ELiRF-UPV at eRisk 2025: New Approaches to the Detection and Early Detection of Symptoms and Signs of Depression

Notebook for the eRisk Lab at CLEF 2025

Andreu Casamayor[1,*,†], Vicent Ahuir[1,†], Antonio Molina[1,†] and Lluís-Felip Hurtado[1,2,†]

[1]Valencian Research Institute for Artificial Intelligence (VRAIN), Universitat Politècnica de València, Camino de Vera s/n, 46022 Valencia. Spain

[2]ValgrAI: Valencian Graduate School and Research Network of Artificial Intelligence, Camí de Vera s/n, València, 46022, Spain

Abstract

This paper describes the approaches taken by the ELiRF-UPV team at Tasks 1 and 2 of eRisk at CLEF 2025. The tasks focus on the detection of mental health disorders in English-language social media, specifically: Search for Symptoms of Depression and Contextualized Early Detection of Depression. For Task 1, we developed lightweight adapters for pre-trained similarity Transformers, enabling efficient adaptation to limited data. These adapters used attention over reference embeddings to enhance input representations for relevance prediction across 21 depression symptoms. For Task 2, we explored three approaches: a SVM model and two Longformer-based models pre-trained in the mental health domain. One of the two Longformer-based approaches employed a straightforward fine-tuning strategy, while the other used task-adapted training enhanced by a data augmentation technique. Based on the validation results, overall performance fell short of expectations. A detailed analysis is required to identify the underlying causes.

Keywords

Longformer, Transformers, Support Vector Machine, Sentence embeddings, self-attention, Mental disorder detection

1. Introduction

Depression, clinically known as major depressive disorder (MDD), is a common and serious mental health condition that negatively affects how a person feels, thinks, and behaves. It is characterized by persistent sadness, loss of interest or pleasure in daily activities, feelings of worthlessness, and, in severe cases, thoughts of death or suicide. Depression affects people across all age groups and backgrounds. Still, it is particularly prevalent among adolescents and young adults, with women being nearly twice as likely to experience it compared to men [1].

Beyond its emotional toll, depression can severely impair physical health and social functioning, often co-occurring with other psychiatric disorders such as anxiety, substance use, or eating disorders. Despite its high prevalence and the availability of effective treatments, depression frequently goes undiagnosed or untreated due to stigma, lack of awareness, and insufficient access to mental health services [2].

As a result, analyzing social interactions has recently become a key approach for identifying the risk of depression. However, detecting depression is complex due to factors such as the quantity and quality

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

*Corresponding author.

†These authors contributed equally.

ancase3@upv.es (A. Casamayor); vahuir@dsic.upv.es (V. Ahuir); amolina@dsic.upv.es (A. Molina); lhurtado@dsic.upv.es (L. Hurtado)

https://vrain.upv.es/elirf/ (A. Casamayor); https://vrain.upv.es/elirf/ (V. Ahuir); https://vrain.upv.es/elirf/ (A. Molina); https://vrain.upv.es/elirf/ (L. Hurtado)

0009-0003-6000-3828 (A. Casamayor); 0000-0001-5636-651X (V. Ahuir); 0000-0001-6537-8803 (A. Molina); 0000-0002-1877-0455 (L. Hurtado)

of the available data. To address this challenge, CLEF eRisk [3, 4] introduced a series of tasks designed to provide high-quality data and foster the development of models for early detection.

This year, the organizers of eRisk have proposed three tasks: (1) Search for Symptoms of Depression, (2) Contextualized Early Detection of Depression, and (3) Conversational Depression Detection via Large Language Models (LLMs).

Our team has focused on the first two tasks, applying distinct approaches to address the specific challenges presented by each one:

- **Task 1**: To approach this task, we designed systems with minimal training computational cost and capable of adapting to limited data scenarios. Our approach focused on developing adapters for pre-trained similarity Transformer models. Our systems leverage the model's existing capabilities while refining its performance specifically for the relevance task tied to the 21 distinct symptoms of depression. The core addition lay in attention mechanisms within this adapter module; these utilized different sets of reference embeddings to dynamically enhance the input representation before applying regression feedforward layers.
- **Task 2**:
    1. The first approach is based on a classical machine learning algorithm: Support Vector Machines (SVM). SVMs have consistently performed in long-text classification tasks, making them a suitable choice for this context. This method is intended to serve as a baseline for assessing the effectiveness of the other models.
    2. The second approach leverages Transformer architectures [5], using a pre-trained Longformer model [6] as its foundation. Due to its capacity to handle significantly longer input sequences, Longformer is particularly well-suited for tasks that demand the assimilation of extensive contextual information. This approach employed a straightforward fine-tuning strategy for model training, where a single sample represents each user.
    3. The final approach builds upon the second by further adapting the model to the specific task of early detection. We employed task-adaptive fine-tuning to better align the training process with the objectives of the target task. To this end, we constructed a new dataset by concatenating user messages to generate multiple samples per user. This strategy exposes the model to various samples of each user's textual data with different labels.

For Task 1, we submitted five runs using the earlier approach. For Task 2, we submitted five runs: one based on the first approach, two based on the second, and two based on the final approach. In each case, the best-performing model was selected through a preliminary evaluation phase, during which multiple model configurations and datasets were tested to identify the most effective setup.

