UET-PsyWar@eRisk 2026: Localized Context Encoding and Memory-Augmented Graph Networks for Early Detection of Depression

Notebook for the eRisk Task 2 Lab at CLEF 2026

Duc-Luong Tran[1,†], Phuong-Anh Chu[1,†], Hoang-Dat Do[1,†], Tu-Phuong Mai[1], Duy-Cat Can[1,2,3] and Hoang-Quynh Le[1,*]

[1]VNU University of Engineering and Technology, 144 Xuan Thuy, Cau Giay, Hanoi, Vietnam
[2]Platform of Bioinformatics, Lausanne University Hospital, Switzerland
[3]Faculty of Biology and Medicine, University of Lausanne, Switzerland

Abstract

This working note describes our participation as team UET-PsyWar in Task 2 of the CLEF eRisk 2026 Lab, which focuses on the early detection of depression within multi-user conversations. We propose two graph-based approaches that differ in the encoding scope of conversational context. The static pipeline constructs conversational graphs from each discussion thread and applies a Graph Attention Network (GAT) for local context encoding, followed by a Time-sensitive LSTM for temporal modeling. The dynamic pipeline maintains a persistent memory state per user, updated at each new interaction, so that when the GAT aggregates context, it incorporates not only the current thread content but also each user's accumulated interaction history across all conversations. We submitted five runs with different configurations of these two base frameworks. Our best run achieves an F_1 of 0.78, ranking 6th among 17 participating teams, with the dynamic pipeline consistently outperforming the static one.

Keywords

Early Depression Detection, Conversational Graphs, Graph Attention Networks, Time-sensitive LSTM

1. Introduction

Depression is one of the most prevalent mental health disorders worldwide [1], yet it often remains undiagnosed due to the reluctance of individuals to seek professional help [2]. Social media platforms, where users freely express their thoughts and emotions, offer a valuable opportunity for passive detection of at-risk individuals [3]. Early detection is particularly important, as timely intervention can significantly improve treatment outcomes [4]. However, depression signals in social media are rarely expressed in isolation, as they emerge within conversations, shaped by surrounding replies and interactions.

In this working note, we describe our participation in Task 2 of the CLEF eRisk 2026 Lab [5], which addresses this challenge by focusing on the early detection of depression within complete, multi-participant discussion threads sourced from Reddit. This year continues the setup introduced in 2025 [6], requiring systems to analyze a target user's behavior within full conversational contexts. The eRisk 2025 testing data is made available as additional training data in the 2026 edition.

Several approaches have been proposed for this task in the previous edition. Among the top-performing teams in 2025, the HIT-SCIR team [7] addressed the lack of conversational training data by using Large Language Models (LLMs) to generate synthetic conversations, then applied MentalBERT [8] and a Transformer [9] to model inter-post relationships. The ELiRF-UPV team [10], which closely

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany
*Corresponding author.
†These authors contributed equally.
22021148@vnu.edu.vn (D. Tran); 23020324@vnu.edu.vn (P. Chu); 24020060@vnu.edu.vn (H. Do); 21020552@vnu.edu.vn (T. Mai); duy-cat.can@chuv.ch (D. Can); lhquynh@vnu.edu.vn (H. Le)

followed the top-performing system, achieved competitive results using a Linear SVM with TF-IDF features, demonstrating that traditional machine learning methods remain effective for this task when combined with appropriate feature engineering.

While these approaches achieved strong results, they process conversational context and temporal dynamics in a relatively simple manner, without explicitly modeling the interaction structure between users within discussion threads. To address this, we propose two graph-based frameworks: a static pipeline that encodes context locally within each thread, and a dynamic pipeline that extends the contextual representation with persistent per-user memory accumulated across all conversations. Specifically, the first represents each discussion thread as a graph and leverages GAT to capture contextual dependencies within the thread, followed by a Time-sensitive LSTM for temporal modeling. The second pipeline tracks a memory state for each user, incrementally updated with each new interaction, allowing the GAT to draw on the full history of each participant's conversations rather than being limited to the current thread.

2. Dataset and Task Setup

The dataset for Task 2 of the CLEF eRisk 2026 Lab is constructed by retrieving the full conversational history of each target user from Reddit, including all discussion threads they participated in over time. Each conversation preserves the complete multi-user context: the target user's posts, all other participants' contributions, and the reply structure. This setup reflects real-world scenarios where the meaning of a post depends on its surrounding dialogue.

