



Boosting Team Modeling through Tempo–Relational Representation Learning

Vincenzo Marco De Luca¹ · Giovanna Varni¹ · Andrea Passerini¹

Received: 10 July 2025 / Accepted: 14 April 2026
© The Author(s) 2026

Abstract

Team modeling remains a fundamental challenge at the intersection of Artificial Intelligence and Social Sciences. Although a variety of computational models have been proposed in the last two decades, most fail to sufficiently integrate Social Sciences insights, such as the critical role of temporal interactions in shaping team dynamics, and do not meet key practical requirements for real-world applications, including the ability to provide real-time, actionable recommendations to enhance team performance. To address these limitations, in this paper, we propose a novel tempo-relational neural architecture that jointly models interactions between team members and the evolution of team dynamics through temporal graphs. We additionally propose a multi-task extension of the architecture that learns shared *social embeddings* for team members enabling the simultaneous prediction of multiple team constructs (e.g., Emergent Leadership, Leadership Style, and Teamwork components). Experiments on two state-of-the-art team datasets show that our tempo-relational architecture outperforms temporal-only and relational-only approaches for team performance prediction, and that its multi-task extension substantially reduces training and inference time without loss of predictive performance. Finally, the integration of explainability techniques within the proposed architectures provides interpretable insights and actionable recommendations to support team improvement. These strengths make our approach particularly well suited for human-centered artificial intelligence applications, such as intelligent decision-support systems in high-stakes collaborative environments.

Keywords Automated team analysis · Temporal graph neural networks · Human-centered artificial intelligence

Introduction

The increasing integration of Artificial Intelligence (AI) systems into everyday life has accelerated the shift from traditional human-computer interaction (HCI), where a single user acts as the decision maker, towards human-machine teaming (HMT) approaches [1]. In HMT, groups of humans and AI agents collaborate, leveraging their complementary strengths to achieve shared goals [2]. This collaboration has amplified the need to design and develop Human-Centered

AI (HCAI) systems, characterized by transparency, trustworthiness, and collaborative intelligence [3]. More specifically, among such systems, Socially Intelligent Agents (SIA) [4] are specific agents equipped with social intelligence, namely the ability to navigate and succeed in social interactions by effectively engaging with others [5]. This capability aligns with the broader vision of theory of mind AI [6], which seeks to enable machines to infer, model, and reason about the mental states of others [7]. To achieve such a goal, accurate and principled computational models of team interactions are needed [8–10]. Although teams have been extensively studied within the Social Sciences, with particular focus on how members interact over time to give rise to emergent team processes (see, e.g., [11–17]), key insights about team dynamics still remain largely unincorporated in computational models. Indeed, most computational models developed in Social Signal Processing [18–21], Affective Computing [22–24], Reinforcement Learning [25], or Large Language models [26, 27] still overlook social science findings. Thus, effective computational team modeling

✉ Vincenzo Marco De Luca
vincenzomarco.deluca@unitn.it

Giovanna Varni
giovanna.varni@unitn.it

Andrea Passerini
andrea.passerini@unitn.it

¹ Department of Information Engineering and Computer Science, University of Trento, Trento, Italy

still remains an open challenge [28–30]. Only a few computational approaches, indeed, capture the temporal evolution of team members' behavior, while many assume independence of observations across time, thereby neglecting dynamic interdependencies. Moreover, team members' relations are often oversimplified or ignored at the architectural level [31], as they are reduced to concatenated or aggregated features [32]. Another critical issue is that distinct yet interdependent constructs such as teamwork and leadership tend to be modeled separately, resulting in redundant learning pipelines [33, 34], although recent works have begun to explore joint modeling approaches [35]. Finally, explainability remains limited: current models rarely provide transparent or actionable justifications for their predictions, despite growing demands for interpretability and accountability in HCAI [36, 37]. Thus, there is a mismatch between established theoretical principles and their operationalization in computational models. This paper is a first step in bringing insights from Social Sciences on temporal dynamics, team members' relations, construct interdependence, and explainability, into a single unified computational approach. To this end, we propose two architectures: the *Tempo-Relational Neural Network* (TRENN), and its multi-task extension, the *Multi-Task Tempo-Relational Neural Network* (MT-TRENN). From a methodological standpoint, these architectural choices not only close the main gaps identified in existing computational team modeling but also yield measurable scientific and practical advances. The main contributions of this work are as follows:

- A Tempo-Relational team computational approach (TRENN): an architecture inspired by Social Sciences insights that models both team dynamics as well as team members' relations. It outperforms alternative strategies in predicting several team constructs (e.g., Teamwork components and Emergent Leadership) in multiple datasets;
- A multi-task extension of Tempo-Relational approach (MT-TRENN): an architecture that extends TRENN to jointly predict multiple team constructs, by learning robust *social embeddings*, i.e., an embedding produced by a shared encoder in a multi-task setting. Such an embedding is designed to capture enough social information to generalize effectively across different team constructs. MT-TRENN offers the benefits of significantly reducing training and inference time, minimizing model storage requirements, while still producing accurate predictions;
- Two explainability modules: two modules that are seamlessly integrated in the architectures. One module generates detailed, interpretable insights in the form of factual explanations, while the other provides actionable suggestions through counterfactual explanations. The

validity of such insights and suggestions is supported by evidence drawn from data.

These advances address key limitations of existing computational approaches. As a result, TRENN and MT-TRENN are strong candidates for HCAI applications in high-stakes collaborative environments. In such settings, a SIA is expected to act as a trustworthy teammate, supporting critical decision-making processes. A compelling setting that illustrates the need for such a unified computational approach is the operating room (see Fig. 1), in which teamwork [38, 39] and leadership [40, 41] may critically affect surgical outcomes. During surgical procedures teams may commit errors due to lapses in coordination, miscommunication, or ineffective leadership. A SIA integrating our architectures can continuously monitor team behavior by analyzing multimodal input, such as paralinguistics, movement, and gaze. The agent predicts multiple team constructs both on-the-fly and offline. It also provides interpretable feedback on the drivers of suboptimal behaviors and supports informed debriefing procedures. Such decision-support capabilities are essential in healthcare, where timely insights can help prevent errors and enhance patient safety. Although the operating theatre serves as a vivid example of high-stakes, error-prone environments [42, 43], our unified computational framework is broadly applicable to any human-machine teaming system.

The remainder of this paper is organized as follows: Sec. “[Related Work](#)” reviews related work ranging from static team modeling approaches to temporal, relational or multi-task approaches; Sec. “[Background](#)” reports foundational team-related definitions in Social Sciences. Next, Sec. “[From Static to Multi-task Tempo Relational Team Modelling](#)” describes the evolutionary trajectory of neural paradigms for team modeling, culminating in our novel TRENN and MT-TRENN architectures, which are discussed in further detail in Sec. “[Implementing TRENN and MT-TRENN](#)”. Then, Sec. “[Experiments and Results](#)” reports experimental evaluation of TRENN, MT-TRENN, and the generation of factual and counterfactual explanations. Finally, Sec. “[Conclusion](#)” draws conclusions and outlines limitations and future work.

Related Work

Automated team modeling has gained increasing attention over the past two decades, driven by the need to develop systems capable of understanding, modeling, and automatically interpreting human social interactions. Progress in this field, however, has been constrained by the technical difficulties of collecting multimodal data in multiparty scenarios as well as by the scarcity of publicly available datasets

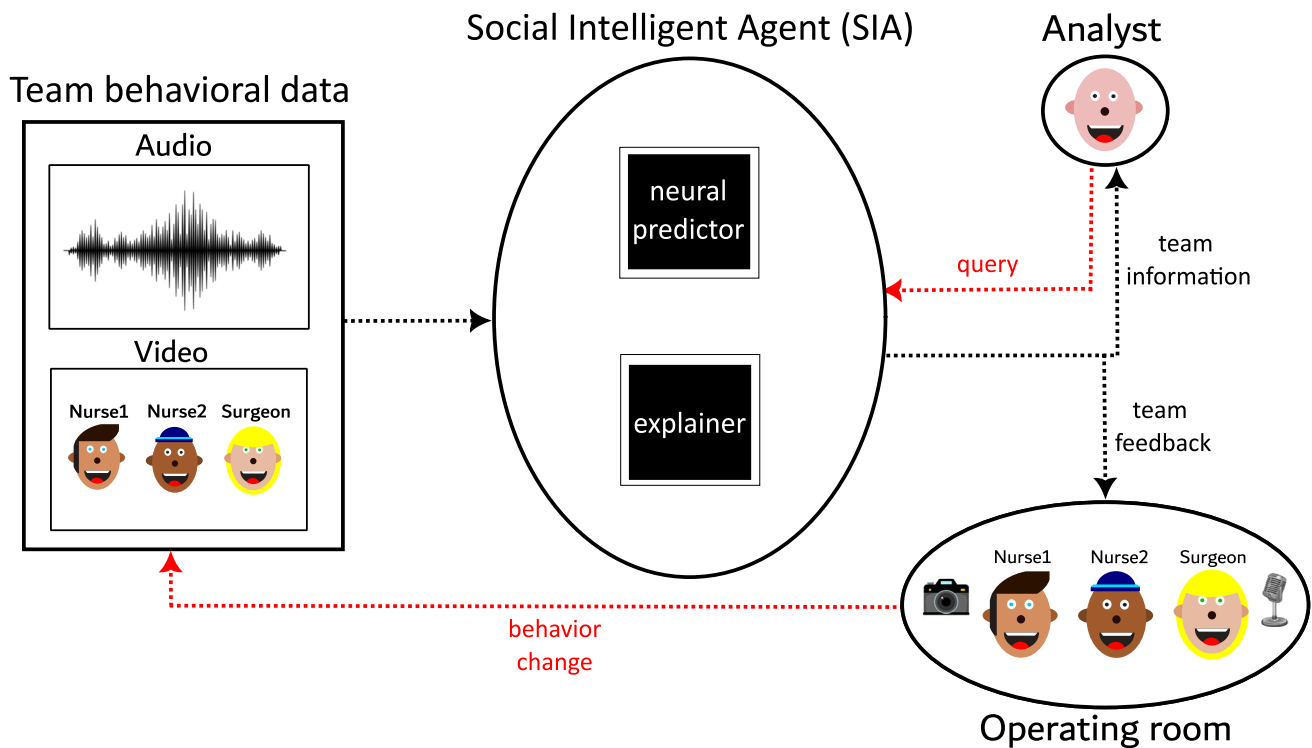


Fig. 1 The SIA takes as input the team's behavioral data (i.e., audio and video) directly captured in the operating room using sensors such as microphones and cameras. The SIA is composed of two modules: a neural predictor, used to infer team-related information, and an explainer, which generates feedback for the team. These outputs can

be provided either offline or on the fly, either to an analyst or directly to the team in the operating room. The red arrows indicate either the query performed by the analyst or the behavioral changes occurring in the scene and captured by the sensing devices

(e.g., [44–49]). In this section, we review previous work on computational team modeling according to the aspects chosen by the authors to model team behavior: (i) static team modeling; (ii) temporal team modeling; (iii) relational team modeling; and (iv) multi-task and multi-construct team modeling.

Static Team Modeling

Static team models rely on individual-level or aggregated multimodal features (e.g., speech, video) to model team-level constructs without explicitly modeling temporal evolution or interactions among team members. Within this paradigm, some of these computational approaches extract behavioral features from one or two modalities and then pool them across team members. The pooled features, computed as a simple average or concatenation [50, 51], form a team-based feature vector that serves as input to machine learning models. The primary objective of these studies is to investigate which features are most informative to predict a specific team construct. One of the earliest works in static team modeling was proposed by Hung and colleagues in 2010 [33]. They fed hand-crafted audio and visual features into a naive Bayes classifier and a support vector machine

to estimate team cohesion. Later studies have also predicted team cohesion, introducing pairwise and team-level features derived from the alignment of para-linguistic speech behavior and fed into a Gaussian mixture model and a kernel density estimator [52]. Related works have explored leadership emergence and its connection to the team members' body pose patterns using support vector machines [53]. In a similar static fashion, acoustic features and linguistic features extracted from conversation transcripts have been used as input for gradient boosting and extra-trees [34, 54] to predict team performance in task-based interactions. Moreover, engagement has been modeled through multimodal features (i.e., speech and visual features) via feed-forward neural networks (FFNN) [55]. Despite the effort of interdisciplinary research, most computational models still do not encode time and relations in their analysis, as shown in recent surveys (e.g., [56]).

TRENN addresses these limitations by explicitly modeling temporal dynamics in the predictive architecture.

Temporal Team Modeling

Temporal team models incorporate the evolution of individual behavior over time but treat team members

independently, without modeling interpersonal relations or cross-team construct dependencies. Social sciences have long demonstrated that team communication patterns can be described as temporal graphs [57]. However, only a few studies explicitly integrate the temporal information into the computational modeling [58]. Early work introducing temporal information proposed an unsupervised approach to model parallel episodes [59] that captures temporally proximal events and overlapping actions, but encoding time only at the individual feature level. An alternative line of research draws from dynamical systems theory [60, 61], which provides a theoretical foundation to understand team behavior as evolving patterns over time. Building on this foundation, more recent works adopt multimodal data-driven sequence-to-sequence modeling strategies [62], including Hidden Markov Models [63], Conditional Restricted Boltzmann Machines [64], Long-Short Term Memory (LSTM) [65], and Fusion Transformers [66]. To further leverage hierarchical temporal patterns, recurrent architectures such as HPNet [67] have been proposed for modeling multi-scale sequences in spatio-temporal data like video frames. Other approaches model discrete symbolic actions through Bayesian approaches to predict team mental states [68], or leverage Multi-Agent Imitation Learning to learn a generative model of team behavior [69] validated for intervention to enhance team performance [70]. Similarly, architectures such as Spatial-TeamFormer [71] and GroupFormer [72] model temporal patterns in multi-agent settings, relying on visual, spatial, or trajectory data. Despite these advances, most team constructs influenced by interactions among members are still modeled without capturing the relational structure [73].

TReNN addresses the limitations of temporal models by jointly modeling temporal and relational information.

Relational Team Modeling

Relational team models incorporate information about how team members influence one another, but ignore the team dynamics over time and the interdependencies among team constructs. Early proposals integrate relations only at the feature level by combining individual-level and team-level features, but these approaches apply Convolutional or Recurrent Neural Networks independently to each team member. Scores as team-member coefficients in the cost function have been also proposed during the optimization process applied to Random Forest outcomes [74]. Attempts to integrate relations at the modeling level were based on network analysis [75, 76], and complex systems [77–79]. The first work [58] adopting Graph Neural Networks (GNNs) [80, 81] uses acoustic and verbal as individual features and speaking turns as edges modeled through Graph

Convolutional Networks (GCNs) [82] to predict only team performances. Sharma and colleagues [83] proposed hierarchical GNNs, fed with visual features, to predict the dominance score and ranking of team members. In recent years, Temporal GNNs (TGNNs) have been proposed to solve tasks such as behavior forecasting [84], impression prediction [85], and human interaction classification [86]. A recent work proposes Cross-Attention [87] between sender and receiver to model interaction in engagement prediction [88]. Moreover, time modeling through Imitation Learning and hierarchical structure has been proposed to predict individual behavior and team behavior [89], but still lacks an explicit modeling of relations. Hypergraphs for recognizing social interaction [90] and GNNs combined with Multimodal Large Language Models [91] are other additional works that leverage GNNs for teams, but without assessing the quality and behavior of the team. Some recent probabilistic frameworks take a different approach to modeling team relations. For example, Bayesian Team Cognition [9] infers latent cognitive states of team members from observed decisions and communications. While this approach provides interpretable insights into teammate beliefs and intentions, it does not produce continuous multi-task behavioral constructs. To the best of our knowledge, no work addresses team process prediction using TGNN architectures yet.

As for the purely temporal approaches, TReNN overcomes the limitations of purely relational approaches thanks to its joint tempo-relational modeling capabilities.

Multi-task Team Modeling

Multi-task team models aim to predict multiple team constructs from individual team member features. They leverage dependencies among constructs by projecting member features into a shared embedding space. This space is later specialized through task-specific fully connected networks. Despite this advance in modeling multiple team constructs, they still disregard the temporal evolution of the team and the importance of modeling relations in the team. Approaches mentioned so far develop separate models to predict different team constructs or different dimensions of the same construct. Consequently, each dimension requires training a model from scratch. This design results in increased spatial and temporal complexity. The number of parameters grows with the number of tasks, and training multiple independent models increases training time. Moreover, these approaches fail to address real-world application requirements. In practice, agents must jointly reason over multiple interlaced team constructs to effectively collaborate with human teams [92–94]. Beyan et al. [95] introduced multi-task learning (MTL) to predict leadership styles in

Table 1 Definitions of key team-related concepts relevant to computational team modeling

Term	Definition
<i>Team</i>	A <i>team</i> is a social unit of more than two individuals with specified roles who interact adaptively, interdependently, and dynamically toward a common and valued goal [99–101].
<i>Team construct</i>	A <i>team construct</i> is any measurable property capturing a specific behavioral, cognitive, affective, or relational aspect of team functioning (e.g., teamwork, leadership style).
<i>Team interaction</i>	<i>Team interaction</i> refers to the exchange of information, behaviors, and affect among team members as they coordinate, communicate, and collaborate to achieve shared goals [102].
<i>Team dynamics</i>	<i>Team dynamics</i> concerns the evolution of team behaviors and member interactions over time, leading to the emergence of complex collective social phenomena [103].