2. Description of Tasks and Datasets

This year, the competition has focused on detecting depression across different formats. For each task, a distinct dataset and a specific objective were provided.

2.1. Task 1

Task 1 ranks a collection of user-generated texts according to the relevance of depression-related symptoms. The ranking reflects how indicative each text is concerning the presence or absence of specific symptoms. For this task, the Beck Depression Inventory (BDI) questionnaire was used, which defines 21 possible symptoms of depression. A text is considered relevant if it provides information about the current state of any symptom, whether that information is positive or negative.

The dataset consisted of a TREC-formatted, sentence-tagged corpus constructed from previous eRisk data [7], in combination with the BDI questionnaire. Each sentence was annotated to facilitate the identification of content relevant to the 21 depressive symptoms defined by the BDI. The provided

training dataset contained 19 806 893 sentences, of which only 28 025 (0.14% of them) were considered informative for one or more depressive symptoms. For testing, 17 553 441 sentences were provided. Participants were expected to label all samples using the BDI questionnaire, specifying the degree of relevance for each symptom, and contribute the top 1000 most informative/relevant sentences per symptom.

2.2. Task 2

Task 2 focuses on the early detection of signs of depression through the comprehensive analysis of social media interactions, that is, considering the full context of conversations. This year, the task differs from previous editions [7], where the analysis was limited to isolated user content. Texts must be processed in the chronological order in which they were created, simulating more realistically how a system would monitor user interactions in real time on blogs, social networks, or other online platforms.

The dataset provided by the organizers has been extracted from the competitions of previous years and follows a specific format [8]. It consists of a collection of writings sourced from social media. The dataset includes both the messages written by a user and the complete interaction of that user with others for each message.

The dataset is composed of two classes: users diagnosed with depression and control users (non-depressed). Table 1 shows the distribution across the different labels.

Table 1
Distribution of samples across the 2017, 2018, and 2022 partitions of the Task 2 dataset.

	2017	2018	2022	Total
None	752	741	1302	2795
Depression	135	79	98	312
Total	887	820	1400	3107

As mentioned, the primary goal of this competition is to predict signs of depression as early as possible. To simulate realistic conditions, the organizers set up a server that sequentially delivers data packets, each containing a conversation in which the target user participates. The system must determine whether the user shows signs of depression by analyzing the current message and the history of prior messages before receiving the next packet.

3. Task 1

For this competition, we wanted to explore solutions with low computational cost during training and that could adapt to situations with limited data. Therefore, we decided to start with similarity Transformer models and add an adapter at the output that would perform the relevance task for the 21 depression symptoms. This adapter adds feedforward layers to perform the regression task and attention mechanisms regarding different sets of reference embeddings to improve the representation passed on to the dense layer intended for regression. In this work, we have explored adding reference embeddings in two aspects: (1) a centroid set extracted from the embeddings of training phrases, and (2) embeddings of question-answer pairs from the 21 questions of the Beck Depression Inventory (BDI) questionnaire for detecting depression symptoms (84 pairs in total, four answers per question).

3.1. Partitions for development

Since the provided training dataset was highly unbalanced towards non-relevant samples, we decided to reduce that imbalance during the development. If we had preserved the real distribution, the systems would have tended to score sentences as irrelevant most of the time. Therefore, we provided two irrelevant sentences for each relevant sentence. Additionally, we made a stratified partition of the relevant samples by combining the symptoms for which the sentence is informative (one sentence could

be relevant for more than one symptom). We reserved 80% of the relevant samples for training and 20% for evaluation. Table 2 shows the distribution of the development dataset.

Table 2
Distribution of samples in the development dataset for Task 1.

	Training	Evaluation
Irrelevant	44 800	11 200
Relevant	22 420	5605
Total	67 220	16 805

3.2. Adapter architecture

The regression adapter was designed to predict relevancy for the 21 symptoms of depression by transforming an input similarity embedding into a richer representation through the concatenation of three distinct embeddings. First, the original embedding served as the foundational input, capturing raw feature interactions. Second, a self-attention layer processed this embedding about a set of precomputed centroid clusters derived from clustering algorithms (e.g., k-means), generating an embedding that encoded similarity and dissimilarity metrics relative to these clusters, thereby highlighting structural patterns or group memberships. Third, another self-attention layer compared the original embedding to the collection of embeddings extracted from each question-answer pair from the BDI guide, which aimed to capture contextual relationships and semantic alignment with the guide. The adapter synthesized a comprehensive representation that integrated local feature interactions, cluster-level abstractions, and external contextual cues by concatenating these three components- original, cluster-based, and reference-aligned components.

In this work, we explored different combinations of these three types of embedding for performing the regression task. In that way, we could evaluate each piece of information's relevance to the system's ranking performance.

3.3. Adapter self-attention mechanism

The self-attention used in this work was based on the one used by the Transformer architecture [5]. The attention mechanism operated within a framework that aimed to dynamically modify the input sentence representations based on a collection of reference embeddings. In this process, the system computed first the attention information over reference embeddings (which embeddings were more relevant for the given sentence embedding), and then modulated the sentence embedding based on that information.