The evaluation was structured into two phases:

- **Training Phase:** A key advantage of this year's edition is the availability of the fully contextualized eRisk 2025 testing dataset for training. While previous iterations relied on isolated posts from the 2017, 2018, and 2022 campaigns [11, 12, 13], we can now train directly on complete, multi-participant discussion threads. Table 1 summarizes the 2025 subset used for training.
- **Interactive Test Phase:** Testing was conducted in a simulated real-time environment. For each target user, the evaluation server released discussion threads chronologically, one per submission round. After processing the current thread along with all previously accumulated data, our system submitted a binary depression prediction and a confidence score before the next thread was released.

Table 1
Statistics of the contextualized dataset (eRisk 2025 subset) used for training

Metric	Count	Percentage
Target Users	**909**	**100.00%**
– Depressed (Positive Class)	102	11.22%
– Control (Negative Class)	807	88.78%
Conversational Context		
Total Discussion Threads	278,596	–
Total Embedded Comments	13,025,230	–
Per-User Statistics		
Avg. Threads per User	306.49	–

3. Proposed Method

Our methodology comprises two graph-based frameworks that differ in the scope of context captured for encoding. The static pipeline constructs conversational graphs from each discussion thread and encodes local contextual dependencies via GAT, followed by a Time-sensitive LSTM for temporal modeling.

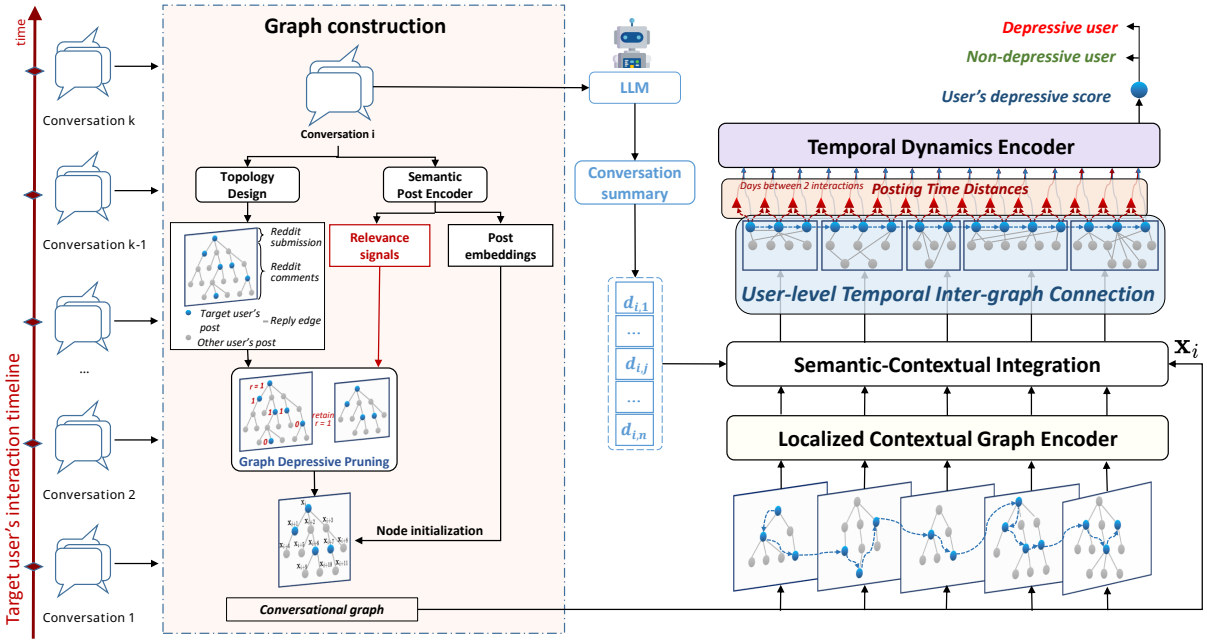

Figure 1: Overview of the static prediction pipeline with optional LLM summarization. The left panel illustrates the Graph Construction phase: conversation threads are converted into tree-structured graphs, pruned for depressive relevance, and initialized with the Semantic Post Encoder. The middle panel shows the LLM-based conversation summarization. The bottom-right panel depicts the Semantic-Contextual Integration via GAT. The top-right panel shows the Temporal Dynamics Encoder via T-LSTM, which produces the final depression score.