Table 2 Core behavioral components of the Big Five in Teamwork framework

Main Component	Definition
<i>Adaptability (A)</i>	The ability of a team to adjust roles, strategies, and resource allocation in response to environmental changes [104].
<i>Back-up behavior (BB)</i>	The ability of team members to anticipate the needs of others and provide timely support when required [104].
<i>Mutual performance monitoring (MPM)</i>	The ability to comprehend the team environment and continuously monitor team members' performance to ensure effective coordination [104].
<i>Team leadership (TL)</i>	The capacity to direct, coordinate, and guide the activities of team members toward shared objectives [104].
<i>Team orientation (TO)</i>	The capacity of team members to prioritize collective goals and outcomes over individual interests [104].

Table 3 Coordinating mechanisms supporting the main teamwork components according to the Big Five in Teamwork framework

Coordinating Mechanism	Definition
<i>Closed-loop communication (CC)</i>	The process by which team members explicitly direct, acknowledge, and confirm messages to ensure accurate receipt and shared understanding [104].
<i>Mutual trust (MT)</i>	The shared belief among team members that others will reliably fulfill their roles and act in the best interest of the team [104].
<i>Shared mental models (SMM)</i>	The common and overlapping understanding among team members regarding tasks, roles, and interaction patterns that supports effective coordination and anticipation of teammates' actions [104].

team interaction. Such an approach, however, completely disregards dynamics and relations among team members. Ghosh et al. [96] propose an approach to predict cohesion through MTL, whereas Maman et al. [97] present two architectures to jointly predict social and task cohesion and the valence of emotion.

Other works explore the possibility of generating social embeddings that can be reused for multiple tasks, but restrict this reuse to fine-tuning strategies [98]. As a result, additional parameters are introduced and task-specific specialization remains necessary for doing inference on other tasks.

MT-TRENN improves over purely static multi-task team modeling approaches by incorporating multi-task learning functionalities in the tempo-relational architecture developed with TRENN.

Background

From a Social Sciences perspective, team effectiveness emerges from the interplay of relational structures and situational forces. Relational Models Theory [105, 106] offers a pathway to understand these relational structures. It describes how social behavior and the quality of the

interactions are guided by shared relational expectations, such as cooperation among peers, hierarchical coordination, or reciprocity in exchanges. These expectations, in turn, shape collaboration, trust, and the emergence of leadership. Field Theory [107] complements this view by conceptualizing team behavior as a function of individual characteristics operating within a dynamic social field shaped by group norms, shared goals, and patterns of interpersonal influence. Together, these perspectives ground teamwork processes in both relational structures and situational context. To systematically analyze the team framework, we first introduce the fundamental concepts that underpin team theory. Table 1 presents clear definitions of the fundamental team-related concepts serving as a reference point for the frameworks and mechanisms discussed in this section. Building on this foundation, the Big Five in Teamwork (BFT) framework [104] identifies five behavioral components, defined in Table 2, supported by three coordinating mechanisms, listed in Table 3, which collectively map individual and collective behaviors into a unified measure of Team Effectiveness (TE).

Definition 1 Teamwork (TW) is the set of tied thoughts, actions, and feelings of each team member that are needed to function as a team and that combine to facilitate coordinated,

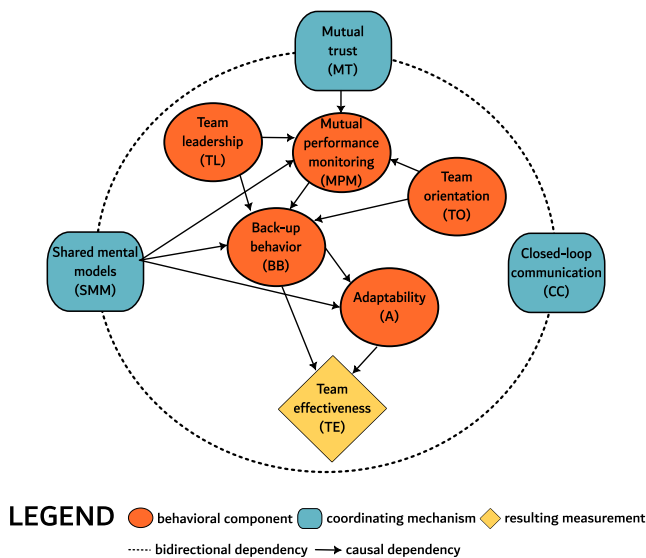


Fig. 2 The BFT framework. It includes five core components (in orange), and three additional components (in blue), representing coordinating mechanisms. All components collectively contribute to Team Effectiveness (TE), which is represented in the yellow node. Arrows indicate direct influences between connected components

adaptive performance and task objectives resulting in value-added outcomes [104, 108, 109].

BFT conceptualizes TW as a set of interrelated mechanisms, as illustrated in Fig. 2: Mutual Trust (MT) is affected by the communication among team members (CC), and influences the presence of Shared Mental Models (SMM) among team members. The Mutual Performance Monitoring (MPM) is guaranteed by the Team Leadership (TL), the Orientation of the Team (TO), and MT. Proper Back-up Behavior (BB) requires effective TL, SMM, MPM, and TO, while Adaptability (A) requires proper BB and SMM.

BFT presents ten propositions that define how these team components contribute to TE. In particular, TL influences TE by setting performance expectations, including performance monitoring and backup behavior. MPM affects TE through effective BB and requires adequate SMM and MT among team members. BB directly contributes to TE by ensuring task completion, although its effects are mediated by the team's ability to adapt to internal and external changes. Effective BB further requires adequate SMM and MPM. A directly affects TE and depends on SMM, MPM, and BB. Finally, TO influences TE through members' willingness to mutually monitor performance and accept support from others through BB.

These interdependent mechanisms illustrate how coordination, trust, and monitoring naturally give rise to patterns of influence within a team, implying that certain team members influence team processes more strongly due to their behavior, relational positioning, and interactions. This

perspective provides a smooth conceptual bridge from the competencies captured by BFT to the study of leadership dynamics.

Definition 2 Emergent Leadership (EL) is an emergent state that characterizes a team member who naturally appears as the leader during social interaction, without any formal leadership role, and whose power comes from peers in the team, rather than from an established authority [110].

Given the pivotal role of leadership, Field Theory and Relational Models Theory jointly suggest that different Leadership Styles can foster or hinder key team constructs including communication, coordination, trust-building, adaptability, and EL [111]. These effects ultimately impact overall team performance and decision-making [112–115].

Definition 3 The Leadership Style (LS) reflects how a team member, i.e., the leader, interacts with others when exhibiting leadership [116].

The interplay of TW, EL, and LS encapsulates fundamental aspects of team functioning, shaping how teams coordinate, adapt, and perform over time [117–120]. Developing computational team models that predict these team constructs over time, and that analyze how individual member and interactions influence them, provides a valuable asset for both advancing Social Sciences theories and designing innovative human-machine teaming applications.

From Static to Multi-task Tempo Relational Team Modeling

This section first outlines the evolutionary trajectory of neural paradigms for team modeling. We first formalize the existing approaches into a taxonomy of static, temporal, and relational models (Sec. “Static and Temporal/relational Modeling”). Then, we present our proposed TRENN approach that jointly captures temporal and relational information, and its extension MT-TRENN designed for the simultaneous prediction of multiple interleaved team constructs through a MTL objective (Sec. “Tempo-relational and Multi-task Modeling”). Implementation details of TRENN and MT-TRENN are reported in Sec. “Implementing TRENN and MT-TRENN”.

We formalize all architectures within an encoder–decoder abstraction for team modeling. The encoder produces a social embedding of team members, which is then decoded into predictions of team constructs. The encoder $f(\cdot)$ maps an input vector $\mathbf{x} \in \mathbb{R}^n$, representing n individual behavioral features per team member, into a hidden representation $\mathbf{h} \in \mathbb{R}^d$. The decoder $g(\cdot)$ maps this hidden representation

$h \in \mathbb{R}^d$ into an output vector $y \in \mathbb{R}^m$, representing $m \geq 1$ team construct scores.

Definition 4 The *social embedding* $h \in \mathbb{R}^d$ is the hidden representation generated by the encoder $f(\cdot)$. It is designed to capture individual and relational information from input data which is informative enough for the decoder $g(\cdot)$ to predict one or more team constructs y for each team member.

Table 4 summarizes the mathematical notation used throughout this section.

Static and Temporal/relational Modeling

Existing approaches range from static, independent representations of team members to models that incorporate either temporal dynamics or interpersonal interactions.

Based on the assumption that individual team member behaviors suffice to predict team constructs, the simplest encoder is *Static Neural Network* (SNN) [33, 34, 52, 54] implemented as FFNN, as shown in Fig. 3a, that models a *static team member*.

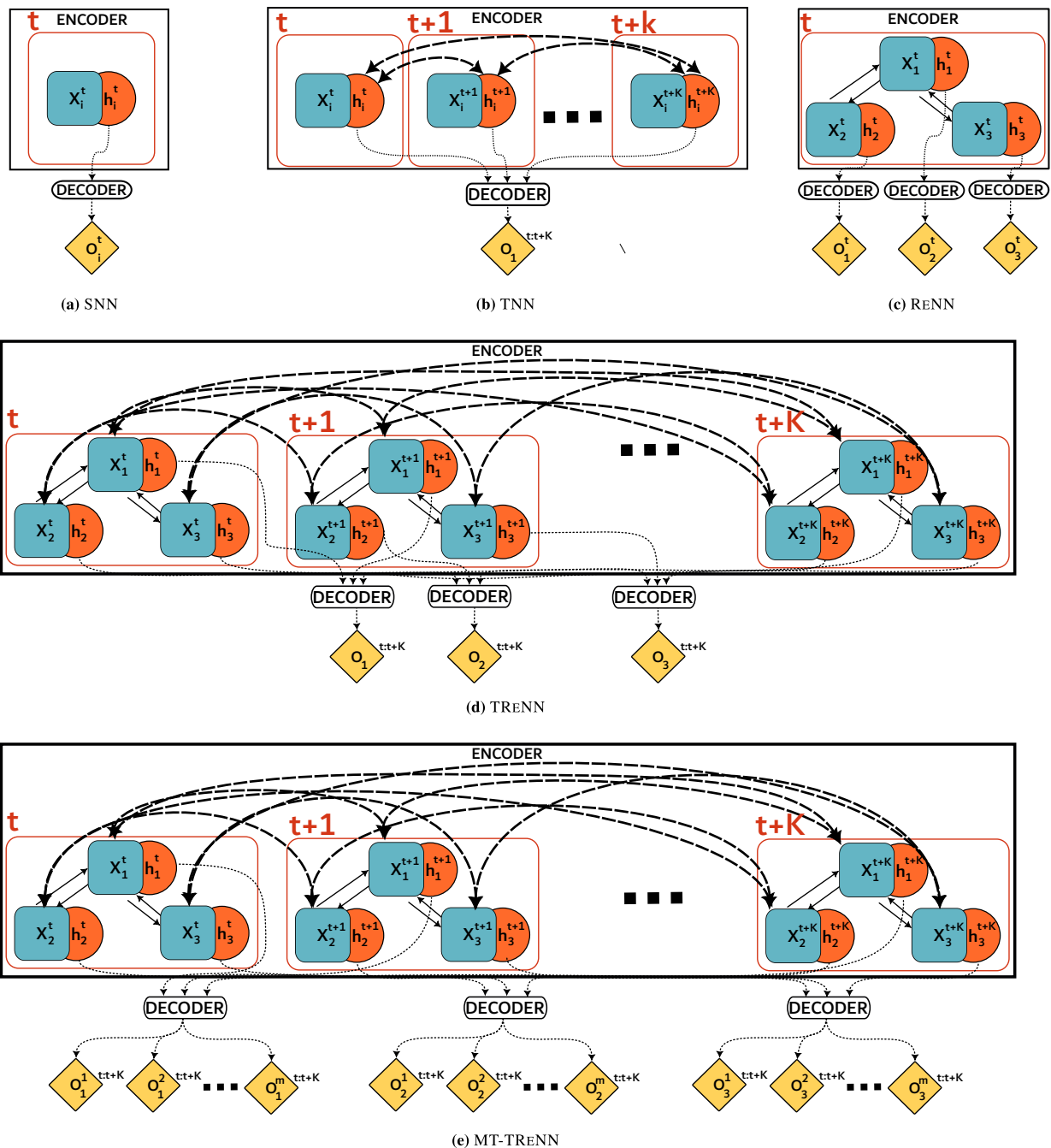
Definition 5 A *static team member* is defined as a pair $\{x, y\}$, where $x \in \mathbb{R}^d$ are input features, and $y \in \mathbb{R}$ is the output score.

Definition 6 A SNN uses its *encoder* f_{snn} to map a *static team member* into their corresponding embedding h (1) and then uses its *decoder* g to map this embedding into the output space $y \in \mathbb{R}$ (2).

$$h = f_{\text{snn}}(x) \quad (1)$$

Table 4 Summary of the notation used for static, temporal, relational, and multi-task team modeling

Symbol	Description
<i>Indices and dimensions</i>	
v, u, i	Team member (node) indices
t	Time index
K	Temporal window length
$t : t + K$	Temporal interval from t to $t + K$
k	Task index in multi-task learning
n	Number of team members ($ \mathcal{V} $)
d	Feature or embedding dimensionality
m	Number of team constructs (tasks)
<i>Feature representations</i>	
x	Feature vector of a single team member
x_v^t	Features of team member v at time t
X	Matrix of all team members' features
X^t	Feature matrix at time t
<i>Embeddings and outputs</i>	
h_v	Static social embedding of team member v
h_v^t	Social embedding of member v at time t
$h_v^{t:t+K}$	Temporal embedding over interval $[t, t + K]$
y_v	Output prediction for team member v
y_v^k	Prediction of task k for member v
Y	Matrix of outputs for all members and tasks
<i>Graphs and neighborhoods</i>	
$\mathcal{V}$	Set of team members (nodes)
$\mathcal{E}$	Set of interactions (edges)
$\mathcal{N}(v)$	Neighborhood of node v
$G^t = (\mathcal{V}, \mathcal{E}, X^t)$	Team graph at time t
$G = \{G^t, \dots, G^{t+K}\}$	Dynamic team as temporal graph sequence
<i>Functions</i>	
$f(\cdot)$	Encoder function
$g(\cdot)$	Decoder function
$g_{\text{mt}}(\cdot)$	Multi-task decoder



$$y = g(h)$$

(2)

SNN models each team member in isolation, ignoring temporal and relational context as well as multiple team constructs. Consequently, SNN fails to capture the behavioral dynamics essential in team settings, resulting in an oversimplified representation of complex social interactions [51]. To overcome these limitations, researchers extended SNN

by modeling individual behavior over time and leveraging principles from dynamical systems [60, 61] or adopting Temporal Neural Networks (TNN) implemented as Recurrent Neural Networks, LSTM [121], Gated-Recurrent Unit [122], or Multi-Head Attention (MHA) [123].

TNN models a *dynamic team member* as a time-series of *static team member* snapshots, as also explored in other works [62–66, 127] and illustrated in Fig. 3b.

Fig. 3 Neural Network-based architectures to model teams. Rectangular boxes with red borders are *encoders*, while round-shaped boxes with black borders are *decoders*. Any encoder contains the input data X_j^{t+i} , and its corresponding hidden representation h_j^{t+i} , where the subscript j refers to the team member and the superscript $t+i$ refers to the timestep. Any decoder receives as input the hidden representation h_j^{t+i} , and maps it into the output space $o_j^{k,t+i}$, where the subscript j refers to the team member and the superscript k refers to the predicted team task, and $t+i$ the timestep. (a) SNN takes as input team member information at a specific time-step X_i^t , and maps it into its corresponding hidden representation h_i^t . Then, the decoder from the hidden representation predicts the output only based on team member and instantaneous information; (b) TNN: the encoder takes as input a sequence of team member information $(X_i^t, X_i^{t+1}, \dots, X_i^{t+K})$; the temporal evolution of team member information is leveraged to produce the hidden representation $(h_i^t, h_i^{t+1}, \dots, h_i^{t+K})$. Next, the decoder from the hidden representation gets the final prediction $o_i^{t,t+K}$ that is based on the temporal evolution of the team member; (c) RENNN: the encoder takes as input the team member information as well as the team member interactions at a specific time step. Let us assume a team composed of three people: X_1^t, X_2^t, X_3^t , the encoder produces the corresponding hidden representation for each of them: h_1^t, h_2^t, h_3^t , while the decoder receives from such a hidden representation outputs the prediction corresponding to each team member o_1^t, o_2^t, o_3^t ; (d) TRENN: the encoder takes as input the team member information as well as the team members interactions over multiple time-steps. Let us assume a team composed of three people: $X_1^{t+i}, X_2^{t+i}, X_3^{t+i}, \forall i \in [0, K]$, the encoder produces the corresponding hidden representation for each of them: $h_1^{t+i}, h_2^{t+i}, h_3^{t+i}$, while the decoder from such a hidden representation gets the prediction corresponding to each team member $o_1^{t+i}, o_2^{t+i}, o_3^{t+i}$; (e) MT-TRENN: the encoder works exactly as in TRENN, while the decoder learns multiple team-tasks: given the hidden representation, it generates, for each team member, the prediction of multiple team-tasks $o_1^{1,t+i}, o_2^{2,t+i}, \dots, o_3^{m,t+i}$

Definition 7 A *dynamic team member* is defined as a pair $\{X, y\}$, where $X = (x^1, x^2, \dots, x^t)$ is a sequence of input features for a specific team member over t consecutive time steps and $y \in \mathbb{R}^m$, where $m = 1$ is the output score referring to a team construct for the team member over the entire sequence.