For computing the attention information, the list of reference embeddings was treated as a sequence of length R (number of references) and the self-attention mechanism was computed in the same way as Vaswani et al. (2017). However, *query* (q), *key* (k), and *value* (v) were modulated by the input embedding x before computing the attention scores by performing the dot product between x and each reference embedding. Once the information of x was taken into account, the attention scores were computed as usual (d is the embedding dimension):

$$AttScores = \frac{q \cdot k^T}{\sqrt{d}}.$$

Once the attention scores were calculated, it was applied a softmax normalization and dropout regularization over the attention weights, followed by the computation of the context-aware output by aggregating values v according to refined probabilities:

$$AttInfo = \text{dropout}(\text{softmax}(AttScores)) \cdot v.$$

With the attention information computed (AttInfo), we used it for modifying the original sentence embedding (x) with a contraction operation to obtain the final sentence embedding (x'):

$$x'_i = x_i \cdot \sum_j AttInfo_{ij}$$

.

3.4. Sentence embedding and reference embeddings

Since the shared task was conducted with English sentences, we employed the `all-MiniLM-L6-v2` (https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2) semantic similarity model from Sentence Transformers [9] to generate sentence embeddings. This choice was driven by two key factors: (1) its compact size significantly reduces computational overhead, making it more efficient for resource-constrained environments; and (2) in preliminary experiments, the embeddings produced by this mo which could be beneficial for our task.

We extracted reference embeddings from two sources:

1. **Clustering centroids:** Based on the training data, we extracted two different sets of centroids. With *KMeans*, we extracted 300 centroids, and with *Agglomerative Clustering*, 300 and 512 centroids. For clustering, we used the Scikit-Learn implementation [10].
2. **BDI embeddings:** For each question, we combined the question with an answer and extracted the embedding. To create each unique question-answer pair, we used the following template: `"question: answer"`.

3.5. System configurations

Table 3 summarizes the different configurations used for approaching task 1. We used one system that combined the sentence embedding with the BDI guide set of embeddings: *x-guide*. Two that combine the sentence embedding with clustering as references: *x-cluster-kmeans[300]* and *x-cluster-agglomerative[300]*. And another two that combine sentence embedding with clustering and BDI as references: *x-cluster-agglomerative[300]-guide* and *x-cluster-agglomerative[512]-guide*.

Table 3
System configurations for Task 1.

Model	Clustering Method	Number Clusters	BDI guide
x-guide	-	-	Yes
x-cluster-kmeans[300]	KMeans	300	No
x-cluster-agglomerative[300]	Agglomerative	300	No
x-cluster-agglomerative[300]-guide	Agglomerative	300	Yes
x-cluster-agglomerative[512]-guide	Agglomerative	512	Yes

All systems were trained during 20 epochs, with a linear scheduler, a linear warmup 0.1, and a learning rate of 1×10^{-4}. To select the final configuration for each model, we employed the validation partition and chose the configuration model that maximized the Average Precision (AP). AP measures how well a ranked list of retrieved documents matches the relevant items, considering both the precision (accuracy of retrieved items) and their ranking order; it is defined as follows:

$$AP = \frac{1}{R} \sum_{k=1}^{N} P(k) \times rel(k)$$

Where:

- R = Total relevant documents for the query.
- $P(k)$ = Precision at cutoff position k.
- $rel(k)$ = Binary indicator (1 if the k-th document is relevant, else 0).

4. Task 2

4.1. System architecture

For this competition, we aimed to investigate two critical factors relevant to this type of task: the size of the conversational context and the effectiveness of task-specific fine-tuning approaches.

The first factor concerns the amount of context required for accurate detection. Since each user may produce a large number of messages and conversations, input length becomes a critical consideration. One of our main objectives was to investigate the impact of contextual information on system performance, that is, to evaluate how different models perform based on the volume of context they are capable of processing. To this end, we evaluated three distinct systems: one based on classical machine learning techniques, another utilizing a RoBERTa model [11], and a third employing a Longformer model. These systems support varying input lengths, enabling us to analyze how context size influences prediction accuracy.

1. Classical machine learning approaches have no limit on the input size.
2. The selected RoBERTa model has a limit of 512 tokens in the input.
3. The selected Longformer model has a limit of 4096 tokens in the input.

The second factor explores the impact of task-specific fine-tuning compared to more general, traditional fine-tuning approaches. To evaluate this, we constructed two distinct datasets for training and evaluating the performance of the Transformer-based systems. This setup allowed us to analyze how closely aligned fine-tuning strategies affect model effectiveness on the depression detection task.

1. **Dataset 1**: We created a single sample per user by aggregating all their messages for positively and negatively labeled users. This approach ensures the dataset captures each user's context and communication patterns.
2. **Dataset 2**: If a priori evidence were available indicating the specific message in which a user begins to exhibit symptoms associated with mental health risk, we could label all preceding messages as negative and the message in question, along with all subsequent ones, as positive. This would allow us to generate additional positive samples, potentially leading to more precise model training.

We split the original dataset into training and evaluation sets for our experiments, allocating 80% of the users to training and 20% to evaluation. We ensured that the proportion of positive (Depression) and negative (None) users was maintained in both partitions. **??** shows the distribution of classes across the dataset.

Table 4
Distribution of samples in Dataset Task 2 for training and evaluation partitions.