The dynamic pipeline extends this by maintaining a persistent memory state per user, updated at each new interaction, so that the GAT incorporates not only the current thread content but also each user's accumulated interaction history across all conversations. Both pipelines share the same Graph Construction phase (Section 3.1.1); they differ in how post embeddings are processed downstream, as described in Section 3.2. Figure 1 illustrates the static prediction pipeline along with the optional LLM-augmented summarization module.

3.1. Static Processing Model

The static prediction pipeline processes a target user's complete interaction history through three sequential phases: Graph Construction, Semantic-Contextual Integration, and Temporal Dynamics Encoding.

3.1.1. Graph Construction

Each conversation thread involving the target user is modeled as a tree-structured graph G_k, where nodes represent individual posts and edges capture reply relationships (Figure 1, left). Nodes are categorized into target user posts and other user posts to distinguish self-expressed mental states from external influence.

To initialize node representations, we design a hierarchical Semantic Post Encoder. Each sentence $s_{i,j}$ within a post p_i is first processed by an Evidence-based Sentence Encoder (ESE) built upon DepRoBERTa [14]. The ESE extracts three clinical signals: (1) BDI-II symptom relevance scores $r_{i,j} \in [0,1]^{21}$, (2) symptom severity scores $se_{i,j} \in [0,1]^{21}$, and (3) emotion category scores $e_{i,j} \in [0,1]^{7}$. The relevance encoder also produces a symptom-informed sentence embedding $\mathbf{v}^r_{i,j}$. These signals are fused into the final sentence embedding:

$$\mathbf{x}_{i,j} = ([\mathbf{v}^r_{i,j}, (\mathbf{se}_{i,j} \oplus \mathbf{e}_{i,j})\mathbf{W}_{\text{fuse}}]\mathbf{W}_{\text{proj}}) \tag{1}$$

where $\oplus$ denotes concatenation, $\mathbf{W}_{\text{fuse}} \in \mathbb{R}^{28\times d_{\text{fuse}}}$ and $\mathbf{W}_{\text{proj}} \in \mathbb{R}^{(d_{\text{bert}}+d_{\text{fuse}})\times d_{\text{bert}}}$ are projection matrices.

An LSTM then aggregates the sentence sequence in discourse order. The final hidden state serves as the post embedding:

$$\mathbf{h}_{i,j}, \mathbf{c}_{i,j} = \text{LSTM}(\mathbf{x}_{i,j}, \mathbf{h}_{i,j-1}, \mathbf{c}_{i,j-1}), \quad \mathbf{x}_i = \mathbf{h}_{i,|p_i|} \tag{2}$$

where $\mathbf{h}_{i,j}$ and $\mathbf{c}_{i,j}$ are the hidden state and cell state at sentence position j within post p_i, initialized as zero vectors. To reduce noise, we apply Graph Depressive Pruning (Figure 1, left), which retains only conversations containing at least one symptom-relevant post from the target user, preserving root-to-target paths and their immediate context while discarding irrelevant branches.

3.1.2. Semantic-Contextual Integration

Assessing depression risk solely from isolated posts is inherently limited, as depressive signals are often embedded within broader conversational dynamics. We employ a Localized Contextual Graph Encoder based on Graph Attention Networks (GAT) [15] to refine each post embedding by attending to its neighbors (Figure 1, bottom-right). For a target node p_i with embedding $\mathbf{x}_i \in \mathbb{R}^{d_{\text{bert}}}$ and its neighbors $p_j \in \mathcal{N}(i)$, the attention coefficients and contextual embedding are computed as:

$$\mathbf{x}'_i = \sigma\left(\sum_{j\in\mathcal{N}(i)} \frac{\exp\big(\text{LeakyReLU}(\mathbf{a}^\top[\mathbf{x}_i\mathbf{W} \oplus \mathbf{x}_j\mathbf{W}])\big)}{\sum_{k\in\mathcal{N}(i)} \exp\big(\text{LeakyReLU}(\mathbf{a}^\top[\mathbf{x}_i\mathbf{W} \oplus \mathbf{x}_k\mathbf{W}])\big)} \mathbf{x}_j\mathbf{W}\right), \tag{3}$$

where $\mathbf{W} \in \mathbb{R}^{d_{\text{bert}}\times d_{\text{bert}}}$ is a learnable weight matrix, $\mathbf{a} \in \mathbb{R}^{2d_{\text{bert}}}$ is a learnable attention vector, and σ denotes the ELU activation function.

