ELiRF-VRAIN at MentalRiskES 2024: Using LongFormer for Early Detection of Mental Disorders Risk

Andreu Casamayor, Vicent Ahuir, Antonio Molina* and Lluís-Felip Hurtado

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

Abstract

This paper describes the approaches taken by the ELiRF-VRAIN team at the shared tasks of MentalRiskES at IberLEF 2024 [1]. These shared tasks involved two activities focused on identifying mental illness on Spanish-language social media: detection of disorder and context detection. Our work consisted of three approaches: one approach based on a Support Vector Machine and the other two based on Transformer architecture pre-trained models, one using BERT-like models and the other using LongFormer models. In order to fine-tune our models, we used a data augmentation process on the data provided by the organization. According to the results, our approaches fit the task correctly.

Keywords

Longformer, Transformers, Support Vector Machine, Mental disorder detection

1. Introduction

A mental disorder is characterized by a clinically significant disturbance in an individual's cognition, emotional regulation, or behavior. It is usually associated with distress or impairment in important areas of functioning [2].

According to the World Health Organization (WHO), 1 in every 8 people is living with a mental disorder, with anxiety and depressive disorders the most common [3]. Although the problem is widely known, the number of people is still increasing, and discrimination against them still exists. Currently, the governments work to prevent and cure mental illness. However, the lack of human and material resources means that many people cannot receive adequate treatment or none at all. In addition to all this, early detection of mental disorders is often difficult.

In this context, detecting mental disorders risk through analyzing social media interactions has acquired great relevance in recent years. Many factors make the problem of mental disorders

detection complicated, such as availability, amount, and quality of data. Providing quality labeled data in Spanish and promoting the creation of models for this early detection is precisely the objective of the MentalRiskES shared tasks.

In the 2024 edition, the competition consisted of three tasks [4]: (1) Detection of mental disorder, (2) Context Detection, and (3) Suicidal ideation detection. Our team participated in the first two tasks.

To tackle task 1, we considered three different approaches.

1. The first approach is based on a classic machine learning algorithm: Support Vector Machines (SVM). SVMs have demonstrated adequate behavior in long text classification tasks such as this case. We consider this approach as an assessment of the performance of classical models.

2. The second approach is based on Transformers [5]. We use a pre-trained RoBERTa model [6] as a basis and then run a fine-tuning process to adjust them to the task domain. We considered two different datasets to do fine-tuning: the one provided by the organization and an expanded version of the dataset through a data augmentation process.

3. The last approach is similar to the second one; however, to capture more context, we use a pre-trained LongFormer model [7]. This way, the model is able to capture more context because of the bigger size of the input layer. We used the same dataset as in the previous approach for the fine-tuning phase.

We submitted three runs for task 1, one for each approach. The best model of each approach was chosen through a previous validation stage in which different parameters and datasets were considered.

To tackle task 2, we sent one system based on the third approach of the first task, a LongFormer-based solution. We chose that approach since it was the more promising one based on the evaluation results of task 1.

2. Description of Dataset and Tasks

The datasets delivered by the organization consisted of a message collection sent to different groups on Telegram [8]. These public groups have the characteristic of being in Spanish and related to mental illnesses. The messages were anonymized and, subsequently, labeled by ten annotators at the user level; that is, each user was labeled considering his/her messages.

Two different datasets were delivered: one for the first two tasks and a different one for the third task. The first dataset, the one with which we worked, has the following sample distribution: 20 users for trial, 465 users for train, and 400 users for test.

As stated above, the main objective of this competition is to predict mental disorders as soon as possible. To achieve realistic behavior, the organization emulated a real conversation by setting up a server that gives out packets of data containing a message for each user. The system must predict the label of each user, considering the current message and all their previous messages, before the classification system will receive the next packet. The goal is to predict each user's mental disorder, if any, as quickly as possible.

2.1. Task 1: Disorder Detection

Task 1 is a multiclass classification task whose objective is to predict if users suffer from depression, anxiety, or none disorder.

Table 1 shows the distribution among the different labels in the dataset for the first task.

	Train	Trial	Total
None	213	10	223
Depression	164	5	169
Anxiety	88	5	93
Total	465	20	485

Table 1
Distribution of samples across the Train and Trial partitions of the Task 1 dataset.

To maximize the available samples for the training process, we joined the Train and Trial partitions to train our systems; the *Total* column of Table 1 shows the final sample distribution of our training dataset.

2.2. Task 2: Context Detection

Task 2 is a two-level multiclass multilabel task: in addition to detecting the mental illness, the context or contexts in which it appears must be detected. There are 7 contexts: addiction, emergency, family, work, social, other, and none.