	Training	Evaluation
None	2236	559
Depression	249	63
Total	2485	622

4.2. Classical Machine Learning Classifier Approach

To evaluate the role of contextual information, we employed a classical machine learning classifier capable of processing the entire input context. A key limitation of Transformer-based models lies in their constrained capacity to manage lengthy texts, primarily due to input size restrictions. This limitation can adversely affect classification performance, as the input may fail to encompass the complete sample, thereby risking the omission of valuable information.

Initially, we compared several classical machine learning classifiers. For this purpose, we utilized the Scikit-learn library [10], which offers a comprehensive set of tools to support experimental workflows. All classifiers were evaluated using their default hyperparameters to ensure a fair and unbiased comparison, without applying data preprocessing or preliminary analysis. Feature extraction was carried out using the TF-IDF method provided by Scikit-learn, which produces a vector representation based on the vocabulary size. The configuration employed in this setting matched the previous year's competition [12], where it demonstrated superior performance in empirical evaluations. Configuration: TF-IDF with "char_wb" analyzer and 4–5 character n-grams, extracting subword patterns within word boundaries. This helps capture prefixes, suffixes, and internal character sequences that are useful for handling spelling variations.

Table 5
The results from different classifiers in the evaluation partition in Task 2. The scores are the Macro-precision, recall, F1_score.

	Precision	Recall	F1-score
Linear SVM	**0.88**	**0.69**	**0.74**
Gradient Boosting	0.60	0.60	0.60
K-Neighboors	0.65	0.65	0.65
Random Forest	0.62	0.58	0.59

Table 5 presents the performance results of the four classical classification approaches evaluated. Among them, the Linear SVM achieved the highest scores in precision and recall, which is further reflected in a superior F1-score compared to the other methods. Beyond the choice of classification algorithm, we also investigated various preprocessing strategies and integrated additional contextual information for each message:

- **Data Preprocessing:**
  1. ***TweetTokenizer with stop word removal*:** In this approach, text was first tokenized using the TweetTokenizer, followed by eliminating stop words to reduce noise in the input.
  2. ***TweetTokenizer with cleaning and lemmatization*:** This method extends the previous approach by incorporating additional preprocessing steps, including text cleaning (removal of non-alphanumeric characters) and lemmatization of tokens to standardize word forms.
- **Sentiment Analysis:** We utilized the Transformer-based model **"lxyuan/distilbert-base-multilingual-cased-sentiments-student"** [13] to perform sentiment analysis on each message for every user. The model classifies input into three sentiment categories: positive, negative, and neutral. The resulting sentiment scores were normalized and integrated as additional features alongside the TF-IDF vectors.

We performed an exhaustive grid search using Scikit-learn's GridSearchCV utility to determine the hyperparameters for each model to improve the performance. The search explored combinations of the parameters `C`, `tol`, and `loss`, evaluating their impact on model performance. The ranges of values considered for each parameter were as follows:

- **C**: [$1e^{-1}$, 1, 10, 100]
- **tol**: [0.1, 0.01, 0.001, 0.0001]
- **loss**: [`hinge`, `squared_hinge`]

In total, we obtained four distinct configurations for experimentation. Table 6 summarizes these configurations, with the hyperparameter values for each listed in the *Hyperparameters* column.

Table 7 presents the results obtained on the evaluation partition. We observe that both the second and fourth configurations achieved identical results, concluding that thorough data cleaning is the fundamental factor influencing performance. Consequently, since both configurations yielded the

Table 6
Summary of the different configurations of the SVM classifiers.

	Data preprocess	Sentiment analysis	Hyperparameters
SVM-t2-1	1	No	'C': 10, 'loss': 'hinge', 'tol': 0.1
SVM-t2-2	2	No	'C': 100, 'loss': 'squared_hinge', 'tol': 0.1
SVM-t2-3	1	Yes	'C': 1, 'loss': 'squared_hinge', 'tol': 0.1
SVM-t2-4	2	Yes	'C': 10, 'loss': 'squared_hinge', 'tol': 0.1

same outcome, we opted for the one that demands fewer computational resources and preprocessing effort. So, the selected configuration was **SVM-t1-2**, corresponding to a Linear SVM integrated without sentiment analysis and the second preprocessing method, encompassing comprehensive data cleaning and preprocessing steps.

Table 7
Results of the different configurations of the SVM classifiers on the evaluation partition for Task 2. In bold, the best result for each metric.

	Precision	Recall	F1-score
SVM-t2-1	0.92	0.67	0.73
SVM-t2-2	0.90	**0.69**	**0.75**
SVM-t2-3	**0.93**	0.68	0.74
SVM-t2-4	0.90	**0.69**	**0.75**

4.3. Straightforward Fine-tuning Approach

Transformer architecture is widely recognized as a state-of-the-art natural language processing system. We employed two Transformer-based models for this shared task: RoBERTa and Longformer [11, 6].

- **RoBERTa**: RoBERTa is known for its robustness and strong performance in various classification tasks. However, a significant limitation of this model lies in its inability to process input sequences longer than 512 tokens. This constraint poses challenges for tasks requiring long-range contextual understanding, such as those explored in this study. As such, RoBERTa was used as a baseline model for comparison with architectures designed to handle longer inputs.
- **Longformer**: The Longformer (short for "Long-Document Transformer") was specifically developed to process long sequences more efficiently, making it better suited than conventional Transformer models like BERT or RoBERTa for tasks involving extended context. The architecture incorporates several key innovations:
  - **New attention mechanism**: Rather than employing full self-attention, Longformer uses a sliding window attention mechanism in which each token attends only to a fixed number of neighboring tokens. This design significantly reduces the computational cost.
  - **Global attention**: In addition to local attention, Longformer enables select tokens to receive global attention, allowing them to attend to all other tokens in the sequence. This hybrid mechanism enhances efficiency and the model's ability to capture long-range dependencies.