Definition 8 A TNN uses its *encoder* f_{tnn} to map a sequence of input features $X \in R^{t \times d}$ into an embedding $h \in R^m$, and, similarly to (2), uses its *decoder* $g(h)$ to produce a single output score $y \in \mathbb{R}^m$, where $m = 1$ for each team member.

$$(h^t, \dots, h^{t+K}) = f_{tnn}(\{x^t, x^{t+1}, \dots, x^{t+K}\}) \quad (3)$$

However, TNN models team members in isolation, missing the interaction information among team members, and ignores the possibility of jointly modeling multiple team constructs. Since the importance of interactions in teams has been widely studied [128, 129], researchers have proposed team modeling approaches grounded in Graph Theory and Network Science [75, 76, 99, 100, 130], Complex Systems [77–79], or GNN [58, 83]. We define

these neural approaches as Relational Neural Networks (RENN), as represented in Fig. 3c, and can be implemented as GCN [82], Graph Attention Network [131], or Graph Isomorphism Network [132]. They, indeed, can learn *Social Embeddings* incorporating individual and interaction behavior via message passing strategies over the network representation of the (static) team, and open the road for relevant xAI-based applications [133, 134], even though they have been largely ignored in the team modeling literature.

Definition 9 A *static team* is a graph $G = (\mathcal{V}, \mathcal{E}, X, Y)$, where $\mathcal{V}$ is the set of team members, $\mathcal{E}$ is the set of interactions, $X \in \mathbb{R}^{n \times d}$ are input features for each of the $n = |\mathcal{V}|$ team members, and $Y \in \mathbb{R}^{n \times m}$ are the corresponding outputs.

Definition 10 A RENNN uses its *encoder* f_{rennn} to encode each team member v as h_v and then, similarly to (2), uses its *decoder* $g(h_v)$ to map the embedding into the corresponding output score for the team member $y_v \in \mathbb{R}^m$, where $m = 1$:

$$h_v = f_{rennn}(x_u \mid u \in \{v \cup \mathcal{N}(v)\}) \quad (4)$$

where x_u represents the features of team member u , while $\mathcal{N}(v)$ denotes the set of neighbors of v . Notice that this definition assumes a single processing step for simplicity. In standard RENNN, this operation is recursively repeated to capture multi-hop relational dependencies, thereby enriching the *social embedding* of each team member. RENNN models interactions among team members.

RENN fails to encode dynamics, a crucial aspect in modeling team behaviour [13]. These limitations have motivated researchers to combine knowledge from dynamical systems and network science to model teams as adaptive complex systems [79]. This approach, however, struggles to process rich feature-based information, an area where deep learning architectures are particularly effective.

Tempo-relational and Multi-task Modeling

To overcome the limitations of the previously mentioned architectures, we propose a novel approach for team modeling that leverages TRENN that extends the view of teams as adaptive complex systems [79] by offering a data-driven, scalable approach that jointly accounts for temporal dynamics and interaction information, as represented in Fig. 3d. TRENN can be implemented using EvolveGCN [135], DySAT [136], or, in general, following the framework TGN [137].

Table 5 Comparison of neural network-based architectures for team modeling, organized by (i) their modeling capabilities (i.e., whether they capture relational information, temporal dynamics, or multiple tasks), and (ii) their input–output mapping. SNN cannot model relations, temporal dynamics, or multi-task predictions due to the structure of its input and output: the input $x \in \mathbb{R}^d$ represents a static team member (Definition 5), which is mapped into a single team construct $y \in \mathbb{R}$ using a SNN (Definition 6), as illustrated in Fig. 3a. TNN models temporal information, but cannot capture relations, since its input $x \in \mathbb{R}^{t \times d}$ only contains a temporal sequence for a single dynamic team member (Definition 5). This input is mapped into a single team construct $y \in \mathbb{R}$ using a TNN (Definition 8; Fig. 3b). ReNN captures relational structure because its input is a static team represented as a graph $G = (\mathbb{V}, \mathbb{E}, X)$ (Definition 9). The model outputs a prediction for each team member, $y \in \mathbb{R}^{|\mathbb{V}|}$, using a ReNN (Definition 8; Fig. 3c). However, it cannot capture temporal dynamics, since its information is limited to static features and relations. TReNN captures both relational and temporal aspects by modeling a dynamic team (Definition 11) as a sequence of temporal graphs $G = \{G^t, \dots, G^{t+k}\}$. It produces a single prediction for each team member over the entire temporal sequence, $y \in \mathbb{R}^{|\mathbb{V}|}$. Finally, MT-TReNN extends TReNN by enabling multi-task prediction. It uses the same dynamic-team input, but outputs multiple constructs per member, $y \in \mathbb{R}^{|\mathbb{V}| \times m}$, with m denoting the number of tasks

Architecture	Type of modeling			Input-Output mapping	
	Relational	Temporal	Multi-task	Input	Output
SNN [33, 34, 50–56]	×	×	×	$x \in \mathbb{R}^d$	$y \in \mathbb{R}$
TNN [58, 61, 63–66, 68–70, 73]	×	✓	×	$x \in \mathbb{R}^{t \times d}$	$y \in \mathbb{R}$
ReNN [58, 74, 83, 90, 91, 124–126]	✓	×	×	$G = (\mathbb{V}, \mathbb{E}, X)$	$y \in \mathbb{R}^{ \mathbb{V} }$
TReNN	✓	✓	×	$G = \{G^t, \dots, G^{t+k}\}$	$y \in \mathbb{R}^{ \mathbb{V} }$
MT-TReNN	✓	✓	✓	$G = \{G^t, \dots, G^{t+k}\}$	$y \in \mathbb{R}^{ \mathbb{V} \times m}$

Legend: ✓ = supported; × = not supported

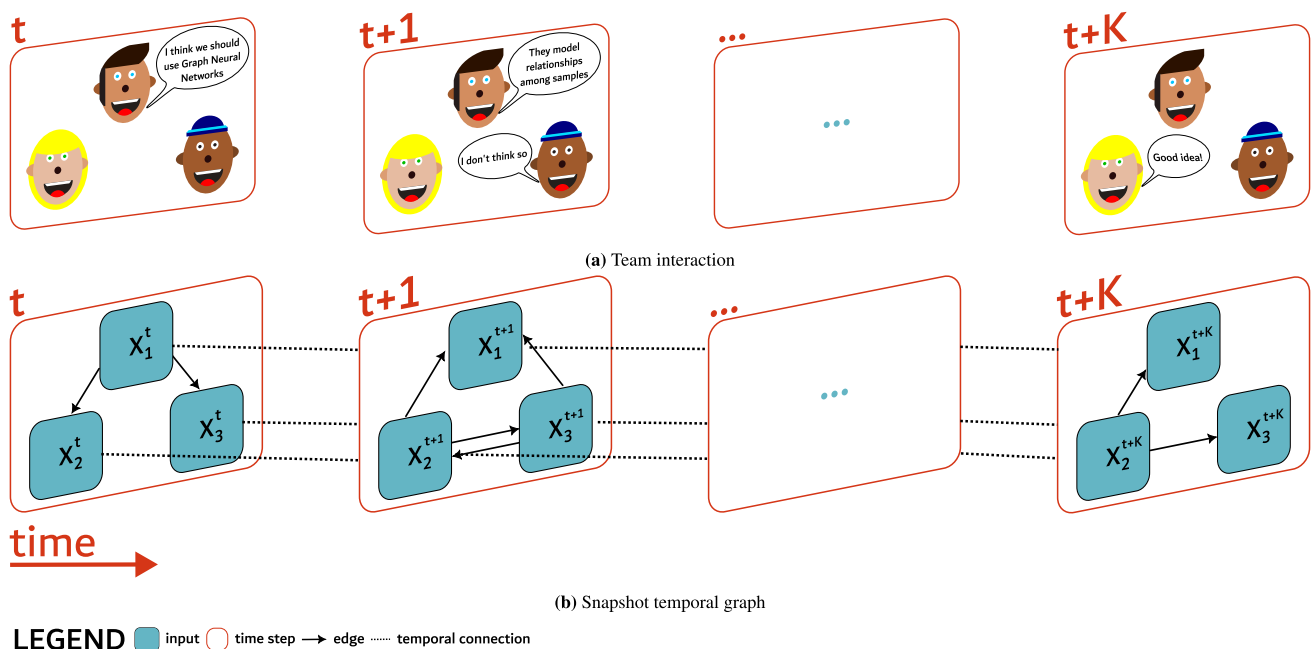
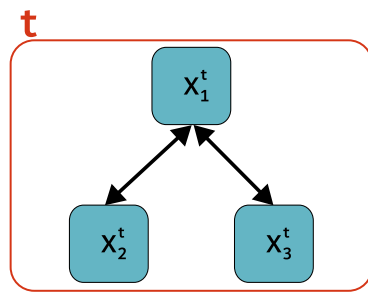
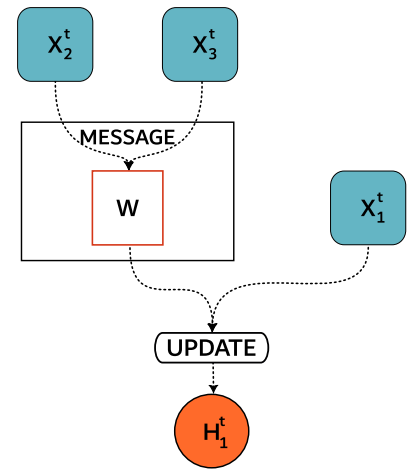


Fig. 4 (a) A team interaction consists of a set of team members continuously interacting, exchanging verbal and nonverbal messages. Let us suppose that three team members are discussing the use of Graph Neural Networks to model team behavior. Initially (at time t), the team member with black hair tells the other team members that they should use Graph Neural Networks, later (at time $t+1$), the team member with the hat expresses his doubt about this idea. After some steps (at time

$t+K$), finally, Lisa also states that she agrees with the others. This team interaction is converted by the temporal graph encoder into a snapshot temporal graph, as represented in (b). Each node X_i^t represents the features of a team member (e.g., paralinguistic features), while each edge connecting a pair of nodes represents a one-to-one interaction (e.g., speaker-broadcaster, where each team member who speaks has an edge connecting themselves to all the other team members)



(a) Temporal graph snapshot



(b) Message Passing procedure for a node

LEGEND ■ input ● hidden representation learnable tensor math operator → tensor flow

Fig. 5 The MPNNs' steps to map topological structure into a single tensor. **(a)** Let us consider a snapshot graph at time t with three nodes (X_1^t, X_2^t, X_3^t) and two bidirectional edges. This topology encloses both individual node information (i.e., features) and topological information (i.e., edges). **(b)** Generation of the hidden representation (i.e., H_1^t) for the first node (i.e., X_1^t). Since the node is connected by an

edge to two neighboring nodes (i.e., X_2^t, X_3^t), the individual features of these two nodes are combined into a single message through the *Message* module. The resulting message is combined with the original representation of the first node (i.e., X_1^t) through the *Update* module, resulting in the final hidden representation of the first node H_1^t

Definition 11 A *dynamic team* is defined as $T = (G, Y)$, where $G = \{G^t, \dots, G^{t+K}\}$ is a sequence of k time-indexed graphs $G^j = (\mathcal{V}^j, \mathcal{E}^j, X^j)$, and $Y \in \mathbb{R}^{n \times m}$ are the corresponding outputs.

Definition 12 Given a *dynamic team* as input, a TRENN uses its *encoder* f_{trenn} to encode the temporal evolution of each team member v into the embedding $h_v^{t:t+K}$, and uses its *decoder*, similarly to (2), to map this embedding into the output space $y_v \in \mathbb{R}^m$, where $m = 1$.

$$(h_v^t, \dots, h_v^{t+K}) = f_{\text{trenn}}(x_u^{t+i} | u^{t+i} \in \{v^{t+1} \cup \mathcal{N}(v^{t+i})\}, \forall i \in \{0, 1, \dots, K\}) \quad (5)$$

Owing to the inherently interleaved nature of team constructs, we extend TRENN with an MTL objective that enables the model to jointly represent and optimize multiple team constructs. This paradigm captures cross-task information and shared latent patterns (e.g., between emergent leadership and teamwork), leading to more robust and generalizable models of team functioning.

Definition 13 A MT-TRENN, similarly to (5), uses its encoder f_{trenn} to map each team member v of a *dynamic team* into its corresponding embedding h_v , and uses its $m \geq 2$ *multi-task decoders* g_{mt} to map the embedding into

multiple team constructs $Y_v \in \mathbb{R}^m$ for each team member in a team.

MT-TRENN, as shown in Fig. 3e, leverages shared parameters across tasks and task-specific decoders to jointly optimize all m team constructs. This architecture enables more explainable and generalizable team modeling, particularly in high-stakes domains such as emergency response and healthcare. Rather than offering a mere incremental improvement, MT-TRENN represents a significant advancement from isolated predictions toward a holistic understanding of team dynamics. Table 5 compares the above mentioned architectures in terms of their modeling capabilities.

Implementing TRENN and MT-TRENN

In this section, we provide a detailed description of the TRENN and MT-TRENN approaches. We first present the input data (Sec. “From Raw-data to Temporal Graph”), followed by the description of the encoder (Sec. “Temporal-graph Encoding”), implemented as a TRENN. Finally, we present the decoder (Sec. “From Encoding to Predictions”), which extends TRENN into MT-TRENN, by leveraging MTL to jointly predict multiple team constructs (i.e., EL, LS, TW).

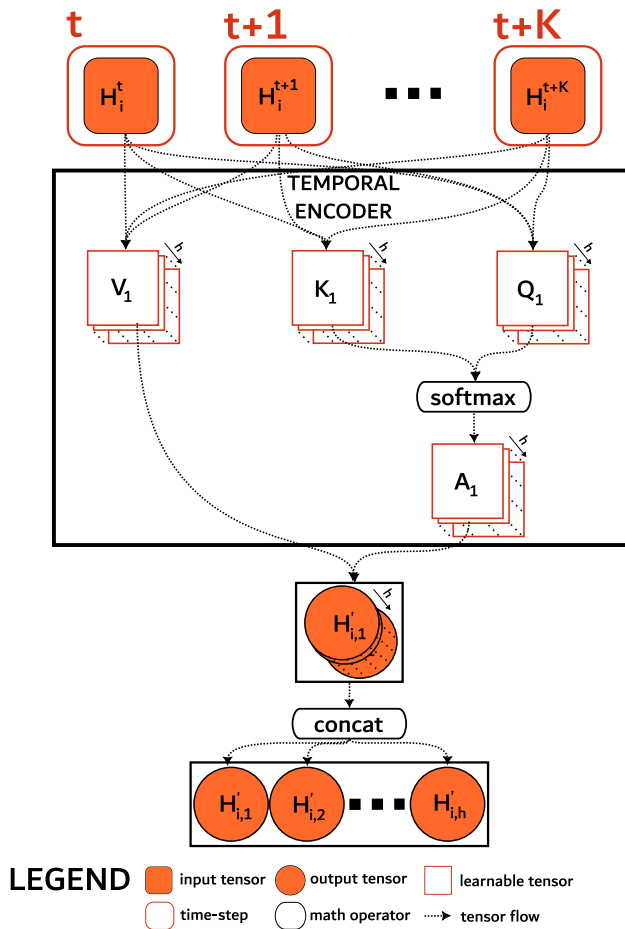


Fig. 6 Given the hidden representation of a team member i at time t , H_i^t , until time-step $t + K$, the input sequence $\{H_i^t, H_i^{t+1}, \dots, H_i^{t+K}\}$ is represented by the orange boxes that are the inputs to generate the query tensor Q and the key tensor K (the white boxes). Next, Q and K are combined to generate the Attention tensor A_j containing the attention scores. Then, the sequence of attention tensors A is combined with the value tensor V to generate the resulting hidden representation encoding temporal information $H' = \{H'_{i,1}, H'_{i,2}, \dots, H'_{i,h}\}$

From Raw-data to Temporal Graph

The raw input data, i.e., team behavioral data, is processed by a temporal graph extractor. This extractor converts team interactions into temporal graphs through four stages: (i) temporal segmentation, (ii) subsegmentation, (iii) feature extraction, and (iv) topology generation.

