Predicting the Evolution of Depressive Symptoms Using Spatiotemporal Graph Neural Networks

Gustavo E. Cavalcante[1], **André L. Vignatti**[1]

[1]Department of Computer Science – Universidade Federal do Paraná (UFPR)
Curitiba – PR – Brazil

{getc21,vignatti}@inf.ufpr.br

Abstract. *Network psychopathology has emerged as an innovative paradigm for understanding mental disorders, modeling them as dynamic systems of interacting symptoms. Previous studies have predominantly focused on cross-sectional analyzes of symptom networks. In contrast, this study explores the temporal dimension through the application of Spatio-Temporal Graph Neural Networks (STGNNs) to predict the symptomatic evolution of patients with depressive symptoms. To this end, longitudinal data from the Experience Sampling Method (ESM) of 129 participants were used, with the implementation and comparison of three GNN architectures: Graph Convolutional Network (GCN), Graph Attention Network (GAT) and TransformerConv. These architectures were combined with a temporal layer of GRU in a previously established population network. The results show that TransformerConv achieved the best overall performance ($R^2 = 0.6126$), demonstrating a statistically significant result and maintaining stability at different depths and cross-validation folds. This study establishes a methodological basis for the application of STGNNs in psychopathology, offering new tools to predict symptomatic trajectories.*

1. Introduction

According to projections from the World Health Organization (WHO), depression is expected to become the leading cause of the global burden of diseases by 2030 [Lépine 2011]. In response to this challenge, emerging perspectives [Bringmann et al. 2022, Borsboom 2017] suggest that mental disorders – including depression – should not be attributed to a single latent cause. Instead, they are better understood as complex systems of symptoms that dynamically interact with one another, thereby sustaining the psychopathological state. In this context, these interactions can be modeled as a graph, giving rise to the network approach to psychopathology.

Similarly, recent advances in machine learning, particularly in the field of Graph Neural Networks (GNNs), have enabled the expressive modeling of data structured as graphs. Among these approaches, Spatio-Temporal Graph Neural Networks (STGNNs) stand out for their ability to integrate topological and temporal information in dynamic systems [Wu et al. 2020]. This characteristic makes their application particularly suitable for longitudinal symptom networks, where the relational structure between variables is relatively stable, while the states of the nodes evolve over time. Although there are applications of GNNs to cross-sectional data in psychiatry [Choi et al. 2021] and STGNNs to mental health data in university students [Harit et al. 2024], to date, no studies have

been identified that use STGNNs to predict the temporal evolution of symptom networks in mental disorders, highlighting a significant gap in the field of mental health.

The central hypothesis of this study is that Spatio-Temporal Graph Neural Networks (STGNNs) can effectively identify dynamic patterns within mental disorder symptom networks, thereby enhancing the prediction of their temporal evolution. Specifically, by modeling depressive symptoms and their interactions as a sequence of graph snapshots, we hypothesize that we can predict future states – identifying whether specific symptoms will intensify, diminish, or resolve. To test this hypothesis, our study was structured into two main stages. In the first stage, we applied STGNN models to predict the future state of symptoms based on previous time windows, using longitudinal data organized as time series on a pre-defined population-level network. In the second stage, we evaluated the performance of these models under different architectural configurations and prediction metrics to assess their precision, robustness, and potential clinical applicability.

The population-level network serving as the spatial structure for the models was derived from Multilevel Vector Autoregressive (ML-VAR) analysis [Bringmann et al. 2013]. In this framework, ML-VAR allows for the estimation of directional temporal relationships between symptoms at the population level, separating within-person and between-person effects. This resulted in an average network representing dynamic symptom dependencies over time, which serves as the underlying graph where nodes represent symptoms and edges encode their estimated temporal interactions.

To operationalize the proposed objectives, three models were developed based on distinct architectures: Graph Convolutional Network (GCN) [Kipf and Welling 2016], Graph Attention Network (GAT) [Veličković et al. 2018], and TransformerConv [Shi et al. 2021]. The architectures were adapted for processing longitudinal data by combining graph convolutions, responsible for capturing spatial dependencies between symptoms, with Gated Recurrent Unit (GRU) layers, designed to model temporal dependencies. The models were trained using data from the Experience Sampling Method (ESM) of 129 participants with depressive symptoms [Bringmann et al. 2013]. Training was conducted using cross-validation and performance was evaluated using multiple metrics (MAE, RMSE, and R^2), both in aggregate and individually for each symptom. In addition, we investigated the impact of architectural depth, ranging from one to four layers, on the predictive performance of the models.