The label distribution in this total dataset can be seen in Table 2. It shows how the contexts of Family, Social, Other, and None are the most common.

	Addiction	Emergency	Family	Work	Social	Other	None
Depression	9	7	47	9	66	26	52
Anxiety	3	10	14	8	25	33	26
Total	12	17	61	17	91	59	78

Table 2
Distribution of samples across the Task 2 dataset.

Table 3 shows how many contexts there are per user. It can be seen that the most common situation is only one context per user.

	1 class	2 classes	3 classes	4 classes
Depression	131	30	7	1
Anxiety	71	18	4	0
Total	202	48	11	1

Table 3
Number of classes per user in Task 2 dataset.

3. System architecture and Techniques

In this kind of task, an important aspect to count on is the amount of context required to perform the detection correctly. Since each user can have many messages, the size of the input to the system must be a factor to consider. One goal of our team was to study the impact of the context in these tasks. That is, measure the capabilities of different systems depending on how much context they can manage. We selected three different systems to achieve this goal: the first based on Support Vector Machines (SVM), the second based on a RoBERTa model, and the third based on a LongFormer model. Every system evaluated has a different size for context:

- SVM has no limit in the input size; it creates a vector as long as the vocabulary size.
- The selected RoBERTa model has a limit of 512 tokens in the input.
- The selected LongFormer model has a limit of 4096 tokens in the input.

Regarding the dataset, we translated all the data into English because the Transformers base models were pre-trained using documents in this language. We used the library **EasyNMT** [9] and the model **OPUS**-MT Spanish-English [10] (https://huggingface.co/Helsinki-NLP/opus-mt-es-en). Furthermore, we created two different datasets to train and evaluate the performance of the transformer-based systems.

Dataset 1. We created only one sample per user by accumulating all his/her messages, for both positive and negative labeled users.

Dataset 2. If we had some a priori evidence of in which message a user begins to present symptoms of mental illness risk, we could label the samples from previous messages as negative, and the samples containing that message and subsequent ones as positive. In this way, we could increase the number of positive samples, in order to achieve a more precise model. This data augmentation process is explained below.

To carry out our experimentation, we divided the original dataset into two partitions: training (80% of users) and development (20% of users), maintaining the proportions of positive and negative samples in each of the partitions. Table 4 shows the distribution of samples in Dataset 1.

	Training	Development
None	178	45
Depression	134	35
Anxiety	76	17
Total	388	97

Table 4
Distribution of samples in Dataset 1 for training and development partitions.

3.1. Data Augmentation

The data augmentation process aims to create more samples per positive user. We said above that we need some evidence of the message in which a user begins to express symptoms of illness. To do this, we relied on the prediction of the SVM-based classifier. We can assume that

all the previous messages to the SVM decision don't express symptoms of illness. To achieve this goal, we followed the next steps:

1. For positive users, we calculated how many messages the SVM needs to classify the user as positive (depression or anxiety). Each user has a different trigger value.
2. For false negatives, we used the mean of the true positive trigger values as the trigger value.
3. For each positive user in the original data set, let n be the number of messages that the SVM model needs to determine this user's mental disorder risk, MAX be the maximum number of messages the model supports as input, and m_i the ith message from the user.

   a) we created $n - 1$ negative samples as follows:

$$(m_1), (m_1m_2), (m_1m_2m_3), ..., (m_1...m_{n-1})$$

   b) and $MAX - n + 1$ positive samples:

$$(m_1...m_n), (m_1...m_nm_{n+1}), ..., (m_1...m_n...m_{MAX})$$

4. Note that the value of MAX depends on which model was used and the number of tokens in the messages. That is, we discard messages from an accumulated history of more than 512 tokens for RoBERTa and 4096 for LongFormer. So, if $n > MAX$ only negative samples are generated.
5. For negative users, we created new samples accumulating the history as before, stopping when the MAX was reached.

The result of this technique is a new dataset with a higher number of positive samples for the training.

	Train	Development
None	4856	45
Depression	2832	35
Anxiety	1387	17
Total	9075	97

Table 5
Distribution of the new dataset: train and development partition.

3.2. Task 1: Disorder Detection

3.2.1. Classical Machine Learning Classifier Approach

To evaluate the context's importance, we wanted to use a classical machine learning classifier that can handle all the context. One of the most important issues of models based on Transformers is their poor ability to deal with large texts, because of their limitation in the input size. This

issue can affect the performance since the input cannot hold all the sample length, and valuable information may be lost in this process.