We conducted a literature review to identify a base model pre-trained on domains related to depression and mental health. According to findings by Alireza Pourkeyvan [14], the current state-of-the-art model for mental disorder detection is MentalRoBERTa [15]. MentalRoBERTa is a domain-adapted variant of the RoBERTa architecture, tailored explicitly for mental health applications. It has been pre-trained on a specialized corpus comprising texts from mental health forums, clinical notes, and general language sources, the Suicidal and Mental Health (SWMH) corpus [16]. This domain-specific pre-training allows the model to more effectively comprehend and process mental health–related language, thereby

improving its performance and relevance in related tasks. In addition to MentalRoBERTa, the authors have also developed MentalLongformer, an adaptation of the Longformer architecture, which is similarly pre-trained on mental health-related data.

The models selected for our experiments were ***AIMH/mental-roberta-large*** [17], representing the RoBERTa-based architecture, and *AIMH/mental-longformer-base-4096* [18], corresponding to the Longformer-based architecture.

Each base model was fine-tuned using **Dataset 1**, applying a straightforward fine-tuning strategy. During this process, the models were trained using the complete contextual information available for each user. Table 8 shows the configuration used in the fine-tuning process.

Table 8
Parameters for the fine-tuning process.

parameter	value
optimizer	AdamW
learning rate	5e-5
lr scheduler type	linear
weight decay	0.01
number of epochs	10
training batch size	16

Table 9 presents the results of the different models on the evaluation partition.

Table 9
RoBERTa's and Longformer's results for Task 2 on evaluation partition.

	Precision	Recall	F1-score
RoBERTa-t2	0.642	0.661	0.647
Longformer-t2	**0.693**	**0.745**	**0.705**

4.4. Task Adaptive Fine-tuning

The second objective of this study was to examine the impact of task-specific training, particularly in the context of early detection. To this end, we implemented a data augmentation strategy designed to adapt the original dataset to the requirements of an early detection setting. This strategy followed the methodology employed in previous editions of the task [12].

The augmentation process aimed to generate additional training instances for each user labeled as positive. Specifically, we sought to identify the message at which the user first exhibited signs of a mental health disorder. Once this critical message was determined, all possible concatenations of the user's message history were generated and labeled according to the following rule: message sequences occurring before the critical message were labeled as **None**, while those including or following it were labeled as **depression**.

To identify the critical message, we conducted a series of experiments in which various models were trained and evaluated to detect the onset point of disorder-indicative language within each user's message sequence.

4.4.1. Best Model for Data Augmentation

The goal of this experimental phase was to identify the model that most accurately detected the critical message—that is, the point at which a user begins to exhibit signs of a mental health disorder. To ensure consistency with the shared task guidelines, we replicated the inference procedure defined by the competition. Specifically, each model processed input batches consisting of a single message per user and was required to make predictions according to this sequential setup as early as possible.

For these experiments, the whole dataset was utilized to determine the critical message for each user labeled as positive. The model that achieved the highest performance in this task was then selected to generate the augmented data used in the early detection training phase.

The models selected for this analysis were **Longformer-t2**, **RoBERTa-t2**, and **SVM-t2-2**, as they demonstrated the highest performance during the training phase. Table 10 presents the experimentation results. Among the evaluated models, **SVM-t2-2** achieved the best performance in classifying users under an early detection framework. Consequently, this model was selected for the data augmentation process.

Table 10
Results from the competition simulation on the full dataset

	Precision	Recall	F1-score
SVM-t2-2	0.928	**0.936**	**0.934**
RoBERTa-t2	0.891	0.896	0.893
Longformer-t2	0.922	0.919	0.921

This technique resulted in a new dataset with increased training samples. Table 11 presents the two datasets generated through the data augmentation process described above, each corresponding to a different maximum input length: 512 tokens for RoBERTa and 4096 tokens for Longformer. **Dataset 2** refers to the version constrained to a maximum sequence length of 512 tokens, while **Dataset 3** corresponds to the version that allows sequences of up to 4096 tokens.

Table 11
Distribution of samples in the new datasets

	None	Depression
Original	2795 (90%)	312 (10%)
Dataset 2	79173 (98%)	1623 (2%)
Dataset 3	858242 (98%)	23254 (2%)

Table 11 presents the relationship between the classes. As can be observed, the application of data augmentation has further exacerbated the class imbalance. To address this issue, we conducted a preliminary analysis to discard information of limited utility for training. This analysis aimed to identify the users who contributed the most informative instances for each class, while disregarding those with minimal contribution. Additionally, we were particularly interested in retaining those users whose data lay near the decision boundaries, as these examples are crucial for effectively delineating class separations.

To identify the users who provide the most informative examples for each class and those located near the decision boundary, we relied on the decisions made by the *SVM-t2-2* model and the scores assigned to each user. For the *None* class, users with highly negative scores were selected, whereas for the *Depression* class, those with highly positive scores were chosen. To capture users near the decision boundary, we selected those misclassified, False Positives (FP), i.e., users incorrectly classified as positive, and conversely, False Negatives (FN). We also included users whose scores were close to zero, as they reflect uncertainty in classification and are likely to lie near the decision threshold.