However, the contextual embedding $\mathbf{x}'_i$ may be dominated by surrounding users' information. To preserve the target user's own signal, we concatenate the original and contextual embeddings:

$$\hat{\mathbf{x}}_i = (\mathbf{x}_i \oplus \mathbf{x}'_i)\mathbf{W}_{\text{sci}} + \mathbf{b}_{\text{sci}}, \tag{4}$$

where $\mathbf{W}_{\text{sci}} \in \mathbb{R}^{2d_{\text{bert}}\times d_{\text{bert}}}$ and $\mathbf{b}_{\text{sci}} \in \mathbb{R}^{d_{\text{bert}}}$ are learnable parameters. This produces a representation that retains the target user's semantic content while incorporating contextual signals from surrounding interactions.

3.1.3. Temporal Dynamics Encoder

A user's mental status at a single time point may be transient and unreliable; detecting depression requires tracking whether negative signals represent sustained patterns or isolated events. We propose the Temporal Dynamics Encoder to model the longitudinal progression of the target user's psychological state.

We chronologically order all of the target user's contextualized post embeddings $\langle \hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_{n(k)} \rangle$ from the Semantic-Contextual Integration phase and model their temporal progression using a Time-sensitive LSTM (T-LSTM) with an Adaptive Forgetting Mechanism (Figure 1, top-right).

Unlike standard LSTMs that assume equidistant time steps, our model decomposes the previous cell state $\mathbf{C}_{t-1}$ into a short-term component $\mathbf{C}^S_{t-1}$ and a long-term component $\mathbf{C}^L_{t-1}$. A learned temporal decay function $g(\Delta t)$ then discounts the short-term memory based on the actual posting time gap Δt between consecutive interactions:

$$\begin{aligned}
\mathbf{C}^S_{t-1} &= \tanh(\mathbf{C}_{t-1}\mathbf{W}_d + \mathbf{b}_d), \\
\mathbf{C}^L_{t-1} &= \mathbf{C}_{t-1} - \mathbf{C}^S_{t-1}, \\
g(\Delta_t) &= \sigma(\Delta_t\mathbf{W}_\delta + \mathbf{b}_\delta), \\
\mathbf{C}^*_{t-1} &= \mathbf{C}^L_{t-1} + (g(\Delta_t) \odot \mathbf{C}^S_{t-1}),
\end{aligned}$$

$$\mathbf{C}_t = \mathbf{f}_t \odot \mathbf{C}^*_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{C}}_t,$$
$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{C}_t),$$

where $\mathbf{W}_d \in \mathbb{R}^{d_{\text{bert}} \times d_{\text{bert}}}$ and $\mathbf{b}_d \in \mathbb{R}^{d_{\text{bert}}}$ are learnable parameters for decomposing memory, $\mathbf{W}_\delta \in \mathbb{R}^{1\times 1}$ and $\mathbf{b}_\delta \in \mathbb{R}$ parameterize the learned time decay, $\mathbf{i}_t, \mathbf{f}_t, \mathbf{o}_t$ are the standard LSTM input, forget, and output gates, $\tilde{\mathbf{C}}_t$ is the cell candidate, and $\odot$ denotes element-wise multiplication.

At each time step t, the depression score is computed as:

$$\text{score}_t = \sigma(\mathbf{h}_t \mathbf{W}_{\text{out}} + b_{\text{out}}) \in [0, 1] \tag{5}$$

where $\mathbf{W}_{\text{out}} \in \mathbb{R}^{d_h \times 1}$ and $b_{\text{out}} \in \mathbb{R}$ project the hidden state to a depression probability. A user is classified as depressed if score_k exceeds a predefined threshold τ.

3.2. Dynamic Processing Model

A key limitation of the static pipeline is that its GAT-based contextual encoding is *local*: it only considers posts within the current thread, with no knowledge of how each user has behaved in previous conversations. The dynamic pipeline addresses this by maintaining a persistent memory state for every user across all conversations. When the GAT aggregates context, it operates not only on the current post content but also on each user's accumulated interaction history. As a result, the context-enriched representation reflects both the local conversational structure and the long-term behavioral trajectory of the target user.