The raw input data consists of a set of multimodal team raw data (e.g., audio, video) $r = (r_1, r_2, \dots, r_r)$, where each segment r_i corresponds to team interactions lasting l_i seconds. In the temporal segmentation stage, given an annotation frequency f , each team behavioral segment r_i is partitioned into $t_i = \lceil \frac{l_i}{f} \rceil$ subsegments, that we refer to as $s_i^{(k)}$, $\forall k \in \{1, 2, \dots, t_i\}$. Each of these subsegments needs to be labeled with appropriate annotation y_i^k ,

$\forall k \in \{1, 2, \dots, t_i\}$ that enables us to define a unit of information as the pair $U_i^{(k)} = (s_i^{(k)}, y_i^{(k)})$. In the subsegmentation stage, each segment $s_i^{(k)}$ is further separated into $z = \lceil \frac{m}{f} \rceil$ subsegments $\tilde{s}_i^{(k),j}$, each lasting s seconds, such that: $s_i^{(k)} = \{\tilde{s}_i^{(k),1}, \tilde{s}_i^{(k),2}, \dots, \tilde{s}_i^{(k),z}\}$. Since annotations are performed at the team member level, the feature extraction stage computes the temporal features of each team member, while the topology generation stage derives their interactions over time. The combination of these two components, individual features and interactions, forms a *dynamic team* as defined in Definition 11 and represented in Fig. 4b. More specifically, given a subsegment $\tilde{s}_i^{(k),j}$, we obtain the corresponding static team $G_i^{(k),j} = \{V_i^{(k),j}, E_i^{(k),j}, X_i^{(k),j}\}$. Then, a dataset composed of p teams is denoted as $T = \{T_1, T_2, \dots, T_p\}$. This pipeline shows how to automatically convert raw input data like the one in Fig. 4a into a multivariate time series of features and interaction defined as a temporal graph. This structure provides flexibility about the length of each team interaction and the topology of each graph, as the set of team members and their interactions can change over time.

Temporal-graph Encoding

The encoder maps input data into a robust and informative representation suitable for downstream predictions. Each sample from the dynamic team dataset (Sec. “From Raw-data to Temporal Graph”) is a dynamic team, i.e., a sequence of temporally ordered team snapshots. To model this structure, we instantiate our TRENN architecture (Definition 12) as a Space-Then-Time TGNN (STT-TGNN) [138]. The STT-TGNN decouples spatial (i.e., relational) modeling and temporal (i.e., dynamics) modeling. This decoupling is aligned with theories in team cognition [139], which conceptualize team constructs as emerging from momentary relational configurations and their subsequent evolution over time. These two consecutive steps are realized as detailed below:

1. **Relational learning:** The relational learning module converts a *dynamic team* into a sequence of tensors encoding relational information. It receives as input a *dynamic team* $T = (G, Y)$ (See Definition 11). Each dynamic team consists of a sequence of *static team* snapshots G^j captured at discrete time steps. Each snapshot G^j is independently encoded into an embedding tensor H^j .
2. **Temporal learning:** The temporal learning module converts the output of the relational learning module, i.e., a sequence of tensors encoding relational information H_i , into a tensor embedding capturing also the temporal evolution of the team.

In the following subsections, we formalize the structure of the *encoder* according to the STT-TGNN implementation: (i) the relational learning module; (ii) the temporal learning module; and (iii) the contrastive learning module, which further enhances the robustness of the embeddings.

Relational Learning

The relational learning module is based on Message Passing Neural Networks (MPNNs) [140], a well-known paradigm to model non-sequential and variable-size relations among input data, as illustrated in Fig. 5. MPNNs propagate information across nodes through an iterative updating mechanism. It encodes information from neighboring nodes and combines it with the hidden representation of the current node. Given a dynamic team, i.e., a sequence of *static team* snapshots G^j , the relational learning module processes each static team snapshot independently $G_i^{(k),j} = \{\mathcal{V}_i^{(k),j}, \mathcal{E}_i^{(k),j}, X_i^{(k),j}\}$. For each team member $v \in \mathcal{V}_i^{(k),j}$, the relational learning module operates through two learnable components:

- a message function computing a message m from the neighboring nodes of the current node $\forall u \in \mathcal{N}(v)$ and their corresponding edge-features $e_{vu} \in \mathcal{E}_i^{(j),k}$:

$$m_v = \sum_{u \in \mathcal{N}(v)} m(h_v, h_u, e_{vu}) \quad (6)$$

- an update function generating an updated node hidden representation h'_v as the combination of the message from the neighboring nodes m_v with the current node hidden representation h_v :

$$h'_v = u(m_v, h_v) \quad (7)$$

Under the MPNNs framework, both the message function and the update function are differentiable and learnable functions. We model the kernel of this function as a GCN [82], which generalizes the convolution operator to graph data in order to compute the hidden representation for all the team members according to the following formula:

$$H^t = \phi \left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X^t W \right) \quad (8)$$

where ϕ is an activation function, t is the current time step, H^t refers to the hidden representation of all the team members at the current time step, $\tilde{A}$ is the adjusted adjacency matrix with added self-connection to increase stability, $\tilde{D}$ is the degree of the adjusted adjacency matrix, X^t is the tensor containing individual speech features of all team members

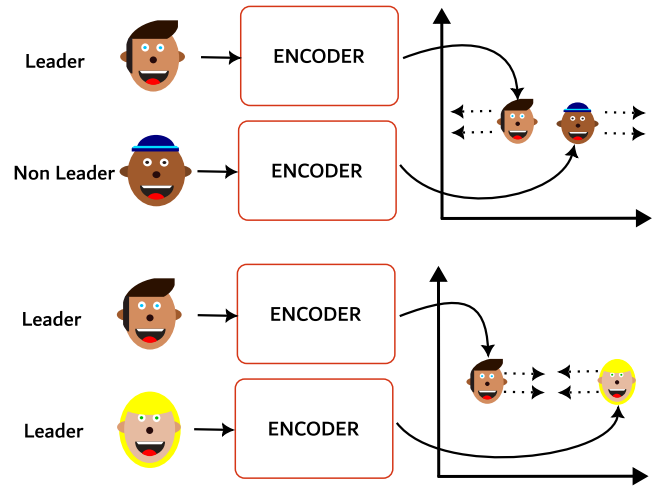


Fig. 7 Pairwise Ranking Loss enforces embeddings of the leaders to be closer in the embedding space, while the embeddings of non-leaders are pushed to be farther from the ones of the leaders

at the time step t and W refers to the learnable convolution parameters.

Temporal Learning

The temporal learning module is based on MHA [123] to effectively capture long-range dependencies across time steps. MHA receives as input the output of the relational learning module. At this stage, each dynamic team is encoded into a sequence of team member embeddings $H^t \in \mathbb{R}^{n \times d}$ for each time step t . Each row of these embeddings $H_i^t \in \mathbb{R}^d$ encodes the behavioral features and the interactions of a team member but it still completely ignores time dependencies.

To model time dynamics, MHA is introduced, it is built on top of the self-attention mechanism, which measures how much a specific time step is relevant to model another time-step in the same sequence, as illustrated in Fig. 6. For each team member, three learnable projection matrices are defined: the query-weight tensor $W_q \in \mathbb{R}^{d \times d_K}$, the key-weight tensor $W_k \in \mathbb{R}^{d \times d_K}$, and the value-weight tensor $W_v \in \mathbb{R}^{d \times d_v}$. Each one of them, respectively, is combined with the input H_i to generate the projected representations computed as:

$$Q_i = H_i W_q \in \mathbb{R}^{K \times d_K}, K_i = H_i W_k \in \mathbb{R}^{K \times d_K}, V_i = H_i W_v \in \mathbb{R}^{K \times d_v} \quad (9)$$

where K_i is the key tensor, Q_i is the query tensor, and V_i is the value tensor, while K is the sequence length, d_K is the key and query tensor dimension, and d_v is the value tensor dimension. Once these matrix multiplications are computed, each self-attention map A_i is computed as reported in (10).

$$A_i = \text{softmax} \left(\frac{Q_i K_i^T}{\sqrt{d_k}} \right) \in R^{K \times K}, \forall i \in \{1, 2, \dots, h\} \quad (10)$$

In (10) the attention scores are converted into attention probabilities via *softmax*, and they can be combined with the value tensor V_i to get the hidden representation as reported in (11).

$$H'_i = A_i \times V_i \in R^{K \times d_v}, \forall i \in \{1, 2, \dots, h\} \quad (11)$$

As previously mentioned, MHA extends the representational capacity of self-attention by applying multiple attention mechanisms in parallel, each with its own set of learnable parameters, and combines them via concatenation as reported in (12):

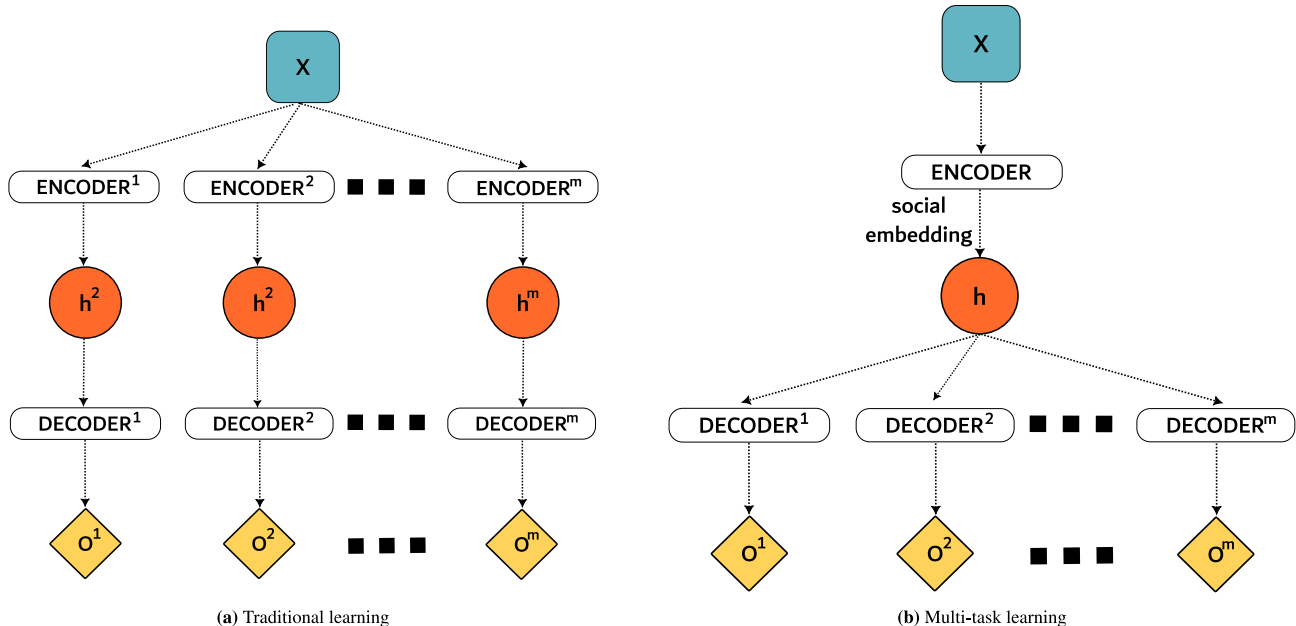
$$\text{MHA}(Q, K, V) = \text{Concat}(H'_1, H'_2, \dots, H'_h) \quad (12)$$

Thus, the model focuses on different temporal aspects of the sequence, improving its ability to capture complex and long-range dependencies in variable-length sequences.

Contrastive Learning

Beyond architectural design, a common challenge in team modeling is the scarcity and noisiness of labeled data, which stems from both the subjectivity of human annotations and the complexity of the high-level constructs being labeled. While data augmentation techniques are commonly used to mitigate the limited availability of data, they alone are not sufficient to ensure effective generalization, especially when the labels themselves are noisy and partially subjective.

To address this limitation, we introduce contrastive learning, which operates directly at the loss level to shape the structure of the embedding space. Rather than relying only on input-level transformations, we encode task-relevant inductive biases into the learning dynamics by incorporating ordinal supervision derived from human assessments. Each team member receives a real-valued score reflecting their perceived role or contribution. From these scores, we derive a within-team ranking: a relative ordering that expresses the team's internal structure. While standard supervised losses typically ignore such relational information, we leverage it via a Pairwise Ranking Loss that promotes embeddings to be aligned with the annotated hierarchy, as shown in Fig. 7. Given a ranking function $f(x)$ and two team members x_i and x_j with x_i ranked higher than x_j , the loss is defined as:



LEGEND ■ input ■ hidden representation ◆ output

Fig. 8 Two principal approaches for modeling multiple tasks in Neural Networks. **(a)**: Traditional learning approach: the input (blue node) feeds multiple backbones having different parameters. The output of the backbones are embeddings (orange nodes) that feed linear layers mapping them into the output space (yellow node) given an input sample (blue square) composed of three features, two parallel encoders composed of two hidden layers having hidden size four and three (orange square), learns different parameters to feed it as input to two

different linear layers mapping these hidden representation into the output space (yellow rhombi). **(b)** Multi-task Learning approach: the input (blue node) feeds a shared backbone that outputs a Social Embedding, an embedding encoding features to solve multiple tasks. The output, indeed, feeds two different specialized heads that learns specific parameters to specialize the embedding into different output spaces (yellow node)

$$L_p = \sum_{(i,j) \in D} \max(0, 1 - (f(x_i) - f(x_j))), \quad (13)$$

where D is the set of correctly ordered pairs of team members. This hinge-based loss enforces a margin between embeddings, encouraging the model to preserve intra-team ranking structure in the latent space. This contrastive signal complements traditional supervised objectives and induces a structured embedding geometry that reflects the underlying team hierarchy, thereby enhancing the consistency, robustness, and cross-task generalization of downstream predictions.

From Encoding to Predictions

When the task is predicting a single team construct, TRENN is employed, and the decoder reduces to a single FFNN. In the following, we focus on the more complex case when multiple team constructs are jointly modeled via the MT-TRENN architecture. In this architecture, the decoder is based on the MTL paradigm [141], which jointly learns multiple tasks (i.e., team constructs) rather than treating them as independent tasks. MTL employs a shared encoder across all the team constructs to map the input *dynamic graph* to a shared *social embedding* space (Definition 4) for each team member. This embedding is then passed to task-specific heads, each predicting a different team construct. This design allows the model to simultaneously capture shared dynamics across tasks while maintaining specialization for individual constructs (Fig. 8). Formally, given a set of tasks $\mathcal{A} = \{A_1, A_2, \dots, A_m\}$, with $m \geq 2$, the resulting MTL loss is defined as a weighted sum of task-specific losses:

$$\mathcal{L}_{\text{MTL}} = \sum_{i=1}^m \lambda_i \mathcal{L}_{T_i} \quad (14)$$

where each weight λ_i reflects the relative importance of the corresponding team construct T_i . To ensure positive coefficients $\lambda_i > 0$ for all i , we parameterize them via an exponential transformation:

$$\lambda_i = \exp(\alpha_i) \quad (15)$$

with $\alpha_i \in \mathbb{R}$ being optimized during the training procedure. This approach ensures no task is ignored during the joint optimization.

Unlike traditional learning (Fig. 8a), in which different team constructs do not share any parameters, and unlike fine-tuning, in which each decoder is optimized *post-hoc* independently for each task requiring an additional training procedure, in MTL (Fig. 8b), the backbone is trained jointly with all the heads, allowing mutual refinement of the shared encoder and task heads.

MTL exhibits two major advantages. First, it acts as a regularizer: joint optimization across tasks acts as an upper bound on the weights and the variance, resulting in the prevention of overfitting. Second, it improves efficiency, reducing both time complexity due to the single training procedure for all the tasks and memory cost due to the shared encoder parameters.

Experiments and Results

In this section, we first present our research questions. Next, we detail the experimental setup, covering datasets, evaluation protocol, and implementation details. Finally, results are reported and discussed. Our work aims to address the following research questions:

- Q1: Does TRENN outperform existing paradigms in team modeling?
- Q2: Does joint learning of Multiple Tasks increase efficiency in team modeling?
- Q3: Does MT-TRENN provide insights about team member contribution to team dynamics?

Datasets and Annotations

We used two state-of-the-art team datasets: PAVIS [46] and ELEA [45, 142]. Both datasets were recorded in laboratory settings and are based on the Winter or the Summer Survival task, a well-established team-decision-making task requiring teams to rank objects by importance for survival in extreme weather conditions. PAVIS is an audio-video dataset consisting of 16 teams of four people. Each team had a designated leader, appointed based on preliminary aptitude questionnaires filled out by participants. ELEA is an audio-video dataset of 40 teams of three or four peers. No specific

Table 6 Summary of the characteristics of the PAVIS and ELEA datasets. * indicates that the labeling procedure has been implemented by four annotators carried out directly for this work

Characteristic	PAVIS	ELEA
Scenario	Survival task	Survival task
# Teams	12	27
# Members per team	4	3-4
Total duration	~5 hours	~7 hours
Video recording	4 cameras	2 cameras
Audio recording	4 microphones	1 microphone
Annotation granularity	6 minutes	6 minutes
Emergent Leadership label	GLIS	GLIS*
Leadership style label	SYMLOG*	SYMLOG*
Teamwork label	BFTQ*	BFTQ*

roles, indeed, were attributed to the team members. In this study, we used a standard subset of these datasets consisting of 12 teams from PAVIS and 27 teams from ELEA, respectively. The other teams were discarded due to technical reasons (e.g., missing audio). The duration of recordings ranges between 10 and 30 minutes: ELEA averages 14.6 minutes (SD=1.68), and PAVIS 21.6 minutes (SD=2.24), further details about the datasets are reported in Table 6.