With the sample filtering, we obtained a new dataset with the following class distribution: 1,039 negative users (None) and 312 positive users (Depression). Table 12 presents the new class distributions after applying data augmentation to the refined dataset. As shown, the proportion between positive and negative samples is preserved.

Table 13 reports the performance of the two models, RoBERTa-t2-t and Longformer-t2-t, which were trained on datasets obtained via the refinement procedure followed by data augmentation.

Table 12
Distribution of samples in the refined datasets

	None	Depression
Original	2795 (90%)	312 (10%)
New Dataset 2	14573 (90%)	1623 (10%)
New Dataset 3	205587 (90%)	23254 (10%)

Table 13
RoBERTa's and Longformer's results for Task 2 on evaluation partition.

	Precision	Recall	F1-score
RoBERTa-t2-t	0.780	0.705	0.734
Longformer-t2-t	**0.839**	**0.824**	**0.805**

5. Runs

5.1. Task 1

Table 14 shows the system configuration used for each submitted run and the results obtained on the evaluation partition. We observe that most of the systems achieve similar performance in AP, Recall-Precision (R-PREC), and Normalized Discounted Cumulative Gain (NDCG). However, the Run5 with all the information (input sentence embedding, cluster centroids, and BDI embeddings) achieved better in Precision at 10 (P@10) than the rest of the systems.

Table 14
Results for the 5 runs on Task 1 in the evaluation partition.

	Model	AP	R-PREC	P@10	NDCG
Run1	x-guide	0.561	0.680	0.943	**0.273**
Run2	x-cluster-kmeans[300]	**0.562**	0.679	0.976	**0.273**
Run3	x-cluster-agglomerative[300]	**0.562**	**0.682**	0.962	0.272
Run4	x-cluster-agglomerative[300]-guide	**0.562**	0.680	0.948	0.272
Run5	x-cluster-agglomerative[512]-guide	0.561	0.679	**0.981**	**0.273**

5.2. Task 2

Table 15 summarizes the selected model for each run, along with the corresponding performance on the evaluation set.

Table 15
Summary of the approaches chosen for each run. Also, the performance achieved by each system in the evaluation partition is considered.

	Task	Model	Initial Context	Precision	Recall	F1-score
Run0	2	SVM-t2-1	0	**0.900**	0.690	0.750
Run1	2	Longformer-t2-0	0	0.693	0.745	0.705
Run2	2	Longformer-t2-100	100	0.693	0.745	0.705
Run3	2	Longformer-t2-t-0	0	0.839	**0.824**	**0.805**
Run4	2	Longformer-t2-t-100	100	0.839	**0.824**	**0.805**

The rationale behind selecting these models is to validate the hypotheses of our research:

1. To assess the impact of context on prediction performance and to evaluate how different models handle long-context inputs, selecting two types of models: SVMs and Longformers.
2. To explore the effect of task-specific fine-tuning for model adaptation, not only about the models' ability to classify samples accurately, but also considering the efficiency of their inference time.

To this end, we have selected two types of LongFormer models (task-specific and straightforward) and two types of initial context constraints (0 and 100) to examine how task-specific training influences the model's decision on when to make a prediction.

5.2.1. Runs Configurations

In addition to selecting the model for each run, the classification systems required configuring additional parameters.

- For each round of the competition, we used a new sample created as the input to the classifier by combining the user's latest message with the preceding ones. We also included all replies to other users within the corresponding post.
- Each system has an initial context constraint, so we configured our system to wait until the initial context was sufficiently large. This context varied between systems and can be seen in Table 15
- The LongFormer system has a limit on tokens. When the system was full, we just returned the last prediction made.

6. Results

6.1. Task 1

Table 16 details our systems' performance in the competition. The results show that the system, which only includes the embeddings of the BDI guide, is the one that performs better, differing from the results in the evaluation partition. We observed significant differences between these results and those from evaluation (Table 14), particularly concerning overall generalization ability compared to their expected level based on internal consistency metrics like high P@10 (typically exceeding 0.9). Notably, considering that organizers manually verified our top-50 samples provided in Table 16, achieving only a P@10 score of 0.1 suggests the ranking capabilities did not transfer from evaluation to test as effectively anticipated based on internal metrics. We cannot yet identify specific reasons for this discrepancy without further analysis and research with the gold labels of the test set used to evaluate the systems.

Table 16
Results for the 5 runs on Task 1. *Highest* refers to the highest values achieved in the competition.

	Model	AP	R-PREC	P@10	NDCG
Run1	x-guide	**0.035**	**0.101**	0.100	**0.216**
Run2	x-cluster-kmeans[300]	0.032	0.095	0.081	0.206
Run3	x-cluster-agglomerative[300]	0.035	0.099	**0.110**	0.211
Run4	x-cluster-agglomerative[300]-guide	0.033	0.099	0.067	0.210
Run5	x-cluster-kmeans[512]-guide	0.032	0.097	0.100	0.209
Highest	-	0.354	0.433	0.876	0.653

6.2. Task 2

Table 17 presents the results achieved by our teams in Task 2. The structure of Table 17 is as follows: each row corresponds to an individual run, with a special row highlighting the highest values obtained in the competition. The systems were ranked based on their Macro-F1 score (last column). A total of 50 different systems (runs) participated in this task.