The dynamic pipeline reuses the same Graph Construction phase (Section 3.1.1), including the Semantic Post Encoder and Graph Depressive Pruning, to produce post embeddings. The key difference lies in how these embeddings are processed: instead of separate graph encoding and temporal modeling, the dynamic pipeline interleaves them — at each time step, user memories are updated, the GAT aggregates context enriched by these memories, and the target user's state is updated based on both the current post and the aggregated context.

3.2.1. Memory-augmented Graph Attention

Every user participating in the conversations maintains a persistent memory state that accumulates information across interactions. Let $\mathbf{s}_u^{(t)} \in \mathbb{R}^{d_m}$ denote the memory state of user u after time step t, initialized as a zero vector. These memory states are updated through two separate mechanisms depending on whether the user is the target user or a neighboring user (Section 3.2.2).

For the target user's post at time t, we compute a context-enriched representation $\mathbf{m}_i^{(t)} \in \mathbb{R}^{d_m}$ by applying a GAT [15] over its neighboring nodes. Unlike the static pipeline, where GAT operates purely on post embeddings, both the target node and its neighbors carry their respective user memory states as additional input:

$$\mathbf{m}_i^{(t)} = \text{LayerNorm}\Big(\text{GAT}\Big([\mathbf{x}_i \oplus \mathbf{s}_{\text{target}}^{(t-1)}],\ \{[\mathbf{x}_j \oplus \mathbf{s}_{u_j}^{(t)}]\}_{j\in\mathcal{N}(i)}\Big)\Big) \tag{6}$$

where, $\mathbf{x}_i$ is the target user's post embedding, $\mathbf{s}_{\text{target}}^{(t-1)}$ is the target user's memory state from the previous time step, $\mathbf{x}_j$ are the neighboring post embeddings, and $\mathbf{s}_{u_j}^{(t)}$ is the memory state of the user who authored neighboring post j. This formulation allows the attention mechanism to be informed by the full interaction history of all participants, not just the current post content.

3.2.2. Memory Update

Neighboring users. The memory state of each neighboring user is updated via a GRU as their posts arrive:

$$\mathbf{s}_{u_j}^{(t)} = \text{GRU}(\mathbf{x}_j,\ \mathbf{s}_{u_j}^{(t-1)}) \tag{7}$$

Target user. The target user's memory is updated following the same T-LSTM formulation described in Section 3.1.3. Here, the input is replaced by the contextualized embedding $\hat{\mathbf{x}}_i = [\mathbf{x}_i \oplus \mathbf{m}_i^{(t)}]$, which is the concatenation of the current post embedding and the aggregated context from the GAT layer:

$$\mathbf{h}_t, \mathbf{C}_t = \text{T-LSTM}(\hat{\mathbf{x}}_i,\ \mathbf{h}_{t-1},\ \mathbf{C}_{t-1},\ \Delta t) \tag{8}$$

The T-LSTM hidden state output becomes the updated memory state: $\mathbf{s}_{\text{target}}^{(t)} = \mathbf{h}_t$. At each time step, the depression score is computed as:

$$\text{score}_t = \sigma(\mathbf{s}_{\text{target}}^{(t)} \mathbf{W}_{\text{out}} + b_{\text{out}}) \in [0, 1] \tag{9}$$

where $\mathbf{C}_{t-1}^*$ is the adjusted cell state, Δt is the time gap between consecutive interactions, and $\mathbf{W}_{\text{out}}, b_{\text{out}}$ are the same projection parameters as in Section 3.1.3. This enables continuous risk monitoring without re-processing the entire interaction history.

3.3. LLM-augmented Summarization

As shown in the middle panel of Figure 1, we adopt a strategy inspired by the HIT-SCIR team [7], using LLMs for thread-level summarization to provide a holistic view of the entire conversation, complementing the localized signals captured by the graph structure. For each conversation, we perform a breadth-first traversal from the root post, collecting comments layer by layer to capture the discussion structure. The collected content is prompted to an LLM (Phi-4 14B [16]) to summarize what is being discussed in the thread and how the target user participates in it. The summary is then encoded into a fixed-length vector $\mathbf{z}_k$ using a pre-trained sentence encoder and concatenated with the post embedding after the GAT layer:

$$\hat{\mathbf{x}}_i^{\text{sum}} = \hat{\mathbf{x}}_i \oplus \mathbf{z}_k \tag{10}$$

where $\hat{\mathbf{x}}_i$ denotes the post-embedding after the GAT layer — specifically from Equation 4 in the static pipeline (Section 3.1.2) and from the T-LSTM aggregation step in the dynamic pipeline (Section 3.2.2). The resulting representation $\hat{\mathbf{x}}_i^{\text{sum}}$ is then used as input to the temporal encoder in place of $\hat{\mathbf{x}}_i$.