The results indicated that the TransformerConv model demonstrated the best overall performance, significantly exceeding the other approaches and exhibiting high stability. In addition, it was observed that there was heterogeneity in the predictive capacity between symptoms, some being consistently more predictable than others. The GAT model exhibited significant performance degradation in deeper architectures, while the GCN maintained acceptable performance with two and three layers.

Thus, the relevance of this research lies in the integration of two prominent fields: network psychiatry, which offers a new paradigm for understanding mental disorders, and GNNs, which represent the state of the art in machine learning for graph-structured data. By combining the structure of symptom networks with their temporal dynamics, this study contributes to the advancement of computational approaches aimed at modeling and predicting psychopathological processes.

2. Related Works

Several studies have explored temporal approaches using GNNs. An example is the work by Jin et al. [Jin et al. 2024], which discusses different application contexts for GNNs in time series (GNN4TS), covering tasks such as forecasting, anomaly detection, imputation, and classification. The paper also reviews relevant work demonstrating the applicability of these techniques in diverse scenarios. Despite this breadth, none of the mentioned applications is directly focused on the study of networks of psychiatric disorders. The work of Rozemberczki et al. [Rozemberczki et al. 2021] organized various STGNN models from the literature into a Python library. However, most of these models are geared toward urban computing problems. We conducted preliminary tests on some of these models which did not yield satisfactory results for the data in this study, but they may be of interest for future research involving new datasets.

In the field of mental health, the review conducted by Ma et al. [Ma et al. 2025] provides a broad overview of GNN applications, highlighting their potential in areas such as elucidating disease mechanisms, diagnosing and classifying mental disorders, risk prediction and intervention, personalized medication and treatment outcome evaluation, as well as community-level prevention strategies. However, most of these investigations focus on genomic or neuroimaging data, with a still scarce exploration in the specific context of symptom networks. This limitation can also be seen in the work of Harit et al. [Harit et al. 2024] where STGNN is applied to predict stress levels and depression scores, but applied in a context where the network nodes are university students and the symptoms are features for these nodes, unlike our work where the network is formed by the symptoms.

Some studies are beginning to advance in this direction. For example, Choi et al. [Choi et al. 2021] developed a Graph Isomorphism Network (GIN) model to predict acute suicidal ideation in young adults, using multidimensional mental health questionnaires and achieving promising results in terms of sensitivity and specificity. Complementing this, Yang et al. [Yang et al. 2024] applied GATs to map and predict hidden relationships between psychiatric variables (such as hallucinations and medication adherence) and metabolic variables (such as body mass index, triglycerides and blood pressure). These studies reinforce the potential of GNNs in mental health and bring graph modeling closer to the study of symptoms. However, they still lack a temporal component for predicting the dynamics of symptoms, which is the focus of our approach in this study.

3. Methodology

3.1. Dataset

The dataset used in this study has previously been published [Bringmann et al. 2013]. As part of an ESM study, it followed 129 participants with residual depressive symptoms over 12 days, with the first six days constituting the baseline period. The subsequent six days occurred after an interval of two to three months, at which point participants were randomly assigned to one of two groups: a treatment group (63 participants receiving therapy, mean age = 44.6, StdDev = 9.7, 79% female) and a control group (66 participants on a waiting list, mean age = 43.2, StdDev = 9.5, 73% female). Daily, participants were randomly prompted by a pager during each of ten 90-minute blocks between 7:30 AM and 10:30 PM. Upon notification, they were required to complete an ESM self-assessment

form. Six specific symptoms were evaluated, each using a 7-point Likert scale, ranging from 1 (strongly disapprove) to 7 (strongly approve) [Boone and Boone 2012]. The symptoms chosen were selected to capture different mood states. To represent positive mood, the symptoms "I feel cheerful" (*cheerful*) and "I feel relaxed" (*relaxed*) were chosen; to represent negative mood, "I feel fearful" (*fearful*) and "I feel sad" (*sad*) were selected. In addition, the symptoms "*worry*" and "*pleasant event*" were included because they could influence the others or reflect the patient's environmental context. Due to the longitudinal structure of the study, participants observed in two distinct periods (baseline and post-intervention) were treated as independent individuals, with their identifiers updated to prevent spurious connections between temporal phases.

Figure 1 shows the population network, defined as the average symptomatic interaction structure estimated for the set of 129 participants. What is visualized are the fixed effects of the multilevel-VAR model, that is, the mathematical expectation of the autoregressive and crossover coefficients for a hypothetical individual whose random parameters are equal to zero. Each symptom is a node (C = happy, E = pleasant event, W = worry, F = fear, S = sad, and R = relaxed), and the relationships between them are shown as weighted directed edges. These edges visually represent the strength of the connection of one symptom at a given moment to another symptom at the next instant.