Annotations were obtained from four independent raters, each scoring different video subsets. The annotation granularity was set at 6-minute intervals. To operationalize the team constructs of interest, raters evaluated team functioning using validated instruments that measure core aspects of TW, EL, and LS. Specifically, the annotations encompass:

- TW components according to the BFT questionnaire [143]: It covers the five core behavioral components (see Table 2) and the three coordinating mechanisms (see Table 3) of the BFT framework. The questionnaire has 37 items assessed on a 7-point Likert scale: 5 items for A, MPM, and CC; 3 items for BB; 4 items for TL and SMS; 9 items for TO, and 2 items for MT;
- the General Leader Impression Scale (GLIS) [144] questionnaire, consisting of 5 items rated on a 5-point Likert scale;
- the Systematic method for the Multiple Level Observation of Groups (SYMLOG) [116, 145] questionnaire, a 26-item questionnaire assessed on a 5-point Likert scale referring to LS through three bipolar dimensions: Dominance vs. Submissiveness, Friendliness vs. Hostility, and Task-orientation vs. Expressiveness.

Data Pre-processing

The experiments we conducted, using TRENN and MT-TRENN implemented as an STT-TGNN (see Sec. “Temporal-graph Encoding”), take as input the datasets described in Sec. “Datasets and Annotations” from which paralinguistics features were extracted. The graph topology is defined through speaker diarization [146], the first step of

pre-processing, that determines “who speaks when”. A team member speaking is defined as a message-sender, while all the other team members as receivers. Such information is encoded in the graph through a directed edge from the node corresponding to the sender to all the other team members. Then, for each speaking team member, the 88 features of the eGeMAPS speech feature set [147] were extracted through the openSMILE library [148]. These features are paralinguistic features related to prosody (e.g., pitch, loudness), voice quality (e.g., jitter, shimmer), and spectral characteristics, validated by speech scientists, psychologists, and computer scientists.

Statistical functionals such as mean, normalized standard deviation, and percentiles are computed over the feature set to summarize their temporal dynamics within each time window.

Evaluation Protocol

We adopted a nested *Leave-One-Group-Out* (LOGO) evaluation protocol. Given the set of all available teams $T = [T_1, T_2, \dots, T_n]$, at each iteration one team T_i is selected as the validation set, and a different team T_j ($j \neq i$) as the test set. The remaining teams denoted as T^{trn} form the training set, i.e.,:

$$T^{\text{trn}} = T \setminus \{T_i \cup T_j\}.$$

This procedure is repeated for all possible combinations of validation-test team pairs, and the final performance is reported as the average over all resulting evaluations. Formally, let $P(f(\cdot), T_j)$ denote a performance metric of the model f on the test team T_j , and $f(T \setminus \{T_i \cup T_j\}, T_i)$ be the model trained on $T \setminus \{T_i \cup T_j\}$ and validated on T_i . Then, the expected performance is defined as:

$$\mathbb{E}[P(T)] = \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{\substack{j=1 \\ j \neq i}}^n P(f(T \setminus \{T_i \cup T_j\}), T_j) \quad (16)$$

Table 7 The average ACC@1 and ACC@Last of EL over 10 different seeds. The significantly best-performing model is boldfaced. Different models are grouped according to their main modeling paradigm (i.e., SNN, TNN, ReNN, TRENN)

Paradigm	Model	ACC@1 - Emergent Leadership		ACC@Last - Emergent Leadership	
		PAVIS	ELEA	PAVIS	ELEA
SNN	RF	79.54±0.18	70.28±0.12	78.97±0.53	71.28±0.47
	FFNN	77.05±0.54	67.68±0.94	78.24±0.72	70.11±0.65
TNN	LSTM	80.69±0.72	71.39±0.72	82.06±0.68	72.29±0.58
	MHA	81.25±0.56	72.48±0.52	83.37±0.45	74.98±0.44
ReNN	GCN	83.96±0.64	73.18±0.76	83.96±0.64	74.22±0.48
TRENN	GCN+MHA	85.95±0.48	74.55±0.48	85.95±0.58	75.42±0.40

The following evaluation metrics were adopted:

- Mean Squared Error (MSE): computed as the average squared difference between predicted and ground truth team construct scores across all team members e in the test set T_j :

$$\mathbb{E}[\text{MSE}(f, T)] = \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j=1, j \neq i}^n \sum_{e \in T_j} (f(e) - e_M)^2 \quad (17)$$

where $f(e)$ is the predicted score and e_M the ground truth for team member member e .

- Accuracy@1 and Accuracy@Last: measuring whether the model correctly identifies the highest- and lowest-ranked team members in the test team T_j according to a specific team construct score of interest:

$$\text{ACC@1} = \mathbb{1} \left(\arg \max_i M(T_j) = \arg \max_i f(T_j) \right) \quad (18)$$

$$\text{ACC@Last} = \mathbb{1} \left(\arg \min_i M(T_j) = \arg \min_i f(T_j) \right) \quad (19)$$

These metrics highlight both the prediction accuracy as well as the model's ability to preserve ordinal rankings within each team, and are measured over all the teams according to the LOGO protocol described above.

Experimental Results

In the following subsections, we investigate our three research questions.

Q1: Does TR_{ENN} outperform existing paradigms in Team modeling? To assess the effectiveness of TR_{ENN} for team modeling, we evaluated it on the task of predicting EL, LS, and the TW components (see Tables 2 and 3).

Emergent Leadership EL prediction is formulated as a regression task in which paralinguistic features are the

predictors, and the predicted variable is a score of EL for each team member in a team. EL scores are computed as the average of the scores obtained by GLIS. Table 7 reports the mean and standard deviation, over 10 different seeds, of ACC@1 and ACC@Last scores across the models described in Sec. “From Static to Multi-task Tempo Relational Team Modelling”: Random Forest (RF) and FFNN as SNNs, LSTM and MHA as TNNs, GCN as ReNN. As instances of TR_{ENN}, the combination of GCN to model relations and MHA to model time direction was used as described in Sec. “Implementing TR_{ENN} and MT-TR_{ENN}”. RF is implemented using 70 trees. The FFNN is designed as a three-layer architecture with the embedding size set to 16 with a weight decay of 0.01 for both ELEA and PAVIS to mitigate overfitting, while the learning rate is set to 0.0025 for PAVIS and 0.00125 for ELEA. The LSTM model is configured as a four-layer architecture with the embedding size set to 32, a weight decay of 0.015, and a learning rate of 0.001 for PAVIS and 0.0075 for ELEA. The MHA model employs four attention heads within a four-layer architecture with embedding size set to 64, with a weight decay of 0.012 for PAVIS and 0.02 for ELEA, and a learning rate of 0.00125 for both datasets. The GCN is implemented as a three-layer architecture with embedding size set to 32, a weight decay of 0.005, and a learning rate of 0.0075 in the case of PAVIS, while weight decay is set to 0.003 and learning rate 0.01 for ELEA. Finally, TR_{ENN} combines a two-layer GCN and a two-layer MHA with four attention heads with embedding size to 64, using a learning rate of 0.006 and a weight decay of 0.009 in the case of PAVIS and learning rate is set to 0.005. We assessed potential significant differences between the performances of the models through statistical tests with a significance α level equal to 0.05. First, tests to compare pairs of models implementing the same paradigm were run. Then, a between-paradigms test was performed with a post hoc analysis when needed and adjusting the p-value using the Bonferroni correction. The Shapiro-Wilk test showed a significant departure from normality of data. Thus, we chose non-parametric paired tests.

Table 8 Average MSE of LS (i.e., Dominance, Friendliness, Task orientation) over 10 different random seeds. The significantly best-performing method is boldfaced. Different models are grouped according to their main modeling paradigm (i.e., SNN, TNN, ReNN, TR_{ENN})

Paradigm	Model	PAVIS			ELEA		
		Dominance	Friendliness	Task orient.	Dominance	Friendliness	Task orient.
SNN	RF	0.661±0.008	0.605±0.009	1.023±0.007	0.729±0.010	0.738±0.009	0.958±0.008
	FFNN	0.738±0.009	0.652±0.011	1.245±0.011	0.758±0.013	0.782±0.0010	0.933±0.015
TNN	LSTM	0.589±0.006	0.592±0.005	0.767±0.007	0.685±0.009	0.715±0.011	0.891±0.009
	MHA	0.538±0.005	0.548±0.008	0.738±0.005	0.678±0.004	0.643±0.009	0.843±0.008
ReNN	GCN	0.513±0.004	0.503±0.005	0.701±0.004	0.638±0.008	0.652±0.010	0.839±0.009
TR _{ENN}	GCN+MHA	0.488±0.005	0.485±0.006	0.675±0.007	0.575±0.005	0.582±0.006	0.788±0.008

We conduct a two-sided Wilcoxon test to compare model pairs within each paradigm. For the SNN paradigm, the test shows that there is a significant difference between RF and FFNN on all metrics and datasets (all $p < 0.05$). Similarly, about the TNN paradigm, MHA performs significantly better than LSTM (all $p < 0.05$). Next, we carried out a Friedman chi-square test to compare models from the different paradigms. We chose RF and MHA from the previous analysis as representatives of SNN and TNN, respectively. The test reveals significant differences among the models. A post-hoc Conover test with a Bonferroni-adjusted α level confirmed that:

- MHA significantly outperforms RF across all metrics and across all datasets.
- ReNN (GCN) significantly outperforms RF and LSTM in ACC@1, but not in ACC@Last on ELEA.
- TRENN significantly outperforms ReNN (GCN) in both ACC@1 and ACC@Last (PAVIS), and also significantly outperforms TNN (MHA) in ACC@Last (ELEA).

These results support our hypothesis that jointly modeling temporal and relational patterns yields a better generalization across both datasets. MHA significantly outperforms LSTM, likely due to its greater capacity to model long-range dependencies, particularly relevant here given that sequences often span over 20 time steps, while LSTMs are prone to information bottlenecks. This may suggest that identifying the least emergent leader depends more on temporal patterns (e.g., silence, speech activity) than on modeling relational information.

Leadership Style We formulated the LS prediction task of each team member as a multi-output regression problem with predictors and predicted variables, the paralinguistic features, and the scores extracted by SYMLOG, respectively. Table 8 reports the MSE for each dimension. RF is implemented using 60 trees. The FFNN is designed as a three-layer architecture with the embedding size set to 16, the weight decay of 0.0125 to mitigate overfitting, and the learning rate set to 0.006 in the case of PAVIS, while in the case of ELEA, the learning rate is set to 0.0025 and the weight decay to 0.008. The LSTM model is configured as

a four-layer architecture with embedding size set to 32, a weight decay of 0.02, and a learning rate of 0.0013 in the case of PAVIS, while the learning rate of 0.0009 for ELEA. The MHA model employs four attention heads within a four-layer architecture with the embedding size set to 16, with a weight decay of 0.0175 and a learning rate of 0.0015 for both datasets. The GCN is implemented as a three-layer architecture with embedding size set to 16, a weight decay of 0.0075, and the learning rate of 0.0075 in the case of PAVIS, while the weight decay is 0.01 and the learning rate 0.006 in the case of ELEA. Finally, TRENN combines a two-layer GCN and a two-layer MHA with four attention heads, with embedding size set to 64, using a learning rate of 0.003 and a weight decay of 0.005 in the case of PAVIS, while learning rate 0.001 and weight decay 0.007 for ELEA. The reported performance scores are averaged over 10 different random seeds. As for EL, we compare the performances of the different models among them. Data deviation from normality suggested the adoption of non-parametric tests. First, we assessed the differences within SNN and TNN paradigms using Hotelling's T^2 test. The test shows that RF significantly outperforms FFNN and MHA significantly outperforms LSTM under both datasets, ELEA and PAVIS. Next, we compared the top-performing models for SNN and TNN paradigms with the other models using a PERMANOVA test with 1000 permutations. This analysis revealed significant differences among the models. A post hoc Pairwise PERMANOVA test with a Bonferroni-adjusted α level shows that:

- MHA and ReNN (GCN) both significantly outperform RF in both ELEA and PAVIS,
- ReNN (GCN) significantly outperforms MHA in both ELEA and PAVIS,
- TRENN (GCN+MHA) significantly outperforms ReNN (GCN) in both ELEA and PAVIS. *The findings support our hypothesis that both temporal and relational information are crucial for accurately modeling LS. Moreover, they indicate that relational information may play a comparatively stronger role in shaping leadership-related constructs.*

Table 9 Average MSE of the TW components over 10 different random seeds. The significantly best-performing method is boldfaced. Different models are grouped according to their main modeling paradigm (i.e., SNN, TNN, ReNN, TRENN)

		PAVIS							
Paradigm	Model	A	BB	CC	MPM	MT	SMM	TL	TO
SNN	RF	1.230±0.011	1.207±0.009	1.202±0.010	1.229±0.009	1.225±0.008	1.221±0.009	1.235±0.008	1.229±0.009
	FFNN	0.976±0.010	1.087±0.011	1.001±0.009	0.905±0.008	1.028±0.011	1.122±0.013	1.084±0.011	1.109±0.009
TNN	LSTM	0.765±0.009	0.763±0.007	0.772±0.008	0.758±0.009	0.854±0.012	0.844±0.009	0.856±0.011	0.824±0.013
	MHA	0.795±0.008	0.778±0.009	0.772±0.008	0.787±0.009	0.892±0.010	0.816±0.011	0.844±0.010	0.887±0.012
ReNN	GCN	0.731±0.008	0.726±0.009	0.731±0.008	0.763±0.007	0.909±0.007	0.855±0.006	0.834±0.008	0.875±0.005
TRENN	GCN + MHA	0.715±0.006	0.740±0.007	0.752±0.008	0.729±0.005	0.824±0.008	0.798±0.009	0.825±0.007	0.768±0.007

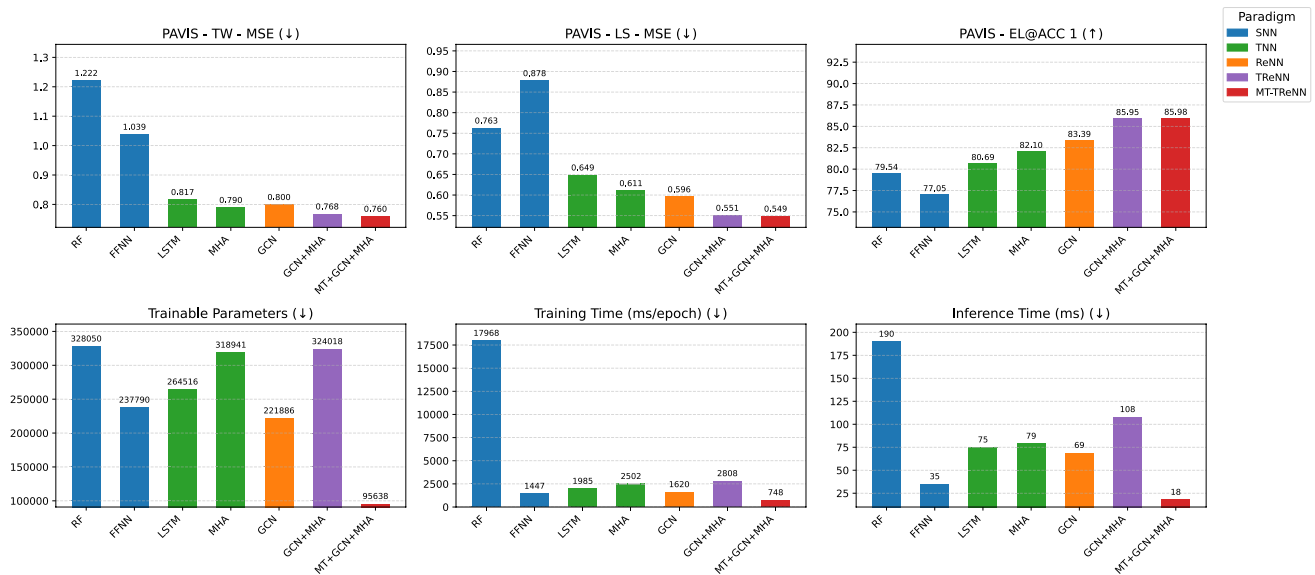


Fig. 9 Models' performance (PAVIS). In the first row, the MSE values of TW and of LS, and EL@ACC 1. In the second row, the trainable parameters required to achieve the performances, the training time (ms) for a single training epoch, and the inference time for a single team

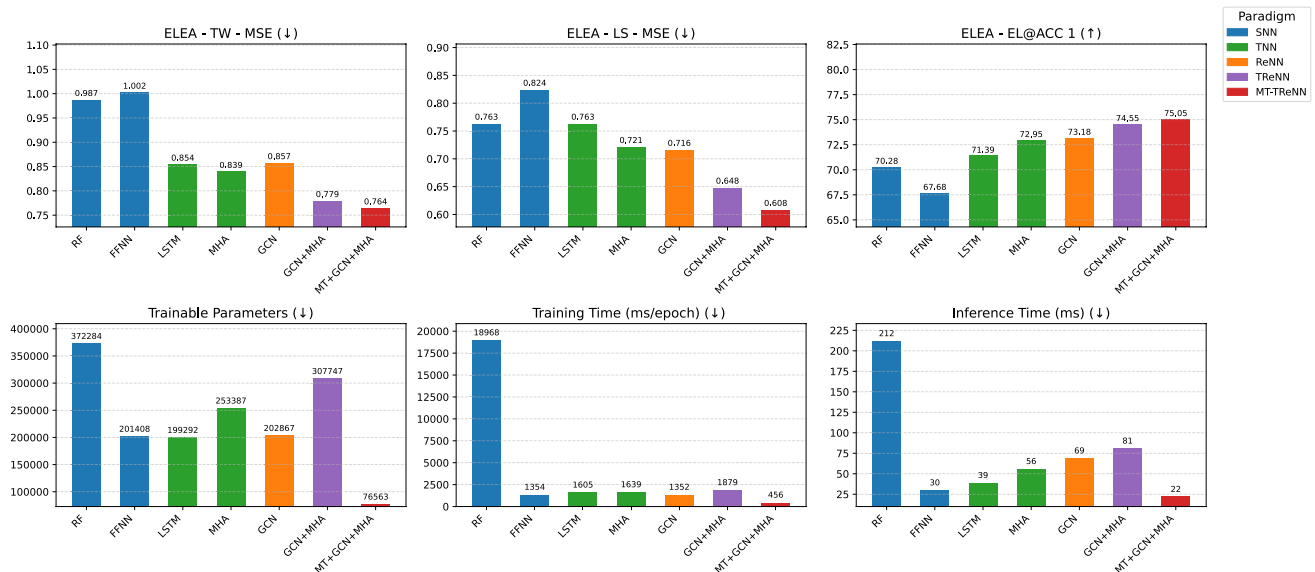


Fig. 10 Models' performance (ELEA). In the first row, the MSE values of TW and of LS, and EL@ACC1. In the second row, the trainable parameters required to achieve the performances, the training time (ms) for a single training epoch, and the inference time for a single team