3.4. Submitted Runs

We design five runs exploring different combinations of our two core frameworks and the LLM summarization strategy.

Run 0 – Static Prediction. Static prediction pipeline (Section 3.1).

Run 1 – Static Prediction with LLM Summaries. Static prediction pipeline with LLM-augmented summarization (Section 3.1 + Section 3.3).

Run 2 – Dynamic Prediction. Dynamic prediction pipeline (Section 3.2).

Run 3 – Dynamic Prediction with LLM Summaries. Dynamic prediction pipeline with LLM-augmented summarization (Section 3.2 + Section 3.3).

Run 4 – Consensus Ensemble. Combines Run 1 and Run 2 using a consensus-based fusion strategy that penalizes disagreement between the two pipelines:

$$\text{score}_{\text{fused}} = \text{clamp}\left(\frac{s_1 + s_2}{2} - \lambda|s_1 - s_2|,\ 0,\ 1\right) \tag{11}$$

where s_1 and s_2 are the scores from Run 1 and Run 2, respectively, and $\lambda = 0.15$ is the disagreement penalty. The final decision is positive if: (1) both runs exceed $\tau = 0.55$; (2) their average exceeds τ and the lower score exceeds $\tau_{\min} = 0.35$; or (3) at least one run exceeds $\tau_{\text{high}} = 0.85$ while the other still exceeds $\tau_{\min}$.

4. Evaluation Results

4.1. Evaluation Metrics

The evaluation follows the official metrics defined by the eRisk organizers [5]. Standard classification metrics include Precision (P), Recall (R), and F-measure (F_1) computed on the positive class. For early detection, ERDE at thresholds of 5 and 50 writings (ERDE_5, ERDE_{50}) penalizes delayed alerts based on the number of posts processed before the decision is made. Median latency over true positives measures how many posts are processed before a correct positive decision:

$$\text{latencyTP} = \text{median}\{k_u : u \in U,\ d_u = g_u = 1\} \tag{12}$$

The latency-weighted F-score [17] combines classification accuracy with detection speed:

$$F_{\text{latency}} = F_1 \times \text{speed} \tag{13}$$

where $\text{speed} = 1 - \text{median}\{\text{penalty}(k_u) : u \in U,\ d_u = g_u = 1\}$, and the penalty for a decision taken after k_u writings is:

$$\text{penalty}(k_u) = -1 + \frac{2}{1 + \exp(-p \cdot (k_u - 1))} \tag{14}$$

with $p = 0.0078$ as used in the official evaluation. A decision at the first writing incurs no penalty ($\text{penalty}(1) = 0$), while detecting after hundreds of writings yields speed near 0.

4.2. Model Performance

We report the results of our five submitted runs in Table 2 and compare against other participating teams in Table 3. Our best configuration, Run 3 (Dynamic Prediction with LLM Summaries), achieves an F_1 of 0.78 and F_{latency} of 0.74, ranking 6th in F_1 among 17 participating teams [18]. Full decision-based evaluation results are available in the overview paper [6].

Table 2
Decision-based evaluation for Task 2 of our configurations.

Team	Run	P	R	F1	ERDE5	ERDE50	latencyTP	speed	$\mathbf{F}_{\text{latency}}$
UET-PsyWar	0	0.62	0.87	0.72	0.15	0.08	15.00	0.95	0.68
UET-PsyWar	1	0.57	**0.88**	0.69	0.16	0.08	15.00	0.95	0.65
UET-PsyWar	2	0.73	0.82	0.77	0.15	**0.07**	15.00	0.95	0.73
UET-PsyWar	3*	**0.76**	0.80	**0.78**	**0.14**	0.08	**14.00**	0.95	**0.74**
UET-PsyWar	4	0.60	**0.88**	0.71	0.15	**0.07**	15.00	0.95	0.68

*Best configuration is highlighted by *. Best scores in* **bold**.