To better understand how the edges of the population network are created, consider a participant named Marie. On a Tuesday morning at 9:02 AM, Marie is notified by the device and records her current state: her worry is 4 (on a scale of 1 to 7) and her sadness is 2. Ninety minutes later, at 10:32 AM, another beep. Marie responds again: her worry is now 5 and her sadness is 3. The following day, the pattern repeats: the higher the worry at one moment, the higher her sadness tends to be at the next moment. The multilevel VAR model gathers sequences, not only from Marie, but from all 129 participants, and estimates that, on average, when worry increases by 1 point at one instant, sadness increases by 0.15 points at the next instant, even after statistically controlling for all simultaneous effects of two symptoms measured at the same previous instant (such as past sadness itself, past fear, past joy). This value of 0.15 is the weight of the edge connecting W to S in Figure 1 and its thickness is proportional to this number.

A red (dashed) edge indicates a negative relationship, while a green (solid) edge indicates a positive one. The network also includes self-loops, which show how an symptom's previous state predicts its own future response. This network was created using an updated version of the original code from the base article [Bringmann et al. 2013] to reproduce the graph construction and edge weight calculation process, ensuring consistency with the previous method.

In summary, the population network, represented as a graph with fixed edge weights, serves as the spatial input. The values from the ESM questionnaire responses form time series for each patient, which act as the temporal input for the models. This is analogous to models described by Jin [Jin et al. 2023] used in urban computing scenarios, where, for example, a road represents a fixed graph as the spatial layer, and temporal traffic sensor data acts as the temporal layer.

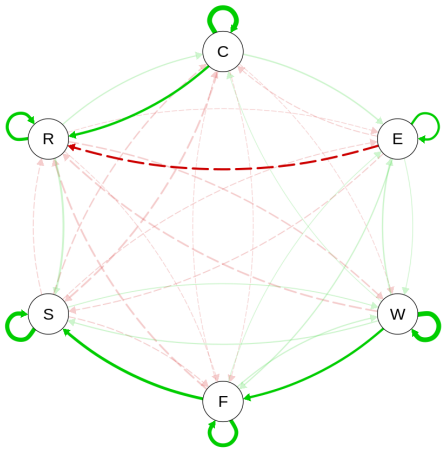

Figure 1. Population Network used as input for STGNN models. The six nodes are: C = cheerful, E = pleasant event, W = worry, F = fearful, S = sad and R = relaxed.

3.2. Pre-processing

For the treatment of missing values, temporal linear interpolation was applied to each symptom variable. In this notation, the data matrix $\mathbf{X}$ is considered, where each element $x_{i,j}$ represents the value for observation (row) i in column (symptom) j. This method estimates a missing value $x_{i,j}$ based on a linear function that connects the previous observation $x_{i_a,j}$ and the next subsequent observation $x_{i_b,j}$ available in the temporal sequence for the same individual, according to the formula:

$$\hat{x}_{i,j} = x_{i_a,j} + \frac{i - i_a}{i_b - i_a}(x_{i_b,j} - x_{i_a,j}),$$

where i identifies the moment of the missing value, and i_a and i_b represent, respectively, the previous moments (a = *previous*) and subsequent (b = *subsequent*) with known values for the same variable j. This procedure was applied column by column ($j = 1, \ldots, 6$), filling only the values for which adequate temporal anchor points existed.

For the remaining values that could not be interpolated due to the lack of temporal anchor points (e.g. missing values in the first or last measurement of an individual for a given variable), global mean imputation of the corresponding column was used. In this second phase, each missing value $x_{i,j}$ was replaced by the arithmetic mean μ_j calculated from all observed values for that specific column j throughout the entire dataset. Formally,

$$\mu_j = \frac{1}{|\mathcal{O}_j|} \sum_{k \in \mathcal{O}_j} x_{k,j},$$

where $\mathcal{O}_j$ is the set of row indices k for which a value is present in column j. Table 1 shows the data structure after the treatments mentioned above.

After imputation of missing values, all variables were global normalized using the Min-Max method. This transformation rescales each original value $x_{i,j}$ to a continuous scale in the range [0, 1], preserving the relative distribution of the data. The operation is defined by:

$$x_{\text{scaled}} = \frac{x_{i,j} - x_{\text{min},j}}{x_{\text{max},j} - x_{\text{min},j}},$$

Table 1. Processed data.