Table 17 presents the results, highlighting **Run 0**, corresponding to **SVM-t2-2**, a Linear SVM model, as the best-performing system. This run achieved sixth place in the competition, making us the second-best team overall. Based on these results, the following conclusions have been drawn:

Although our initial assumption was that the Longformer would outperform due to its computational power and capacity to handle large texts, the SVM ultimately achieved superior results, largely thanks

Table 17
Results for the 5 runs on Task 2. *Highest* refers to the highest values achieved in the competition. The values inside the parentheses indicate our position in the ranking.

	Model	Precision	Recall	F1-score	latencyTP
Run0	SVM-t2-2	**0.78** (1)	**0.81 (35)**	**0.79 (6)**	7.00
Run1	Longformer-t2-0	0.37	0.62	0.46	**1.00** (1)
Run2	Longformer-t2-100	0.83	0.47	0.60	8.00
Run3	Longformer-t2-t-0	0.68	0.67	0.67	7.00
Run4	Longformer-t2-t-100	0.68	0.67	0.67	7.00
Highest	-	0.78	0.100	0.85	1.00

to its ability to effectively process long-context inputs. Beyond this, it is important to highlight that the SVM relies on features extracted through TF-IDF, a technique inherently focused on capturing vocabulary frequency and distribution. This focus on specific words or n-grams may play a decisive role in detecting signs of depression, as certain linguistic patterns can serve as strong indicators. The combination of these two factors likely plays a fundamental role in enhancing detection performance. These findings suggest that classical approaches, such as SVMs, continue to hold significant value for mental health detection tasks, particularly due to their robustness in managing extensive contexts. Moreover, their efficiency and lower computational demands make them especially well-suited for deployment in resource-constrained environments.

Secondly, concerning the performance gap between the evaluation and test phases, led our team to hypothesize that the reason lies in the refinement of the corpus provided by the competition. Specifically, by removing users based on the criteria defined by the *SVM-t2-2* model, we may have inadvertently eliminated critical information necessary for training the models. As a result, while the models performed competitively during evaluation, the lack of this information during training likely caused a drop in performance when moving to test-time inference. Conversely, the users selected for training may have also introduced noise or unnecessary information into the training process. Therefore, one of the underlying causes is not only the removal of important information but the very act of removing data itself. In this particular case, it is possible that the more data the model was exposed to during training, the better it was able to generalize and adjust during the inference phase.

When examining the results obtained by the Longformer models, it becomes clear that the use of task-specific training has led to superior performance. This improvement can be attributed to the adaptation of the training process through the newly generated dataset, created using data augmentation techniques. Such training has effectively aligned the models with the specific demands of the task, allowing them to generalize more effectively during the test phase. Moreover, this approach has enhanced the overall stability of the models, enabling them to achieve a better balance between precision and recall, a critical factor for reliable performance in practical applications.

By comparing *Run-1* and *Run-3*, which both correspond to Longformer models without initial context constraints, we observe that the model trained with task-specific adaptation not only achieved superior results, but also, when examining the *LatencyTP*, demonstrates a clear tendency to wait for more contextual information before making predictions. This finding suggests that task-specific training endows the model with a learned capacity to delay decisions until sufficient evidence is gathered, rather than rushing to early conclusions. Such behavior reinforces the broader principle that, in complex prediction tasks, acting rapidly is not always advantageous; instead, making informed decisions based on richer context can lead to more accurate and reliable outcomes.

7. Conclusion

In this paper, we have presented the participation of the ELiRF-UPV team in the shared tasks of eRisk at CLEF 2025. Beyond evaluating both traditional classification models and cutting-edge Transformer architectures, our team's most innovative contribution was the application of LongFormer models

to expand the context available for decision-making. We leveraged pre-trained models specifically tailored to the mental health domain and introduced a new data augmentation and refined technique that customizes model training to the specific task at hand.

In task 1, our evaluation phase showed promising results for the designed systems. However, the ranking performance observed during evaluation did not translate consistently to the test phase. We recognize that without access to the test data, we were limited in providing deeper insights into this performance discrepancy or analyzing its specific causes on real-world tasks. Further research and analysis are necessary to pinpoint critical areas where refinement could enhance generalization capabilities.

In task 2, while the results were promising for some systems, they ultimately did not fully demonstrate the potential performance of LongFormer models for this type of task. However, the experiments did show that classical classifiers remain highly competitive, thanks to their lack of input length restrictions, faster processing, and low computational resource requirements.

For future work, three main lines of improvement are identified. On the one hand, efforts will focus on enhancing early detection so that the system requires less context to make accurate decisions; on the other hand, the application of Explainable Artificial Intelligence (XAI) techniques will be explored to gain deeper insights into the system's behavior. Finally, explore alternative data augmentation techniques to improve upon the current approach, aiming to fully leverage all the information provided by the competition, or to enhance dataset refinement in order to avoid the loss of important information.