Table 3
Decision-based evaluation for Task 2 (top 10 teams by best F_1).

Team	Run	P	R	F1	ERDE5	ERDE50	latencyTP	speed	$\mathbf{F}_{\text{latency}}$
HUGETIME	0	0.80	0.87	**0.83**	0.15	**0.05**	9.00	0.97	**0.81**
UNED-GELP	2	0.79	0.85	0.82	0.11	0.06	4.00	0.99	0.81
NoirByte	0	**0.88**	0.76	0.82	0.15	0.09	20.00	0.93	0.76
MindAILab	0	0.84	0.80	0.82	0.16	0.08	16.00	0.94	0.77
INSA-Lyon	0	0.76	0.85	0.80	0.16	0.07	13.00	0.95	0.76
UET-PsyWar	3	0.76	0.80	0.78	0.14	0.08	14.00	0.95	0.74
LCDAA	2	0.78	0.78	0.78	0.13	0.08	10.00	0.96	0.75
Lotu-ixa	4	0.75	0.77	0.76	0.12	0.07	6.50	0.98	0.74
erisk-cedri	0	0.73	0.77	0.75	0.10	0.07	1.50	**1.00**	0.75
VANGUARD	0	0.62	**0.90**	0.74	0.14	**0.05**	6.50	0.98	0.72

Shaded row indicates our submission.

Overall performance. Our best run achieves a competitive F_1 (0.78) with a 0.05 gap to the best system HUGETIME (0.83), and our ERDE scores are also competitive ($\text{ERDE}_5 = 0.14$, $\text{ERDE}_{50} = 0.08$). However, the primary gap lies in detection latency: our best latencyTP is 14.00, substantially higher than HUGETIME (9.00), UNED-GELP (4.00), and erisk-cedri (1.50).

Ablation analysis. Adding LLM summaries improved the dynamic pipeline (Run 3 vs. Run 2: F_1 from 0.77 to 0.78, precision from 0.73 to 0.76) but hurt the static pipeline (Run 1 vs. Run 0: F_1 from 0.72 to 0.69). This asymmetry can be explained by the difference in what each pipeline already captures. In the static pipeline, the GAT encodes local context directly from the current thread — the same content the BFS-based summary is derived from — making the summary largely redundant and potentially noisy. In the dynamic pipeline, the GAT aggregates context enriched by cross-conversation memory, which dilutes the local thread signal; the LLM summary then fills this gap by providing a focused view of the current thread structure, acting as a genuinely complementary input.

5. Conclusion

In this working note, we present our approaches for Task 2 of the CLEF eRisk 2026 Lab, focusing on the contextualized early detection of depression within multi-participant conversations sourced from Reddit. We proposed two graph-based frameworks: (i) a static prediction pipeline that represents each discussion thread as a conversational graph and encodes local contextual dependencies via GAT, followed by a Time-sensitive LSTM for temporal modeling; and (ii) a dynamic prediction pipeline that maintains a persistent memory state per user, updated at each new interaction, so that the GAT incorporates both current thread content and each user's accumulated interaction history across all conversations. We further incorporated LLM-generated thread summaries as a complementary contextual signal and designed a consensus-based fusion strategy to combine predictions from both pipelines.

Among the five submitted runs, Run 3 (Dynamic Prediction with LLM Summaries) achieved the best results with an F_1 of 0.78 and F_{latency} of 0.74, demonstrating the effectiveness of persistent memory integration over local context encoding. All systems processed the full 500 evaluation threads, confirming the scalability of the proposed framework.

For future work, we plan to reduce detection latency by designing stricter rule-based early decision mechanisms. We also aim to develop a unified architecture that more tightly couples thread-level conversational context with the user's temporal posting trajectory, rather than modeling them as separate components, so as to capture the complementary strengths of both pipelines within a single end-to-end framework.

Acknowledgments

This work was supported by the Vietnam National Foundation for Science and Technology Development (NAFOSTED) under the project "Research and Development of a Personalized Machine Learning System for Early Alzheimer's Disease Diagnosis from Adaptive Multimodal Biomarkers" (Grant No. 102.05-2025.54).