subject	day	beep	period	cheerful	pleasant event	worry	fearful	sad	relaxed
10720	1	1	0	3.975557	1.270291	2.742963	1.766794	2.356751	4.12608
10720	1	2	0	3.975557	1.270291	2.742963	1.766794	2.356751	4.12608
10720	1	3	0	3.975557	1.270291	2.742963	1.766794	2.356751	4.12608
10720	1	4	0	4.000000	3.000000	6.000000	5.000000	4.000000	2.00000
10720	1	5	0	2.000000	1.000000	5.000000	4.000000	2.000000	3.00000
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
10863	10	7	1	5.000000	1.000000	1.000000	1.000000	1.000000	4.00000
10863	10	8	1	5.000000	1.000000	1.000000	1.000000	1.000000	4.00000
10863	10	9	1	5.000000	1.000000	1.000000	1.000000	1.000000	4.00000
10863	10	10	1	5.000000	1.000000	1.000000	1.000000	1.000000	4.00000
10863	10	NaN	1	5.000000	1.000000	1.000000	1.000000	1.000000	4.00000

where $x_{\min,j}$ and $x_{\max,j}$ represent, respectively, the minimum and maximum values observed for each column j throughout the entire dataset.

Finally, we applied a sliding window method to convert each participant's time series into fixed-length sequences. This method generates input-output pairs (X_i, y_i). Here, X_i is a vector containing the participant's responses from the last five consecutive assessment moments (beeps), and y_i is the single response value at the immediately following beep (the prediction target). For example, if the time series for the intensity of symptoms is [3, 5, 4, 6, 2, 1], using a window of length $L = 5$ would generate the first pair as $X_i = [3, 5, 4, 6, 2]$ and $y_i = 1$.

3.3. Architectures, Training and Evaluation

All architectures share a modular structure consisting of a spatial layer that captures relationships between graph nodes, a temporal module that models the evolution of features over time, and a fusion mechanism that combines both representations to generate the final predictions.

The architecture based on TransformerConv employs layers of this type as its spatial mechanism, using multi-head attention (with 2 heads) to weight the influence between neighboring nodes. This is followed by a temporal attention module that applies self-attention to the temporal sequence of each individual node, and by a GRU layer that captures long-term temporal dependencies. The temporal representations (from attention) and the contextual representations (from the GRU) are combined via summation before the final linear projection. The second architecture replaces TransformerConv with GAT layers while maintaining the same temporal components and fusion mechanism. Both architectures allow configuring the number of GNN layers, incorporating Layer Normalization (LayerNorm) and ELU activations after each intermediate layer, as well as dropout (0.1) for regularization. The third architecture, based on GCN, employs simple graph convolutions without spatial attention mechanisms, coupled with a GRU layer for temporal modeling. This setup utilizes ReLU activations and excludes an explicit temporal attention module.

For all architectures, the hidden representation dimensionality was fixed at 8 channels. Training utilized Adam optimizer with a learning rate of 0.001, a batch size of 32,

and the mean squared error (MSE) loss function. Early stopping was used with a patience of 10 epochs, based on validation loss, and the maximum number of epochs was limited to 100. Validation followed a hybrid strategy, with 20% of the data permanently held as a test set, and the remaining 80% underwent stratified k-fold cross-validation (k=5).

For the final evaluation, we used multiple metrics: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and coefficient of determination (R^2), given by the following equations:

$$\text{MAE} = \frac{\sum_{i=0}^{N-1} |y_i - \hat{y}_i|}{N}, \quad \text{RMSE} = \sqrt{\frac{\sum_{i=0}^{N-1} (y_i - \hat{y}_i)^2}{N}}, \quad \text{R}^2 = 1 - \frac{\sum_{i=1}^{N} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{N} (y_i - \bar{y})^2},$$

where N is the number of predictions (i.e., the total number of test pairs (X_t, y_t) across all patients), y_i represents the true value of the target symptom at a future time step for the i-th prediction, $\hat{y}_i$ is the predicted value for that same symptom and time step, generated by the model and $\bar{y}$ is the mean of all true values y_i in the test set. The code and dataset are available on our public GitHub repository[1].

4. Results

The results are reported in two formats: aggregated (with R^2, MAE and RMSE) and individually for each symptomatic variable (with R^2 and MAE). Each model was evaluated with varying depths (1 to 4 GNN layers) to investigate the impact of network depth on predictive capacity and to mitigate over-smoothing effects [Li et al. 2018]. Table 2 presents the aggregated performance metrics (R^2, MAE, and RMSE) for the three STGNN architectures evaluated, considering different depths (1 to 4 GNN layers), as well as the overall average. TransformerConv demonstrated consistent superiority across all metrics, achieving an average R^2 of 0.6216 (±0.0018), an average MAE of 0.5313 (±0.0023), and an average RMSE of 0.9024 (±0.0020). GCN showed intermediate performance (R^2 = 0.3580 ± 0.2308), while GAT obtained the worst aggregate performance (R^2 = 0.0736 ± 0.6238), accompanied by high variability between configurations.