Acknowledgments

This work is partially supported by MCIN/AEI/10.13039/501100011033, by the "European Union" and "NextGenerationEU/MRR", and by "ERDF A way of making Europe" under grants PDC2021-120846-C44 and PID2021-126061OB-C41. Partially supported by the Vicerrectorado de Investigación de la Universitat Politècnica de València PAID-01-23. It is also partially funded by the Spanish Ministerio de Universidades under the grant FPU21/05288 for university teacher training.

Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT and Grammarly in order to: Grammar and spelling check, Paraphrase, translate, and reword. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] R. C. Kessler, P. Berglund, O. Demler, R. Jin, D. Koretz, K. R. Merikangas, A. J. Rush, E. E. Walters, P. S. Wang, The epidemiology of major depressive disorder: results from the National Comorbidity Survey Replication (NCS-R), JAMA 289 (2003) 3095–3105.

[2] World Health Organization, Depression, 2021. Retrieved from https://www.who.int/news-room/fact-sheets/detail/depression.

[3] J. Parapar, A. Perez, X. Wang, F. Crestani, Overview of eRisk 2025: Early Risk Prediction on the Internet (Extended Overview), in: Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2025), Madrid, Spain, 9-12 September, 2025, volume To be published of *CEUR Workshop Proceedings*, CEUR-WS.org, 2025.

[4] J. Parapar, A. Perez, X. Wang, F. Crestani, Overview of eRisk 2025: Early Risk Prediction on the Internet, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction - 16th International Conference of the CLEF Association, CLEF 2025, Madrid, Spain, September 9-12, 2025, Proceedings, Part II, volume To be published of *Lecture Notes in Computer Science*, Springer, 2025.

[5] 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). URL: https://arxiv.org/abs/1706.03762, accessed: 2024-05-15.

[6] I. Beltagy, M. E. Peters, A. Cohan, Longformer: The Long-Document Transformer, arXiv preprint arXiv:2004.05150 (2020). URL: https://arxiv.org/abs/2004.05150.

[7] J. Parapar, P. Martín-Rodilla, D. E. Losada, F. Crestani, Overview of eRisk 2024: Early Risk Prediction on the Internet, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. 15th International Conference of the CLEF Association (CLEF 2024), volume 14959 of *Lecture Notes in Computer Science*, Springer, 2024, pp. 73–92. URL: https://doi.org/10.1007/978-3-031-71908-0_4. doi:10.1007/978-3-031-71908-0_4.

[8] D. E. Losada, F. Crestani, A Test Collection for Research on Depression and Language Use, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. 7th International Conference of the CLEF Association (CLEF 2016), volume 9822 of *Lecture Notes in Computer Science*, Springer, 2016, pp. 28–39. URL: https://doi.org/10.1007/978-3-319-44564-9_3. doi:10.1007/978-3-319-44564-9_3.

[9] N. Reimers, I. Gurevych, Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2019. URL: https://arxiv.org/abs/1908.10084.

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, É. Duchesnay, Scikit-learn: Machine Learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830. URL: https://jmlr.org/papers/v12/pedregosa11a.html.

[11] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A Robustly Optimized BERT Pretraining Approach, arXiv preprint arXiv:1907.11692 (2019). URL: https://arxiv.org/abs/1907.11692.

[12] A. Casamayor, V. Ahuir, A. Molina, L.-F. Hurtado, ELiRF-VRAIN at eRisk 2024: Using LongFormers for Early Detection of Signs of Anorexia, in: L. Cappellato, N. Ferro, D. E. Losada, H. Müller (Eds.), Working Notes of CLEF 2024 - Conference and Labs of the Evaluation Forum, volume 3740 of *CEUR Workshop Proceedings*, CEUR-WS.org, Grenoble, France, 2024, pp. 803–812. URL: https://ceur-ws.org/Vol-3740/paper-65.pdf.

[13] L. X. Yuan, distilbert-base-multilingual-cased-sentiments-student (revision 2e33845), 2023. URL: https://huggingface.co/lxyuan/distilbert-base-multilingual-cased-sentiments-student. doi:10.57967/hf/1422.

[14] A. Pourkeyvan, R. Safa, A. Sorourkhah, Harnessing the Power of Hugging Face Transformers for Predicting Mental Health Disorders in Social Networks, IEEE Access 12 (2024) 28025–28035. URL: http://dx.doi.org/10.1109/ACCESS.2024.3366653. doi:10.1109/access.2024.3366653.

[15] S. Ji, T. Zhang, L. Ansari, J. Fu, P. Tiwari, E. Cambria, MentalBERT: Publicly Available Pretrained Language Models for Mental Healthcare, in: N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, J. Odijk, S. Piperidis (Eds.), Proceedings of the Thirteenth Language Resources and Evaluation Conference, European Language Resources Association, Marseille, France, 2022, pp. 7184–7190. URL: https://aclanthology.org/2022.lrec-1.778.

[16] S. Ji, X. Li, Z. Huang, E. Cambria, Suicidal ideation and mental disorder detection with attentive relation networks, Neural Computing and Applications (2021).

[17] AIMH, MentalRoBERTa: A Robustly Optimized BERT Pretraining Approach for Mental Health, 2024. URL: https://huggingface.co/AIMH/mental-roberta-large, accessed: 2024-05-15.

[18] AIMH, MentalLongformer: A Long-Document Transformer Model for Mental Health, 2024. URL: https://huggingface.co/AIMH/mental-longformer-base-4096, accessed: 2024-05-15.