Teamwork We formalized the prediction of the TW components as a set of regression tasks in which the predicted variables are the scores of each Teamwork component, computed as the average of the corresponding sub-scale of the BTF questionnaire. Table 9 shows the values of MSE over 10 runs with different random seeds. RF is implemented using 75 trees. The FFNN is designed as a two-layer architecture with 16 as the embedding size, with a weight decay of 0.0075 and a learning rate of 0.002 for PAVIS, and a weight decay of 0.01 with a learning rate of 0.001 for ELEA. The LSTM model adopts a four-layer architecture

with 32 as the embedding size, using a weight decay of 0.02 and a learning rate of 0.0075 for PAVIS, while for ELEA, a weight decay of 0.01 and a learning rate of 0.002 are applied. The MHA model uses four attention heads within a three-layer architecture with 32 as embedding size, with a weight decay of 0.008 and a learning rate of 0.00075 for PAVIS, and a weight decay of 0.01 with a learning rate of 0.00125 for ELEA. The GCN is implemented as a three-layer architecture with 32 as the embedding size, using a weight decay of 0.007 and a learning rate of 0.002 for PAVIS, and a weight decay of 0.0012 with a learning rate

of 0.001 for ELEA. Finally, TRENN combines a two-layer GCN with a two-layer MHA (four attention heads) with 64 as embedding size, using a learning rate of 0.001 and a weight decay of 0.0025 for PAVIS, and a learning rate of 0.0015 with a weight decay of 0.005 for ELEA. A Hotelling's T^2 test was used to verify the possible differences in the performance of the SNN and TNN models. The results show that FFNN outperforms RF, and LSTM outperforms MHA, for the SNN and TNN paradigms ($p < 0.05$) under both datasets. Next, a PERMANOVA test with 1000 permutations was run among the best SNN and TNN models and the other ones. This analysis revealed significant differences among the models. Thus, a post hoc Pairwise PERMANOVA test with a Bonferroni-adjusted α correction shows that:

- LSTM and RENN (GCN) both significantly outperform FFNN in both PAVIS and ELEA,
- RENN (GCN) significantly outperforms MHA in both PAVIS and ELEA,
- TRENN (GCN+MHA) significantly outperforms RENN (GCN) in both PAVIS and ELEA. *These findings support our hypothesis that jointly modeling temporal and relational information is crucial for accurately modeling TW. TRENN outperforms the other models in most teamwork dimensions, apart from the BB and CC components, for which it performs slightly worse than RENN. MHA significantly outperforms LSTM, likely due to its better capacity to model long-range dependencies, which is fundamental in such a case where we have a sequence of 20 time steps, while LSTM is prone to lose information in longer sequences due to information bottlenecks. Moreover, they indicate that relational information (GCN) may play a comparatively stronger role than temporal information (MHA) in shaping TW components, indeed, GCN significantly outperforms MHA.*

Q2: Does joint learning of Multiple Tasks increase efficiency in team modeling? To address Q2, we compared the performance of the MT-TRENN architecture with single-task alternatives.

Figures 9 and 10 report the results on the PAVIS and the ELEA datasets, respectively. The hyperparameter tuning of RF, FFNN, LSTM, MHA, GCN, and GCN+MHA is unaltered with respect to the previous, while MT-TRENN is defined as a GCN+MHA with 13 multi-task heads as a decoder, according to the design proposed in Sec. “Implementing TRENN and MT-TRENN”. The learning rate is set to 0.00025, and the importance of each task is automatically learned as described in Sec. “From Encoding to Predictions”, the weight decay is set to 0.001, the GCN is composed of two layers, the MHA is a two-layer architecture

with four heads, and the decoder is realized as a two-layer FFNN. The dropout is introduced in both the encoder and the FFNN. The top row compares predictive performance: the left plot shows MSE across TW components, the center plot reports average LS MSE (Dominance, Friendliness, Task Orientation), and the rightmost plot presents EL accuracy (ACC@1). MT-TRENN and TRENN consistently and significantly outperform all other architectures, while the performance improvement of MT-TRENN over TRENN is not statistically significant according to Hotelling's T^2 test. The effectiveness of MT-TRENN is apparent when considering computational costs and runtimes see the panel in Figs. 9 and 10. MT-TRENN achieves, indeed, the lowest number of trainable parameters (left panel), the fastest training per epoch (middle panel), and the shortest inference time (right panel) among all architectures. These benefits arise because, unlike task-specific models that require multiple forward passes over independently processed samples, MT-TRENN produces all predictions in a single forward pass over a unified team representation. When compared to TRENN, the only other architecture achieving comparable predictive performance, MT-TRENN has 70% fewer parameters, 75% reduction in training time, and 85% reduction in inference time.

The experimental results show that joint modeling multiple team constructs provides substantial advantages in terms of computational costs while retaining the predictive performance of the single task tempo-relational alternative. These results support the existence of a shared, tempo-relational embedding space in which high-level team dynamics coexist, providing a promising direction for future work on team representation learning.

Q3: Does MT-TRENN provide insights about team member contribution to team dynamics? Understanding why a team achieves its performance, and how team dynamics could be improved, is crucial in high-stakes collaborative settings. These challenges can be addressed, respectively, through two complementary forms of explainability: factual and counterfactual. In the case of SNN, explanations identify features or samples, in the case of TNN, also temporal steps, while in graphs, explanations can target nodes and edges, in graph-based temporal models, explanations can naturally target nodes (team members), edges (interactions), and time steps. This structural richness makes TRENN and MT-TRENN excellent candidates for producing two types of explanations. The first, factual explanations, provide a retrospective understanding of the factors driving model predictions. The second, counterfactual explanations, identify actionable modifications to the interaction dynamics that could improve team performance. In our framework, both forms of explanations are grounded in the predicted

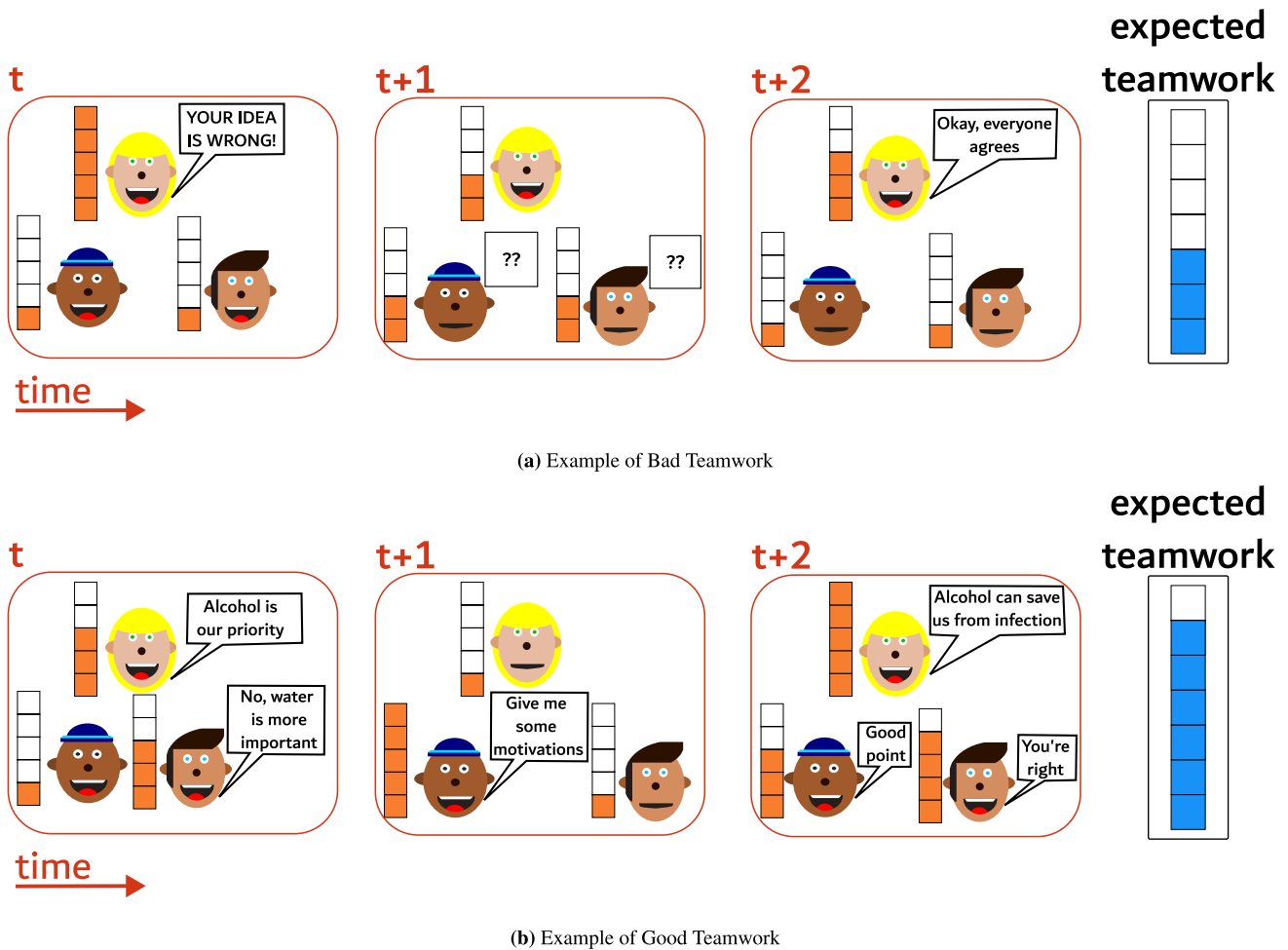


Fig. 11 Examples of factual explanations for poor ((a)) and good ((b)) teamwork: orange bars indicate individual attribution scores, while the blue bar shows the expected teamwork quality

teamwork effectiveness, computed as the average teamwork score across team members and teamwork components:

$$\mathbb{E}[\text{TW}] = \frac{1}{|G|} \sum_{v \in G} \frac{1}{m} \sum_{j=1}^m f_{\text{trenn}}(T, v)_j \quad (20)$$

This formulation enables MT-TReNN to move beyond post-hoc interpretation towards decision-support in collaborative environments by grounding both explanation types in a unified quantitative measure of team effectiveness. Factual explanations primarily support diagnostic understanding, highlighting which team members, interactions, and temporal segments most influenced the predicted teamwork score. Counterfactual explanations, instead, support actionable intervention, identifying minimal modifications to the interaction topology that are expected to improve teamwork. Taken together, these two forms of explanations allow TReNN and MT-TReNN to move beyond prediction, providing interpretable and practically useful insights into team

dynamics. Note that this section does not aim to provide a quantitative evaluation of explanation quality, as such an assessment would require either ground-truth explanations or domain experts capable of judging candidate actions. Instead, our goal is to highlight the potential of the explainability component to identify meaningful patterns that can inform and improve team behavior.

Factual explanations As illustrated in Fig. 1, in an Operating Room scenario, an analyst may post-hoc inquire about leadership or mutual trust among team members. To support this type of analysis, we rely on factual post-hoc explanations that highlight which interactions and team members most influence the model's predictions over time. In this study, we employed an instance-based and gradient-based method, Saliency Map [149]. Intuitively, Saliency Maps reflect how influential each input feature is for the model's output, which in our setting is the expected TW (20). The influence of a team member is computed by averaging its feature-level saliency scores. The rich topological structure of TReNN and MT-TReNN enables insights from both

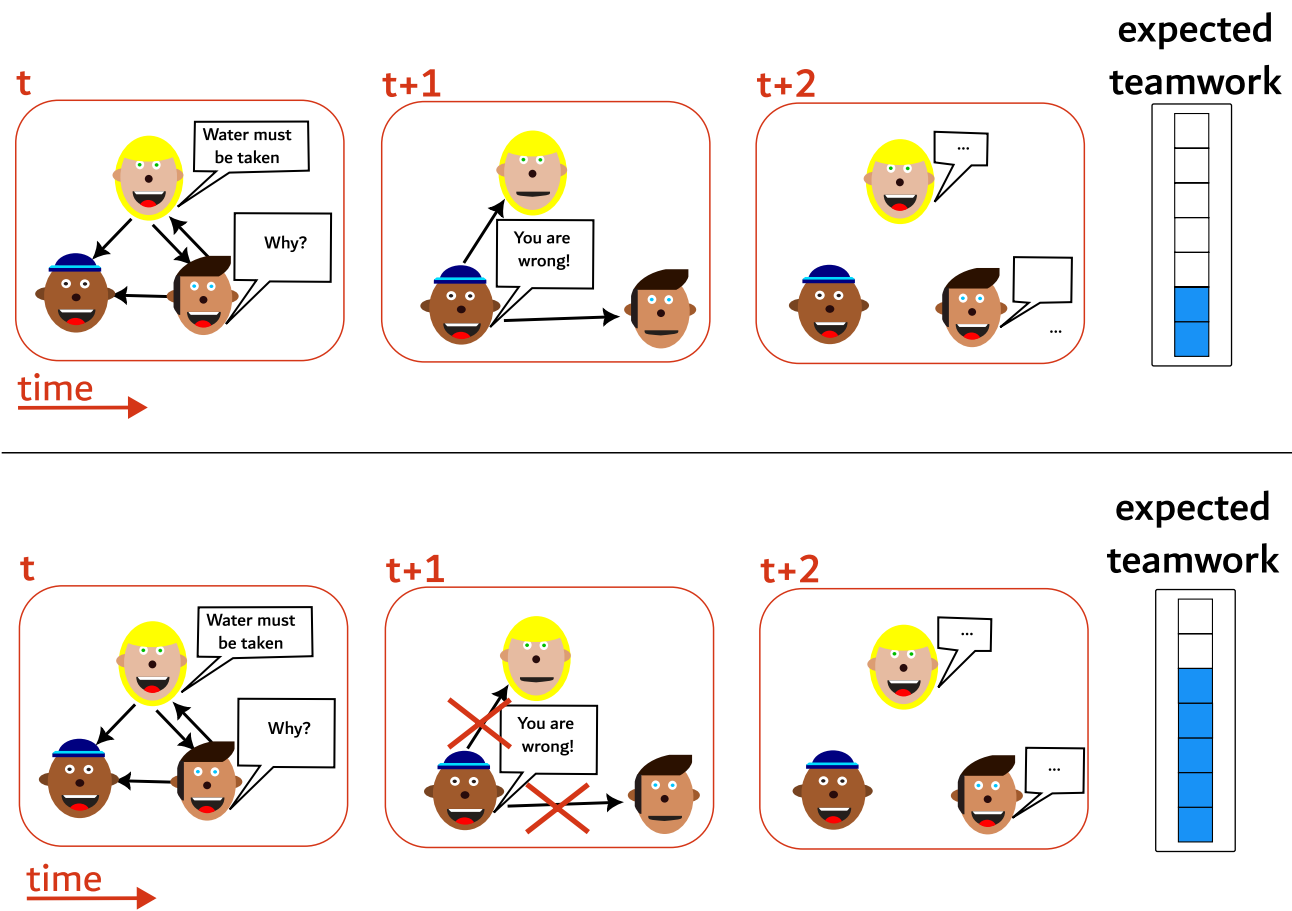


Fig. 12 A counterfactual example: In the top panel, a team interaction dynamic that results in poor teamwork is depicted. In the bottom panel, the counterfactual explanation suggested by CoDy is reported

temporal and relational perspectives, capturing how time and interactions influence individual team member-level predictions.

Figure 11 shows two anecdotal examples from the data: in each of them, team members are placed beside their attribution map score, which is discretized in five importance levels (orange bins), while the TW is shown on a seven-bin scale (blue bins). In the top panel of the figure, we depict a scenario of bad teamwork. The team leader (the blonde girl) aggressively addresses the other team members (timestep t), who refrain from responding, likely because of her aggressive tone ($t+1$), so that she can unilaterally end the argument in her favour ($t+2$). This situation is effectively captured by the attribution maps: the team leader has a maximal attribution score at the beginning and a high score at the end, while the other team members always have low scores, resulting in a low teamwork attribution score (blue bar). Conversely, the bottom panel illustrates a high-teamwork scenario. At timestep t , two team members (i.e., the blonde girl and the black-haired guy) discuss which object is more important to bring with them to solve the task. At timestep $t+1$, another team member (i.e., the one wearing a hat) asks for

clarification, fostering a collaborative tone. At the timestep $t+2$, the blonde girl offers a clear and well-reasoned explanation, prompting the agreement of the other two members. Individual attribution scores highlight the active participation of each team member, especially the request for clarification of the boy with the hat and the thoughtful response by the girl, resulting in a high average teamwork score.