An analysis of performance variability, measured by the standard deviation of metrics across different depths, revealed significant differences between architectures (Table 2). TransformerConv exhibited exceptional stability, with an R^2 standard deviation of only 0.0018, in contrast to the high variability observed for GAT (StdDev = 0.6238) and the moderate variability of GCN (StdDev = 0.2308). This pattern of low variability for TransformerConv was also maintained for MAE (StdDev = 0.0023) and RMSE (StdDev = 0.0020), indicating robustness with respect to network depth. Analysis of the number of epochs until convergence, defined by early stopping with a patience of 10 epochs, revealed distinct training dynamics (Table 2). GCN required, on average, the highest number of epochs (74.8 ± 12.0), suggesting a slower convergence process. GAT exhibited a lower average (51.7 ± 28.9), yet with extremely early stopping for the four-layer configuration (15.4 epochs), indicating a rapid deterioration of the validation loss. TransformerConv displayed intermediate behavior (68.4 ± 17.1), with a gradual reduction in the number of epochs as depth increased, suggesting greater learning efficiency in deeper architectures.

[1]https://github.com/GustavoEmanuel901/Predicting-the-Evolution-of-Depressive-Symptoms-Using-Spatiotemporal-Graph-Neural-Networks

Table 2. Aggregate performance metrics for GCN, GAT, and TransformerConv architectures at different depths. Epochs refer to the average number of training epochs up to early stopping in the five folds of cross-validation.

Architecture	Layers	R^2	MAE	RMSE	Epoch
GCN	1	0.0211	1.1031	1.4641	87.4
	2	0.5287	0.6806	0.9931	59.2
	3	0.4823	0.7422	1.0361	72.8
	4	0.4001	0.8327	1.1229	79.8
	Mean	0.3580	0.8396	1.1540	74.8
	StdDev	0.2308	0.1864	0.2136	11.99
GAT	1	0.0820	1.0816	1.3907	84.4
	2	0.5547	0.6390	0.9798	60.6
	3	0.4675	0.7597	1.0789	46.2
	4	-0.8099	1.6645	1.9455	15.4
	Mean	0.0736	1,0362	1,3487	51.65
	StdDev	0.6238	0.4586	0.4347	28.85
TransformerConv	1	0.6194	0.5331	0.9050	83.0
	2	0.6227	0.5279	0.9014	82.2
	3	0.6235	0.5322	0.9003	60.0
	4	0.6211	0.5320	0.9031	48.4
	Mean	0.6216	0.5313	0.9024	68.4
	StdDev	0.0018	0.0023	0.0020	17.07

Finally, varying the number of GNN layers (1 to 4) revealed contrasting behaviors among the architectures (Figure 2). TransformerConv maintained high and stable performance across all depths, with minimal variation in R^2 (a difference of only 0.0041 between 1 and 4 layers). In contrast, GCN exhibited non-monotonic behavior, reaching peak performance with two layers (R^2 = 0.5287), followed by progressive degradation. The GAT showed a pattern similar to GCN between 1 and 3 layers, but suffered a performance collapse with four layers (R^2 = -0.8099), indicating severe overfitting and a failure to generalize.

4.1. Results by Symptoms

The analysis by symptom revealed consistent performance differences among the evaluated architectures. Table 3 summarizes the average performance metrics (R^2 and MAE) for each symptom, considering variations in network depth (1 to 4 layers).

In general, TransformerConv demonstrated superior and more stable performance for all symptoms, systematically outperforming GCN and GAT. For 'cheerful', TransformerConv achieved $R^2 = 0.6436 \pm 0.0001$, a performance significantly higher than GCN (0.5541) and GAT (0.3207), also distinguished for its virtually negligible variability. A similar pattern was observed for 'pleasant event', a particularly challenging symptom: while GCN and GAT exhibited performance close to or below zero (R^2 = -0.0106 and -0.6231, respectively), TransformerConv maintained consistent predictive capacity (R^2 = 0.4863 ± 0.0026). For anxiety-related symptoms, TransformerConv demonstrated particularly high performance. For 'fearful', it achieved the highest R^2 among all variables analyzed (0.6821), along with the lowest MAE (0.3472), indicating high predictive precision. Furthermore, this architecture also outperformed GCN and GAT (R^2 = 0.6550),

