via constrained optimization of estimated mutual information," *IEEE Access*, vol. 12, pp. 137970–137987, 2024.
[54] L. Breiman, "The individual ergodic theorem of information theory," *Ann. Math. Statist.*, vol. 28, no. 3, pp. 809–811, Sep. 1957.
[55] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[56] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," 2015, *arXiv:1511.07289*.

OMER LUXEMBOURG (Student Member, IEEE) received the B.Sc. degree (cum laude) in computer engineering from the Ben-Gurion University of the Negev, Israel, in 2021, where he is currently pursuing the direct Ph.D. degree in electrical and computer engineering. His research interests include machine learning and information theory, primarily in the domain of time-series data.

DOR TSUR (Student Member, IEEE) received the B.Sc. (cum laude) and M.Sc. (summa cum laude) degrees in electrical and computer engineering from the Ben-Gurion University of the Negev, Israel, in 2020 and 2023, respectively, where he is currently pursuing the Ph.D. degree in electrical and computer engineering. His research interests include machine learning, information theory, and statistical signal processing.

HAIM PERMUTER (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees (summa cum laude) in electrical and computer engineering from the Ben-Gurion University of the Negev, Israel, in 1997 and 2003, respectively, and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, USA, in 2008. From 1997 to 2004, he was an Officer at the Research and Development Unit, Israeli Defense Forces. Since 2009, he has been with the Department of Electrical and Computer Engineering, Ben-Gurion University of the Negev, where he is currently a Professor and the Luck-Hille Chair in electrical engineering and also the Head for Communication, Cyber, and Information Track. He was a recipient of several awards, among them are the Fulbright Fellowship, Stanford Graduate Fellowship (SGF), Allon Fellowship, and U.S.–Israel Binational Science Foundation Bergmann Memorial Award. He served on the editorial board for IEEE TRANSACTIONS ON INFORMATION THEORY, from 2013 to 2016.