Factual explanations generated via saliency maps enable an interpretable understanding of which team members and interactions most strongly influence the model's predictions over time. This exemplifies how TRENN can capture meaningful temporal and relational patterns that align with domain knowledge, making the model's reasoning process transparent and trustworthy.

Counterfactual explanations As illustrated in Fig. 1, in an Operating Room scenario, team interactions can exhibit inadequate behaviors that negatively affect teamwork. Such situations can be addressed through targeted suggestions provided by the SIA, either online or offline, with the goal of directly influencing team dynamics. To this end, we rely on counterfactual explanations that identify which interactions should be modified or avoided to promote more effective

collaboration, thereby fostering awareness of detrimental behaviors during operations. In our setting, we aim to detect critical interactions that can potentially lead to enhanced TW. This task can be achieved via greedy approaches acting on the topology of a dynamic team, like CoDy [150]. At each iteration, CoDy performs the following step: i) select a node based on a selection score balancing exploration and exploitation; (ii) simulate its prediction outcome; (iii) expand the node through its child nodes; (iv) backpropagate updated scores to guide the search toward minimal yet effective counterfactuals. In our setting, this approach detects those interactions that can increase TW the most.

Figure 12 depicts an example in which, at timestep t , a team member suggests an action (bringing water in the Winter Survival task) and another team member asks for an explanation. At the next timestep, the third team member forcefully asserts his opposing view, discouraging the others from speaking and ultimately undermining teamwork. The counterfactual explanation suggests the removal of the intervention of the third team member as the best modification to improve TW.

Counterfactual explanations provide actionable insights into crucial relational dynamics whose modification or removal could enhance team performance, highlighting the practical value of TRENN beyond pure prediction.

Limitations and Future Work

Despite the promising results, this work presents several limitations that should be acknowledged. First, the experimental evaluation is conducted on the limited number of datasets commonly adopted in the literature, which may constrain the generalizability of the findings. Team interactions are inherently domain-dependent, and the datasets considered include a restricted number of teams, primarily observed in controlled settings. Future work will focus on validating TRENN and MT-TRENN in larger-scale and in-the-wild team scenarios, as such data become publicly available. Second, the temporal graph extractor relies exclusively on paralinguistic features, which provide only a partial view of team behavior and limit the expressiveness of the resulting social embeddings. Incorporating additional modalities, such as lexical or visual cues, represents a natural extension of the proposed framework. Finally, although TRENN and MT-TRENN enable factual and counterfactual explanations, the lack of ground-truth explanations limits their evaluation to qualitative analyses. Future work will explore human-centered evaluation protocols to assess the usefulness and trustworthiness of the generated explanations. Overall, addressing these limitations will further strengthen the applicability, robustness, and interpretability of tempo-relational models for team behavior analysis.

Conclusion

We presented TRENN, a novel tempo-relational architecture for team modeling that jointly captures the relational and temporal structure of team dynamics. TRENN leverages an automatic temporal graph extractor to convert team behavioral data into temporal graphs, a tempo-relational encoder to generate an embedding from temporal graphs, and a decoder to predict a team construct from the generated embedding. TRENN outperforms alternative approaches that rely solely on temporal or relational information on two state-of-the-art datasets on EL, LS, and TW prediction. On top of TRENN, we presented its multi-task extension, MT-TRENN, that simultaneously can predict multiple team constructs by replacing the traditional decoder with several multi-task heads.

MT-TRENN shows predictive performance comparable to TRENN while drastically reducing space and time complexity. Beyond predictive performance, experiments show that these novel architectures can generate interpretable insights as factual explanations and actionable suggestions to enhance team performance as counterfactual explanations. While the lack of ground truth explanations prevents an extensive evaluation of the quality of the generated explanations, their actionability suggests the potential for on-field evaluations involving real teams and decision-making support tasks. Finally, the current experimental setting shows promising results using paralinguistic features only. Further improvements can be expected by incorporating more complex multimodal features.

Acknowledgements We acknowledge Raffaella Sabrina Fellone and Giuseppe De Luca for supporting the annotation process.

Author Contributions V.M.D.L. experimental setting design and validation; technical implementation; writing G.V. experimental setting design, writing, and reviewing A.P. experimental setting design, writing, and reviewing

Funding Open access funding provided by Università degli Studi di Trento within the CRUI-CARE Agreement. Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Health and Digital Executive Agency (HaDEA). Neither the European Union nor the granting authority can be held responsible for them. Grant Agreement no. 101120763 - TANGO. AP and VMDL acknowledge the support of the MUR PNRR project FAIR - Future AI Research (PE00000013) funded by the NextGenerationEU.

Data Availability This work is validated on the following state-of-the-art datasets in social interaction of small teams: 1) Leadership Corpus: <https://pavis.iit.it/leadership-corpus> 2) ELEA: <https://www.idiap.ch/en/scientific-research/data/elea>

Declarations

Competing Interests The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Chignell M, Wang L, Zare A, Li J (2023) The evolution of HCI and human factors: Integrating human and artificial intelligence. *ACM Trans Comput-Human Interact.* 2023;30(2):1–30.
- Kaasinen E, Anttila A-H, Heikkilä P, Laarni J, Koskinen H, Väättä A. Smooth and resilient human-machine teamwork as an industry 5.0 design challenge. *Sustainability.* 2022;14(5):2773.
- Garibay OO, Winslow B, Andolina S, Antona M, Bodenschatz A, Coursaris C, et al. Six human-centered artificial intelligence grand challenges. *Int J Human-Comput Inter.* 2023;39(3):391–437.
- Wang X, Yang J, Wang Y, Miao Q, Wang F-Y, Zhao A, et al. Steps toward industry 5.0: Building “6S” parallel industries with cyber-physical-social intelligence. *IEEE/CAA Journal of Automatica Sinica.* 2023;10(8):1692–703.
- Goleman D. *Social intelligence: The new science of human relationships.* Bantam Books; 2006.
- Williams Jessica, Fiore Stephen M, Jentsch Florian. Supporting artificial social intelligence with theory of mind. *Front Artif Intell.* 2022;5(2022):750763.
- Cuzzolin F, Morelli A, Cirstea B, Sahakian BJ. Knowing me, knowing you: theory of mind in AI. *Psychological Medicine.* 2020;50(7):1057–61.
- Human-AI Teaming. State-of-the-art and research needs. National Academies of Sciences, Engineering and Medicine, Washington DC. 2022;10(2022):26355.
- Westby S, Riedl C. Collective intelligence in human-AI teams: A Bayesian theory of mind approach. In *Proceedings of the AAAI Conference on Artificial Intelligence.* 2023;37:6119–27.
- Tavella F, Manzi F, Vinanzi S, Di Dio C, Massaro D, Cangelosi A, et al. Towards a computational model for higher orders of Theory of Mind in social agents. *Front RobotAI.* 2024;2024(11):1468756.
- McGrath JE. *Social psychology: A brief introduction.* (No Title) 1964;(1964).
- ID Steiner. *Group process and productivity;* 1972.
- Marks MA, Mathieu JE, Zaccaro SJ. A temporally based framework and taxonomy of team processes. *Acad Manag Rev.* 2001;26(3):356–76.
- Kozlowski SWJ, Ilgen DR. Enhancing the effectiveness of work groups and teams. *Psychol Sci Public Interest.* 2006;7(3):77–124.
- Mathieu J, Maynard MT, Rapp T, Gilson L. Team effectiveness 1997–2007: A review of recent advancements and a glimpse into the future. *J Manag.* 2008;34(3):410–76.
- Kozlowski SWJ, Chao GT. Unpacking team process dynamics and emergent phenomena: Challenges, conceptual advances, and innovative methods. *Am Psychol.* 2018;73(4):576.
- Klonek F, Twemlow M, Tims M, Parker SK. It's About Time! Understanding the Dynamic Team Process-Performance Relationship Using Micro-and Macroscale Time Lenses. *Group & Organization Management.* 2024;2024:10596011241278556.
- Vinciarelli A, Pantic M, Bourlard H. Social signal processing: Survey of an emerging domain. *Image and Vision Computing.* 2009;27(12):1743–59.
- Cristani M, Raghavendra R, Del Bue A, Murino V. Human behavior analysis in video surveillance: A social signal processing perspective. *Neurocomputing.* 2013;2013(100):86–97.
- Palaghias N, Hoseinitabatabaei SA, Nati M, Gluhak A, Moessner K. A survey on mobile social signal processing. *ACM Computing Surveys (CSUR).* 2016;48(4):1–52.
- Stergiou Alexandros, Poppe Ronald. Analyzing human-human interactions: A survey. *Computer Vision and Image Understanding.* 2019;188(2019):102799.
- Chanel G, Mühl C. Connecting brains and bodies: applying physiological computing to support social interaction. *Interacting with Computers.* 2015;27(5):534–50.
- Kaluarachchi T, Reis A, Nanayakkara S. A review of recent deep learning approaches in human-centered machine learning. *Sensors.* 2021;21(7):2514.
- de Visser EJ, Topoglu Y, Joshi S, Krueger F, Phillips E, Gratch J, et al. Designing man's new best friend: Enhancing human-robot dog interaction through dog-like framing and appearance. *Sensors.* 2022;22(3):1287.
- Abbasi NI, Spitale M, Anderson J, Ford T, Jones PB, Gunes H. Can robots help in the evaluation of mental wellbeing in children? an empirical study. In *2022 31st IEEE international conference on robot and human interactive communication (RO-MAN).* IEEE. 2022;1459–1466.
- Zu W, Song W, Chen R, Guo Z, Sun F, Tian Z, Pan W, Wang J. Language and sketching: An llm-driven interactive multimodal multitask robot navigation framework. In *2024 IEEE International Conference on Robotics and Automation (ICRA).* IEEE. 2024;1019–1025.
- Li W, Meng Z, Zhou J, Wei D, Gan C, Pfister H. Socialgpt: Prompting llms for social relation reasoning via greedy segment optimization. *Adv Neural Inf Process Syst.* 2024;2024(37):2267–91.
- Allen JA, Fisher C, Chetouani M, Chiu MM, Gunes H, Mehu M, et al. Comparing social science and computer science workflow processes for studying group interactions. *Small Group Research.* 2017;48(5):568–90.
- Reiter-Palmon R, Sinha T, Gevers J, Odobez JM, Volpe G. Theories and models of teams and groups. *Small Group Research.* 2017;48(5):544–67.
- Jackson MH. Modeling of Small Groups in Computational Sciences: A Prospecting Review. *Small Group Research.* 2024;2024:10464964241279164.
- Maman L, Lehmann-Willenbrock N, Chetouani M, Likforman-Sulem L, Varni G. Modeling the Interplay Between Cohesion Dimensions: a Challenge for Group Affective Emergent States. *IEEE Transactions on Affective Computing.* 2024;15(3):1526–38.
- Corbellini N, Ceccaldi E, Varni G, Volpe G. An exploratory study on group potency classification from non-verbal social behaviours. In: *International Conference on Pattern Recognition.* Springer. 2022;240–255.
- Hung H, Gatica-Perez D. Estimating Cohesion in Small Groups Using Audio-Visual Nonverbal Behavior. *IEEE Trans Multimedia.* 2010;12(6):563–75.
- Murray G, Oertel C. Predicting group performance in task-based interaction. In: *Proceedings of the 20th ACM International Conference on Multimodal Interaction.* 2018;14–20.
- Sabry S, Maman L, Varni G. An Exploratory Computational Study on the Effect of Emergent Leadership on Social and Task Cohesion. In: *Companion Publication of the 2021 International Conference on Multimodal Interaction.* New York, NY, USA: Association for Computing Machinery; 2021. p. 263–72.

36. Wang D, Yang Q, Abdul A, Lim BY. Designing theory-driven user-centric explainable AI. In: Proceedings of the 2019 CHI conference on human factors in computing systems. 2019;1–15.
37. Goodman B, Flaxman S. European Union regulations on algorithmic decision-making and a “right to explanation”. *AI magazine*. 2017;38(3):50–7.
38. Catchpole K, Mishra A, Handa A, McCulloch P. Teamwork and error in the operating room: analysis of skills and roles. *Annals of Surgery*. 2008;247(4):699–706.
39. Pronovost P. Teamwork matters. Developing and enhancing teamwork in organizations: Evidence-based best practices and guidelines. San Francisco, CA: Jossey-Bass. 2013;2013:11–2.
40. Hu Y-Y, Parker SH, Lipsitz SR, Arriaga AF, Peyre SE, Corso KA, et al. Surgeons’ leadership styles and team behavior in the operating room. *J Am College of Surgeons*. 2016;222(1):41–51.
41. Parker SH, Yule S, Flin R, McKinley A. Surgeons’ leadership in the operating room: an observational study. *Am J Surgery*. 2012;204(3):347–54.
42. Brown A, Patterson DA. To err is human. In Proceedings of the First Workshop on evaluating and architecting system dependability (EASY’01); 2001.
43. Makary MA, Daniel M. Medical error—the third leading cause of death in the US. *Bmj*. 2016;353.
44. Kraaij W, Hain T, Lincoln M, Post W. The AMI meeting corpus. In: Proceedings of the International Conference on Methods and Techniques in Behavioral Research. 2005;1–4.
45. Sanchez-Cortes D, Aran O, Gatica-Perez D. An audio visual corpus for emergent leader analysis. In: Workshop on multimodal corpora for machine learning: taking stock and road mapping the future, ICMMLMI. Citeseer; 2011.
46. Beyan C, Carissimi N, Capozzi F, Vascon S, Bustreo M, Pierro A, Becchio C, Murino V. Detecting emergent leader in a meeting environment using nonverbal visual features only. In: Proceedings of the 18th ACM international conference on multimodal interaction. 2016;317–324.
47. Braley M, Murray G. The group affect and performance (GAP) corpus. 2018;(2018):1–9.
48. Bhattacharya I, Foley M, Ku C, Zhang N, Zhang T, Mine C, Li M, Ji H, Riedl C, Welles BF, et al. The unobtrusive group interaction (UGI) corpus. In: Proceedings of the 10th ACM Multimedia Systems Conference. 2019;249–254.
49. Müller P, Dietz M, Schiller D, Thomas D, Zhang G, Gebhard P, André E, Bulling A. Multimediata: Multi-modal group behaviour analysis for artificial mediation. In: Proceedings of the 29th ACM International Conference on Multimedia. 2021;4878–4882.
50. Enayet A, Sukthankar G. Analyzing team performance with embeddings from multiparty dialogues. In: 2021 IEEE 15th International Conference on Semantic Computing (ICSC). IEEE. 2021;33–39.
51. Yang C, Wang K, Chen PQ, Cheung MKM, Zhang Y, Fu EY, Ngai G. Multimediata 2023: Engagement level detection using audio and video features. In: Proceedings of the 31st ACM International Conference on Multimedia. 2023;9601–9605.
52. Nanninga MC, Zhang Y, Lehmann-Willenbrock N, Szilávik Z, Hung H. Estimating Verbal Expressions of Task and Social Cohesion in Meetings by Quantifying Paralinguistic Mimicry. In: Proceedings of the 19th ACM International Conference on Multimodal Interaction. Association for Computing Machinery. 2017;206–215.
53. Beyan C, Katsageorgiou V-M, Murino V. Moving as a leader: Detecting emergent leadership in small groups using body pose. In: Proceedings of the 25th ACM international conference on Multimedia. 2017;1425–1433.
54. Kubasova U, Murray G, Braley M. Analyzing verbal and non-verbal features for predicting group performance; 2019. arXiv preprint [arXiv:1907.01369](https://arxiv.org/abs/1907.01369).
55. Müller P, Balazia M, Baur T, Dietz M, Heimerl A, Penzkofer A, Schiller D, Brémond F, Alexandersson J, André E, et al. Multi-Mediate’24: Multi-Domain Engagement Estimation. In: Proceedings of the 32nd ACM International Conference on Multimedia. 2024;11377–11382.
56. Rasipuram S, Jayagopi DB. Automatic multimodal assessment of soft skills in social interactions: a review. *Multimedia Tools and Applications*. 2020;79(19):13037–60.
57. Bavelas A. Communication patterns in task-oriented groups. *The journal of the acoustical society of America*. 1950;22(6):725–30.
58. Lin Y-S, Lee C-C. Predicting performance outcome with a conversational graph convolutional network for small group interactions. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE. 2020;8044–8048.
59. Miller C. Using Parallel Episodes of Speech to Represent and Identify Interaction Dynamics for Group Meetings. In: Proceedings of the Group Interaction Frontiers in Technology (Boulder, CO, USA) (GIFT’18). Association for Computing Machinery, New York, NY, USA. 2018;Article 10, 7 pages.
60. McGrath JE. Small group research, that once and future field: An interpretation of the past with an eye to the future. *Group dynamics: Theory, research, and practice*. 1997;1(1):7.
61. Gorman JC, Dunbar TA, Grimm D, Gipson CL. Understanding and modeling teams as dynamical systems. *Frontiers in psychology*. 2017;2017(8):1053.
62. Sutskever I. Sequence to Sequence Learning with Neural Networks; 2014. arXiv preprint [arXiv:1409.3215](https://arxiv.org/abs/1409.3215).
63. Avci U, Aran O. Predicting the performance in decision-making tasks: From individual cues to group interaction. *IEEE Transactions on Multimedia*. 2016;18(4):643–58.
64. Beyan C, Katsageorgiou V-M, Murino V. A Sequential Data Analysis Approach to Detect Emergent Leaders in Small Groups. *IEEE Transactions on Multimedia*. 2019;21(8):2107–16.
65. Lin Y-S, Liu Y-C, Lee C-C. An Interaction-process-guided Framework for Small-group Performance Prediction. *ACM Transactions on Multimedia Computing, Communications and Applications*. 2023;19(2):1–25.
66. Amer A, Bhuvaneshwara C, Addluri GK, Shaik MM, Bonde V, Müller P. Backchannel Detection and Agreement Estimation from Video with Transformer Networks. In: 2023 International Joint Conference on Neural Networks (IJCNN). 2023;1–8.
67. Qiu J, Huang G, Lee TS. A neurally-inspired hierarchical prediction network for spatiotemporal sequence learning and prediction; 2019.
68. Seo S, Kennedy-Metz LR, Zenati MA, Shah JA, Dias RD, Unhelkar VV. Towards an AI coach to infer team mental model alignment in healthcare. In: 2021 IEEE Conference on Cognitive and Computational Aspects of Situation Management (CogSIMA). IEEE. 2021;39–44.
69. Seo S, Han B, Unhelkar V. Automated task-time interventions to improve teamwork using imitation learning; 2023. arXiv preprint [arXiv:2303.00413](https://arxiv.org/abs/2303.00413).
70. Seo S, Han B, Harari RE, Dias RD, Zenati MA, Salas E, Unhelkar V. Socratic: Enhancing Human Teamwork via AI-enabled Coaching; 2025. arXiv preprint [arXiv:2502.17643](https://arxiv.org/abs/2502.17643).
71. Yu C, Yang X, Gao J, Yang H, Wang Y, Wu Y. Learning efficient multi-agent cooperative visual exploration. In: European Conference on Computer Vision. Springer. 2022;497–515.
72. Li S, Cao Q, Liu L, Yang K, Liu S, Hou J, Yi S. Groupformer: Group activity recognition with clustered spatial-temporal transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021;13668–13677.
73. Junior JC, Güçlütürk Y, Pérez M, Güçlü U, Andujar C, Baró X, et al. First impressions: A survey on vision-based apparent