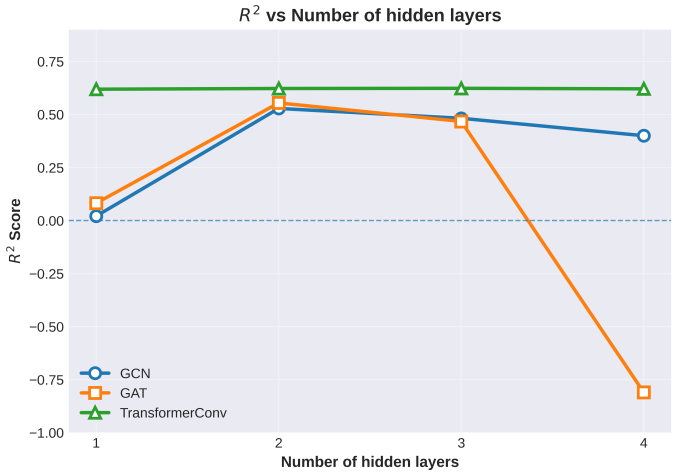

Figure 2. Coefficient of determination (R^2) as a function of the number of GNN layers for the three architectures.

while maintaining low variability at different depths. Similar results were observed for 'sad', where TransformerConv (R^2 = 0.6511) demonstrated substantially superior performance compared to the other architectures, in contrast to GCN, which exhibited a reduced R^2 (0.0737) and high variability (StdDev = 0.7272). For 'relaxed', all architectures showed relatively good performance, although TransformerConv maintained a slight advantage (R^2 = 0.6117) over GCN (0.6023).

Figure 3 illustrates the impact of the number of layers on performance (R^2) for each symptom. Distinct patterns emerged among the architectures: GCN exhibited non-monotonic behavior, with peak performance at two layers for most symptoms (5 out of 6), the only exception being 'fearful', which achieved its best result with one layer. GAT displayed a pattern similar to GCN between one and three layers but suffered a performance collapse with four layers for all symptoms, particularly 'pleasant event' (R^2 = -2.3248) and 'fearful' (R^2 = -1.7138), indicating architectural instability at greater depths. In contrast, TransformerConv maintained notably stable performance across all depths, with minimal fluctuations in R^2 (e.g., a maximum variation of only 0.0036 for 'sad').

Analysis of variability, measured by the standard deviation of R^2 scores between runs, reinforces the robustness of the TransformerConv architecture. Its average StdDev of R^2 between symptoms was remarkably low (0.0023), indicating high predictive consistency regardless of the model depth. In contrast, GCN showed moderate to high variability (average StdDev of R^2 = 0.260), and GAT demonstrated the highest instability (average StdDev of R^2 = 0.662). In addition to R^2, the MAE confirmed TransformerConv's superiority, as it achieved the lowest mean absolute errors for all symptoms, with an overall average of 0.5313. Its performance for 'fearful' (MAE = 0.3472) is particularly noteworthy. In contrast, GAT exhibited the highest MAEs in most cases, especially for 'pleasant event' (1.6248) and 'fearful' (0.9606).

Taken together, the results indicate that TransformerConv not only outperforms the GCN and GAT architectures in terms of absolute performance but also demonstrates greater stability and consistency across symptoms. Although GCN and GAT exhibited wide performance variation between variables (GCN: R^2 from -0.0106 to 0.6023; GAT:

Table 3. Average performance (R^2 and MAE) of evaluated architectures by symptom. Values in bold indicate the best performance for each symptom.

Symptom	Architecture	R^2 (Mean ± StdDev)	MAE (Mean ± StdDev)	Best Configuration
cheerful	GCN	0.5541 ± 0.0679	0.6991 ± 0.0854	2 layers
	GAT	0.3207 ± 0.3369	0.8736 ± 0.3020	2 layers
	TransformerConv	**0.6436 ± 0.0001**	**0.5283 ± 0.0025**	4 layers
pleasant event	GCN	-0.0106 ± 0.4918	1.2785 ± 0.4202	2 layers
	GAT	-0.6231 ± 1.1927	1.6248 ± 0.7439	2 layers
	TransformerConv	**0.4863 ± 0.0026**	**0.6920 ± 0.0077**	4 layers
worry	GCN	0.5721 ± 0.1732	0.7212 ± 0.1839	2 layers
	GAT	0.3696 ± 0.2996	0.9706 ± 0.3600	2 layers
	TransformerConv	**0.6550 ± 0.0018**	**0.5714 ± 0.0051**	3 layers
fearful	GCN	0.3573 ± 0.0679	0.6759 ± 0.0420	1 layer
	GAT	-0.1947 ± 1.0903	0.9606 ± 0.6279	2 layers
	TransformerConv	**0.6821 ± 0.0026**	**0.3472 ± 0.0071**	2 layers
sad	GCN	0.0737 ± 0.7272	1.0443 ± 0.4670	2 layers
	GAT	0.2746 ± 0.4845	0.8766 ± 0.4299	2 layers
	TransformerConv	**0.6511 ± 0.0036**	**0.4899 ± 0.0032**	2 layers
relaxed	GCN	0.6023 ± 0.0361	0.6190 ± 0.0511	2 layers
	GAT	0.2942 ± 0.3575	0.9105 ± 0.3101	2 layers
	TransformerConv	**0.6117 ± 0.0032**	**0.5586 ± 0.0069**	2 layers

-0.6231 to 0.3696), TransformerConv maintained consistently high R^2 values (0.4863 to 0.6821) with low variability between symptoms (StdDev = 0.076).