Declaration on Generative AI

During the preparation of this report, Grammarly and ChatGPT were used solely for grammar correction and improving the readability of the manuscript. No AI tools were used to generate the scientific content, analyses, or experimental results presented in this work. All methodology design, implementation, experimentation, and interpretation were carried out entirely by the authors.

References

[1] World Health Organization, Depressive disorder (depression), https://www.who.int/news-room/fact-sheets/detail/depression, 2026. Accessed: 2026-05-24.

[2] A. Aguirre Velasco, I. Silva Santa Cruz, J. Billings, M. Jimenez, S. Rowe, What are the barriers, facilitators and interventions targeting help-seeking behaviours for common mental health problems in adolescents? a systematic review, Focus 23 (2025) 98–118.

[3] W. B. Tahir, S. Khalid, S. Almutairi, M. Abohashrh, S. A. Memon, J. Khan, Depression detection in social media: A comprehensive review of machine learning and deep learning techniques, IEEE Access 13 (2025) 12789–12818.

[4] K. Habtamu, R. Birhane, M. Demissie, A. Fekadu, Interventions to improve the detection of depression in primary healthcare: systematic review, Systematic Reviews 12 (2023) 25.

[5] A. Perez, J. Parapar, X. Wang, F. Crestani, Overview of erisk 2026 early risk prediction on the internet: Symptom ranking and conversational approaches for depression and adhd, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction - 17th International Conference of the CLEF Association, CLEF 2026, Jena, Germany, September 21–24, 2026, Proceedings, volume To be published of *Lecture Notes in Computer Science*, Springer, 2026.

[6] J. Parapar, A. Perez, X. Wang, F. Crestani, erisk 2025: contextual and conversational approaches for depression challenges, in: European Conference on Information Retrieval, Springer, 2025, pp. 416–424.

[7] Y. Zi, B. Wang, Y. Zhao, B. Qin, Hit-scir@ erisk2025: exploring the potential of a learnable screening model and risk post buffer-based framework for contextualized early prediction of depression on social media, Working Notes of CLEF (2025) 9–12.

[8] S. Ji, T. Zhang, L. Ansari, J. Fu, P. Tiwari, E. Cambria, Mentalbert: publicly available pretrained language models for mental healthcare. arxiv, Preprint posted online on October 29 (2021).

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[10] A. Casamayor, V. Ahuir, A. Molina, L.-F. Hurtado, Elirf-upv at erisk 2025: New approaches to the detection and early detection of symptoms and signs of depression (2025).

[11] D. E. Losada, F. Crestani, J. Parapar, Erisk 2017: Clef lab on early risk prediction on the internet: experimental foundations, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2017, pp. 346–360.

[12] D. E. Losada, F. Crestani, J. Parapar, Overview of erisk: early risk prediction on the internet, in: International conference of the cross-language evaluation forum for european languages, Springer, 2018, pp. 343–361.

[13] J. Parapar, P. Martín-Rodilla, D. E. Losada, F. Crestani, Overview of erisk 2022: Early risk prediction on the internet, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer International Publishing, Cham, 2022, pp. 233–256. doi:10.1007/978-3-031-13643-6_18.

[14] R. Poświata, M. Perełkiewicz, Opi@ lt-edi-acl2022: Detecting signs of depression from social media text using roberta pre-trained language models, in: Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion, Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 276–282. doi:10.18653/v1/2022.ltedi-1.40.

[15] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, Graph attention networks, arXiv preprint arXiv:1710.10903 (2017).

[16] M. Abdin, J. Aneja, H. Behl, S. Bubeck, R. Eldan, S. Gunasekar, M. Harrison, R. J. Hewett, M. Javaheripi, P. Kauffmann, et al., Phi-4 technical report, arXiv preprint arXiv:2412.08905 (2024).

[17] F. Sadeque, D. Xu, S. Bethard, Measuring the latency of depression detection in social media, in: Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM '18, Association for Computing Machinery, New York, NY, USA, 2018, p. 495–503. doi:10.1145/3159652.3159725.

[18] A. Perez, J. Parapar, X. Wang, F. Crestani, Overview of erisk 2026 early risk prediction on the

internet: Symptom ranking and conversational approaches for depression and adhd (extended overview), in: Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2026), Jena, Germany, 21–24 September, 2026, volume To be published of *CEUR Workshop Proceedings*, CEUR-WS.org, 2026.