- personality trait analysis. *IEEE Transactions on Affective Computing*. 2019;13(1):75–95.
74. Subburaj SK, Stewart AE, Ramesh Rao A, D'Mello SK. Multimodal, Multiparty Modeling of Collaborative Problem Solving Performance. In: *Proceedings of the 2020 International Conference on Multimodal Interaction (Virtual Event, Netherlands) (ICMI '20)*. Association for Computing Machinery, New York, NY, USA. 2020;423–432.
 75. Leenders RTAJ, Contractor NS, DeChurch LA. Once upon a time: Understanding team processes as relational event networks. *Organizational Psychology Review*. 2016;6(1):92–115.
 76. Chai A, Le JP, Lee AS, Lo SM. Applying graph theory to examine the dynamics of student discussions in small-group learning. *CBE—Life Sciences Education*. 2019;18(2):ar29.
 77. Dooley KJ. A complex adaptive systems model of organization change. *Nonlinear dynamics, psychology, and life sciences*. 1997;1(1997):69–97.
 78. Arrow H, McGrath JE, Berdahl JL. *Small groups as complex systems: Formation, coordination, development, and adaptation*. Sage Publications; 2000.
 79. Ramos-Villagrasa PJ, Marques-Quinteiro P, Navarro J, Rico R. Teams as complex adaptive systems: Reviewing 17 years of research. *Small group research*. 2018;49(2):135–76.
 80. Gori M, Monfardini G, Scarselli F. A new model for learning in graph domains. In: *Proceedings. 2005 IEEE international joint conference on neural networks*. 2005;Vol. 2. IEEE, 729–734.
 81. Scarselli F, Gori M, Tsoi AC, Hagenbuchner M, Monfardini G. The graph neural network model. *IEEE Trans Neural Netw*. 2008;20(1):61–80.
 82. Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks; 2016. [arXiv:1609.02907](https://arxiv.org/abs/1609.02907).
 83. Sharma G, Ghosh S, Dhall A, Hayat M, Cai J, Gedeon T. GraphITTI: Attributed Graph-based Dominance Ranking in Social Interaction Videos. In: *Companion Publication of the 25th International Conference on Multimodal Interaction*. 2023;323–329.
 84. Yang F, Yin W, Inamura T, Björkman M, Peters C. Group behavior recognition using attention-and graph-based neural networks. In: *ECAI 2020*. IOS Press. 2020;1626–1633.
 85. Bai C, Bolonkin M, Regunath V, Subrahmanian VS. DIPS: a dyadic impression prediction system for group interaction videos. *ACM Trans Multimedia Comput Commun Appl*. 2023;19(1s):1–24.
 86. Malik M, Isik L. Relational visual representations underlie human social interaction recognition. *Nature Communications*. 2023;14(1):7317.
 87. Tan H, Bansal M. Lxmert: Learning cross-modality encoder representations from transformers; 2019. [arXiv:1908.07490](https://arxiv.org/abs/1908.07490).
 88. Li J, Yu Y, Chen Y, Zhang Y, Jia P, Xu Y, Li Z, Wang M, Hong R. DAT: Dialogue-Aware Transformer with Modality-Group Fusion for Human Engagement Estimation. In: *Proceedings of the 32nd ACM International Conference on Multimedia*. 2024;11397–11403.
 89. Seo S, Unhelkar V. Hierarchical Imitation Learning of Team Behavior from Heterogeneous Demonstrations; 2025. [arXiv:2502.17618](https://arxiv.org/abs/2502.17618).
 90. ang W, Qing L, Wang P, Li L, Zhu C. *IEEE Transactions on Image Processing: Hypergraph Mamba Reasoning Based Social Relation Recognition*; 2025.
 91. Tang W, Qing L, Wang P, Li L, Peng Y. Cross-attention fusion of graph and MLLM for social relation recognition. *Information Fusion*. 2025;2025(123):103259.
 92. Sebo S, Stoll B, Scassellati B, Jung MF. Robots in groups and teams: a literature review. In: *Proceedings of the ACM on Human-Computer Interaction 4, CSCW2 (2020)*. 2020;1–36.
 93. Chugunova M, Sele D. We and it: an interdisciplinary review of the experimental evidence on how humans interact with machines. *Journal of Behavioral and Experimental Economics*. 2022;99(2022):101897.
 94. Dahiya A, Aroyo AM, Dautenhahn K, Smith SL. A survey of multi-agent human-robot interaction systems. *Robotics and Autonomous Systems*. 2023;161(2023):104335.
 95. Beyan C, Capozzi F, Becchio C, Murino V. Multi-task learning of social psychology assessments and nonverbal features for automatic leadership identification. In: *Proceedings of the 19th ACM International Conference on Multimodal Interaction*. 2017;451–455.
 96. Ghosh S, Dhall A, Sebe N, Gedeon T. Automatic prediction of group cohesiveness in images. *IEEE Transactions on Affective Computing*. 2020;13(3):1677–90.
 97. Maman L, Ceccaldi E, Lehmann-Willenbrock N, Likforman-Sulem L, Chetouani M, Volpe Gualtiero, et al. Game-on: A multimodal dataset for cohesion and group analysis. *IEEE Access*. 2020;8(2020):124185–203.
 98. Morgan MM, Bhattacharya I, Radke RJ, Braasch J. Classifying the emotional speech content of participants in group meetings using convolutional long short-term memory network. *The Journal of the Acoustical Society of America*. 2021;149(2):885–94.
 99. Dyer JL. Team research and team training: A state-of-the-art review. *Human factors review*. 1984;26(1984):285–323.
 100. Salas E. 1992. Toward an understanding of team performance and training. *Teams: Their training and In Their training and performance/Ablex*; 1992.
 101. Moreland RL. Are Dyads Really Groups? *Small Group Research*. 2010;41(2):251–67.
 102. Cooke NJ, Gorman JC, Myers CW, Duran JL. Interactive team cognition. *Cognitive science*. 2013;2013(37):255–85.
 103. Delice F, Rousseau M, Feitosa J. Advancing teams research: What, when, and how to measure team dynamics over time. *Frontiers in psychology*. 2019;10(2019):1324.
 104. Salas E, Sims DE, Burke CS. Is there a “big five” in teamwork? *Small group research*. 2005;36(5):555–99.
 105. Fiske AP. *Structures of Social Life: The Four Elementary Forms of Human Relations*. Free Press; 1991.
 106. Fiske AP. *The Four Elementary Forms of Sociality: Framework for a Unified Theory of Social Relations*. *Psychological Review*. 1992;99(4):689–723.
 107. Lewin K. *Field theory*, 1948.
 108. Morgan BB, Glickman AS, Woodard EA, Blaiwes AS. Measurement of team behaviors in a Navy training environment. *PsycEXTRA Dataset*; 1986.
 109. Salas E, Sims DE, Klein C. *Cooperation at Work*; 2004.
 110. Stein RT. Identifying emergent leaders from verbal and nonverbal communications. *Journal of Personality and Social psychology*. 1975;32(1):125.
 111. Ford DL, Kiran MI. Perceived leader effectiveness across organizational roles: Exploratory evidence from Central Eurasia. *Eurasian Journal of Business and Economics*. 2008;1(2):131–56.
 112. Zaccaro SJ, Rittman AL, Marks MA. Team leadership. *The leadership quarterly*. 2001;12(4):451–83.
 113. Burke CS, Stagl KC, Klein C, Goodwin GF, Salas E, Halpin SM. What type of leadership behaviors are functional in teams? A meta-analysis *The leadership quarterly*. 2006;17(3):288–307.
 114. Sauer SJ. Taking the reins: the effects of new leader status and leadership style on team performance. *Journal of Applied Psychology*. 2011;96(3):574.
 115. Dinh JE, Lord RG, Gardner WL, Meuser JD, Liden RC, Hu J. Leadership theory and research in the new millennium: Current theoretical trends and changing perspectives. *The leadership quarterly*. 2014;25(1):36–62.

116. Bales RF. SYMLOG: Case study kit with instructions for a group self study. (No Title); 1980.
117. Kozlowski SWJ, Gully SM, McHugh PP, Salas E, Cannon-Bowers JA. A dynamic theory of leadership and team effectiveness: Developmental and task contingent leader roles. *Research in personnel and human resources management*. 1996;14(1996):253–306.
118. Kozlowski SWJ, Ilgen DR. Enhancing the effectiveness of work groups and teams. *Psychological science in the public interest*. 2006;7(3):77–124.
119. DeRue DS, Ashford SJ. Who will lead and who will follow? A social process of leadership identity construction in organizations. *Acad Manag Rev*. 2010;35(4):627–47.
120. I. Norton Jr W, L. Ueltschy Murfield M, S. Baucus M. Leader emergence: the development of a theoretical framework. *Leadership & Organization Development Journal*. 2014;35(6):513–529.
121. Schmidhuber Jürgen, Hochreiter S, et al. Long short-term memory. *Neural Comput*. 1997;9(8):1735–80.
122. Cho K, et al. Learning phrase representations using RNN encoder–decoder for statistical machine translation; 2014. arXiv preprint [arXiv:1406.1078](https://arxiv.org/abs/1406.1078).
123. Vaswani A. Attention is all you need. *Advances in Neural Information Processing Systems*; 2017.
124. Dotti D, Popa M, Asteriadis S. Being the center of attention: A person-context cnn framework for personality recognition. *ACM Transactions on Interactive Intelligent Systems*. 2020 Nov 9;10(3):1–20.
125. Chien W-S, Chou H-C, Lee C-C. Self-assessed emotion classification from acoustic and physiological features within small-group conversation. In: *Companion Publication of the 2021 International Conference on Multimodal Interaction*. 2021; 230–239.
126. Chien W-S, Chou H-C, Lee C-C. Belongingness and satisfaction recognition from physiological synchrony with a group-modulated attentive BLSTM under small-group conversation. In: *Companion Publication of the 2021 International Conference on Multimodal Interaction*. 2021;220–229.
127. Kim J, Shah JA. Improving team's consistency of understanding in meetings. *IEEE Transactions on Human-Machine Systems*. 2016;46(5):625–37.
128. Cannon-Bowers JA, Salas E. Teamwork competencies: The interaction of team member knowledge, skills, and attitudes. *Workforce readiness: Competencies and assessment*. 1997;1997:151–74.
129. Pype P, Mertens F, Helewaut F, Krystallidou D. Healthcare teams as complex adaptive systems: understanding team behaviour through team members' perception of interpersonal interaction. *BMC health services research*. 2018;18(2018):1–13.
130. Park S, Grosser TJ, Roebuck AA, Mathieu JE. Understanding work teams from a network perspective: A review and future research directions. *Journal of Management*. 2020;46(6):1002–28.
131. Velickovic P, Cucurull G, Casanova A, Romero A, Lio P, Bengio Yoshua, et al. Graph attention networks stat. 2017;1050(20):10–48550.
132. Xu K, Hu W, Leskovec J, Jegelka S. How Powerful are Graph Neural Networks?. In: *International Conference on Learning Representations*; [n. d.].
133. De Luca VM, Longa A, Lio P, Passerini A. xAI-Drop: Don't Use What You Cannot Explain. In: *Proceedings of the 3rd Learning on Graphs Conference (Proceedings of Machine Learning Research, Vol. 269)*, Wolf G, Krishnaswamy S, editors. PMLR, 2025;16:1–16:22. <https://proceedings.mlr.press/v269/luca25a.html>
134. Azzolin S, Longa A, Teso S, Passerini A. Reconsidering Faithfulness in Regular, Self-Explainable and Domain Invariant GNNs. In: *The 13th International Conference on Learning Representations*; [n. d.].
135. Pareja Aldo, Domeniconi Giacomo, Chen Jie, Ma Tengfei, Suzumura Toyotaro, Kanezashi Hiroki, et al. Evolvegc: Evolving graph convolutional networks for dynamic graphs. In: *Proceedings of the AAAI conference on artificial intelligence*. 2020;34:5363–70.
136. Sankar A, Wu Y, Gou L, Zhang W, Yang H. Dysat: Deep neural representation learning on dynamic graphs via self-attention networks. In: *Proceedings of the 13th international conference on web search and data mining*. 2020;519–527.
137. Rossi E, Chamberlain B, Frasca F, Eynard D, Monti F, Bronstein M. Temporal graph networks for deep learning on dynamic graphs; 2020. arXiv preprint [arXiv:2006.10637](https://arxiv.org/abs/2006.10637).
138. Cini A, Marisca I, Zambon D, Alippi C. Graph deep learning for time series forecasting; 2023. arXiv preprint [arXiv:2310.15978](https://arxiv.org/abs/2310.15978).
139. Courtright SH, Thurgood GR, Stewart GL, Pierotti AJ. Structural interdependence in teams: An integrative framework and meta-analysis. *J Appl Psychol*. 2015;100(6):1825.
140. Gilmer J, Schoenholz SS, Riley PF, Vinyals O, Dahl GE. Neural message passing for quantum chemistry. In *International conference on machine learning*. PMLR. 2017;1263–1272.
141. Zhang Y, Yang Q. A survey on multi-task learning. *IEEE transactions on knowledge and data engineering*. 2021;34(12):5586–609.
142. Sanchez-Cortes D, Aran O, Mast MS, Gatica-Perez D. A Non-verbal Behavior Approach to Identify Emergent Leaders in Small Groups. *IEEE Transactions on Multimedia*. 2012;14(3):816–32.
143. van Roosmalen TM. The development of a questionnaire on the subjective experience of teamwork, based on Salas, Sims and Burke's "the big five of teamwork" and Hackman's understanding of team effectiveness. Master's thesis. Norges teknisk-naturvitenskapelige universitet, Fakultet for ;2012.
144. Lord RG, Foti RJ, De Vader CL. A test of leadership categorization theory: Internal structure, information processing, and leadership perceptions. *Organizational behavior and human performance*. 1984;34(3):343–78.
145. Koenigs RJ, Hare SE, Hare AP. SYMLOG reliability and validity. San Diego: Symlog Consulting Group; 2002.
146. Radford A, Kim JW, Xu T, Brockman G, McLeavey C, Sutskever I. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*. PMLR. 2023;28492–28518.
147. Eyben F, Scherer KR, Schuller BW, Sundberg J, André E, Busso C, et al. The Geneva minimalistic acoustic parameter set (GeMAPS) for voice research and affective computing. *IEEE transactions on affective computing*. 2015;7(2):190–202.
148. Eyben F, Wöllmer M, Schuller B. Opensmile: the munich versatile and fast open-source audio feature extractor. In: *Proceedings of the 18th ACM international conference on Multimedia*. 2010;1459–1462.
149. Simonyan K, Vedaldi A, Zisserman A. Deep inside convolutional networks: Visualising image classification models and saliency maps; 2013. arXiv preprint [arXiv:1312.6034](https://arxiv.org/abs/1312.6034).
150. Qu Z, Gomm D, Färber M. GreeDy and CoDy: Counterfactual Explainers for Dynamic Graphs; 2024. arXiv preprint [arXiv:2403.16846](https://arxiv.org/abs/2403.16846).