5. Conclusion and Future Work

Our results demonstrate the feasibility and effectiveness of using Spatio-Temporal Graph Neural Networks (STGNNs) to predict depressive symptom evolution from longitudinal data. Among the evaluated architectures, the TransformerConv-based model stood out, showing superior performance (average $R^2 = 0.6216$) and remarkable stability (R^2 StdDev = 0.0018), significantly surpassing GCN and GAT. This success stems from its integration of multi-head self-attention mechanisms with spatial aggregation, enabling the learning of more expressive representations of symptom interdependencies over time. While GCN and GAT performed less effectively, they remain relevant. GCN was found to be highly sensitive to the choice of hidden channel count, suggesting that refined hyperparameter optimization could yield improvements. Similarly, GAT's instability, especially in deeper architectures, aligns with the literature [Veličković et al. 2018], highlighting the need for careful configuration. Symptom-level analysis revealed distinct predictability patterns. Anxiety-related symptoms, such as 'fearful' ($R^2 = 0.6821$) and 'worry' ($R^2 = 0.6550$), were more predictable than positive experiences like 'pleasant event' ($R^2 = 0.4863$). This suggests that different symptomatological dimensions exhibit distinct temporal dynamics, possibly reflecting greater stability and regularity in anxiety symptoms compared to pleasant events, which are often more contextual and episodic.

Despite promising results, this study has limitations. In particular, the use of a static population-level network as the spatial input prevents the capture of potential changes in symptom interactions over time or in response to treatment. Investigating STGNNs with dynamic graphs is therefore a key direction for future work, especially given their success in other domains [Wang et al. 2024].

Future work should apply these models to larger datasets with more participants

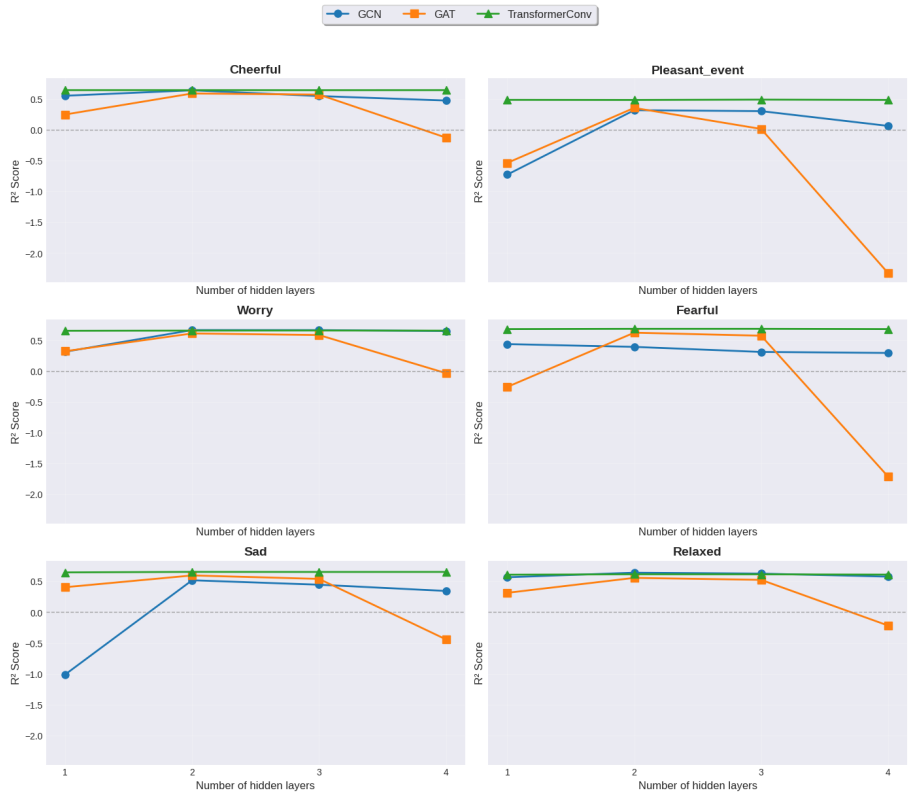

Figure 3. Performance (R^2) by number of layers for each symptom across the three architectures.

and longer time series, enabling the exploration of deeper architectures. Incorporating alternative temporal layers, such as LSTM or more sophisticated temporal attention mechanisms, could enhance predictive capacity and enable multi-step forecasting. Preliminary experiments with larger time windows ($L > 5$) and longer forecast horizons (i.e., a target y_i containing multiple temporal information point) yielded no significant gains, indicating the need for larger data volumes or improved regularization strategies. Furthermore, comparing these results with purely temporal models (e.g., GRU or LSTM) and conducting systematic hyperparameter optimization represent natural extensions of this work.

In summary, this work constitutes a relevant step in applying STGNNs to psychopathological dynamics. The results demonstrate the technical feasibility of this approach and its potential to enhance understanding, prediction, and, in the future, support personalized clinical interventions in mental health.

References

Boone, H. N. and Boone, D. A. (2012). Analyzing likert data. *Journal of Extension*.

Borsboom, D. (2017). A network theory of mental disorders. *World psychiatry*.

Bringmann, L. F., Albers, C., Bockting, C., Borsboom, D., Ceulemans, E., Cramer, A., Epskamp, S., Eronen, M. I., Hamaker, E., Kuppens, P., Lutz, W., McNally, R. J., Molenaar, P., Tio, P., Voelkle, M. C., and Wichers, M. (2022). Psychopathological networks: Theory, methods and practice. *Behaviour Research and Therapy*.

Bringmann, L. F., Vissers, N., Wichers, M., Geschwind, N., Kuppens, P., Peeters, F., Borsboom, D., and Tuerlinckx, F. (2013). A network approach to psychopathology: New insights into clinical longitudinal data. *Plos One*.

Choi, K. S., Kim, S., Kim, B.-H., Jeon, H. J., Kim, J.-H., Jang, J. H., and Jeong, B. (2021). Deep graph neural network-based prediction of acute suicidal ideation in young adults. *Nature*.

Harit, A., Sun, Z., Yu, J., and Moubayed, N. A. (2024). Monitoring behavioral changes using spatiotemporal graphs: A case study on the studentlife dataset. In *NeurIPS 2024 Workshop on Behavioral Machine Learning*.

Jin, G., Liang, Y., Fang, Y., Huang, J., Zhang, J., and Zheng, Y. (2023). Spatio-temporal graph neural networks for predictive learning in urban computing: A survey. *IEEE transactions on knowledge and data engineering*.

Jin, M., Koh, H. Y., Wen, Q., Zambon, D., Alippi, C., Webb, G. I., King, I., and Pan, S. (2024). A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection. *IEEE Trans. on Pattern Analysis and Mach. Int.*

Kipf, T. N. and Welling, M. (2016). Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

Li, Q., Han, Z., and Wu, X.-M. (2018). Deeper insights into graph convolutional networks for semi-supervised learning. In *Proc. of the AAAI conference on artificial intelligence*.

Lépine, J.-P. (2011). The increasing burden of depression. *Neuropsychiatric Disease and Treatment*.

Ma, W., Su, Z., Chen, Q., Zhai, H., Jiang, J., and Shi, H. (2025). Advancing mental health research with graph neural networks: A comprehensive survey. In *2025 IEEE 41st International Conference on Data Engineering Workshops*.

Rozemberczki, B., Scherer, P., He, Y., Panagopoulos, G., Riedel, A., Astefanoaei, M., Kiss, O., Beres, F., Lopez, G., Collignon, N., and Sarkar, R. (2021). PyTorch Geometric Temporal: Spatiotemporal Signal Processing with Neural Machine Learning Models. In *Proc. of the 30th ACM Int. Conf. on Inf. and Knowledge Management*.

Shi, Y., Huang, Z., Feng, S., Zhong, H., Wang, W., and Sun, Y. (2021). Masked label prediction: Unified message passing model for semi-supervised classification. In *Proc. of the 30th International Joint Conference on Artificial Intelligence (IJCAI-21)*.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. (2018). Graph attention networks. In *6th Int. Conf. on Learning Representations (ICLR 2018)*.

Wang, J., Sun, Z., Yuan, C., Li, W., Liu, A., Wei, Z., and Yin, B. (2024). Dynamic graphs attention for ocean variable forecasting. *Eng. App. of Artificial Intelligence*.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Yu, P. S. (2020). A comprehensive survey on graph neural networks. *IEEE trans. on neural net. and learning systems*.

Yang, H., Zhu, D., Liu, Y., Xu, Z., Liu, Z., Zhang, W., and Cai, J. (2024). Employing graph attention networks to decode psycho-metabolic interactions in schizophrenia. *Psychiatry Research*.