Firstly, we did an experiment where we compared different types of classical machine learning classifiers. Scikit-learn library [11] provided us with the tools to develop this experiment. The configuration was to use all the default classifiers to select the better one. The results can be seen in Table 6. The table shows that the best classifier was the Linear SVM.

	precision	recall	f1-score
Linear SVM	**0.74**	**0.74**	**0.73**
Gradient Boosting	0.50	0.48	0.49
K-Neighboors	0.44	0.50	0.47
Random Forest	0.61	0.55	0.59

Table 6
The results from different classifiers in the development partition. The scores are the Macro-precision, recall, f1-score.

Once the classifier was chosen, we wanted to test different approaches:

- **Preprocess of Data**:

  1. First approach: Transform the text into tokens using TweetTokenizer and then eliminate stop words.
  2. Second Approach: Same as the first approach with the addition of methods to clean the text, eliminate non-alphanumerical characters and others, and lemmatize tokens.

- **Sentimental Analysis**: We used the model **"lxyuan/distilbert-base-multilingual-cased-sentiments-student"** [12] to proceed with a sentimental analysis of every message per user. We obtained 3 results, positive messages, negative messages, and neutral messages, all normalized in the end. We add these results as a new feature for the TF-IDF.
- **TF-IDF**: The class `TfidfVectorizer` in *Scikit-learn* was used to vectorize the data. We tested different configurations for the analyzer and ngram_range number, and used the default values for the other features.

To find the best models for every approach, we did an exhaustive grid search over some specific parameters, such as regularization parameter C, different tols (Tolerance for stopping criteria), and different loss.

We obtained 6 different approaches. Table 7 shows the different configurations used in the experimentation, the column TF-IDF refers to the type of analyzers (word or char) used and the number of n-grams. The last column refers to the best model found in the search grid.

The result shows in Table 8 the best configuration is the **SVM-4**, using the most completed preprocess for the data, sentimental analysis, "char_wb" as the analyzer and (4-5) as ngram_range. This model was used for *Run0* in Task 1.

	Preprocess data approach	Sentiment analysis	TF-IDF	Best Model
SVM-1	1	No	"char_wb" , 4-5 n-gram	'C': 1, 'loss': 'squared_hinge', 'tol': 0.1
SVM-2	2	No	"char_wb" , 4-5 n-gram	'C': 1, 'loss': 'squared_hinge', 'tol': 0.1
SVM-3	1	Yes	"char_wb" , 4-5 n-gram	'C': 1, 'loss': 'hinge', 'tol': 0.1
SVM-4	2	Yes	"char_wb" , 4-5 n-gram	'C': 1, 'loss': 'hinge', 'tol': 0.1
SVM-5	1	No	"word" , 1-2 n-gram	'C': 10, 'loss': 'hinge', 'tol': 0.1
SVM-6	2	No	"word" , 1-2 n-gram	'C': 10, 'loss': 'hinge', 'tol': 0.1
SVM-7	1	Yes	"word" , 1-2 n-gram	'C': 10, 'loss': 'hinge', 'tol': 0.1
SVM-8	2	Yes	"word" , 1-2 n-gram	'C': 10, 'loss': 'hinge', 'tol': 0.1

Table 7
Summary of the different configurations of the SVM classifiers.

	precision	recall	f1-score
SVM-1	0.74	0.74	0.73
SVM-2	0.76	0.75	0.75
SVM-3	0.75	0.75	0.74
SVM-4	**0.79**	**0.76**	**0.76**
SVM-5	0.70	0.68	0.69
SVM-6	0.72	0.70	0.71
SVM-7	0.71	0.69	0.70
SVM-8	0.73	0.72	0.72

Table 8
Results of the different configurations of the SVM classifiers on development partition. In bold, the best result for each metric.

3.2.2. BERT-like Model Approach

It is well known that the state-of-the-art models in NLP are based on Transformers. Models like BERT or RoBERTa usually provide good versatility for classification tasks. However, these types of models usually can not handle more than 512 tokens, which could be a problem for tasks with long contexts such as the current ones. Therefore, we used one of these models as a baseline to compare other models with a better capacity to handle large contexts. Some research made by **Alireza Porkeyvan** [13] shows that the state of the art in mental disorder detection is **MentalRoBERTa** [14]. MentalRoBERTa is a RoBERTa-like model specialized in mental health. This model is pre-trained using a special corpus of texts from mental health forums, clinical notes, and normal corpus. Consequently, MentalRoBERTa provides better adaptation for the mental health-related language, which brings a lot of possible applications related to this domain.

The model chosen was ***AIMH/mental-roberta-large*** [15], a RoBERTa model trained with posts on Reddit related to mental health. This model can be found in HuggingFace [16] public hub (https://huggingface.co/AIMH/mental-roberta-large). Furthermore, we wanted to compare a specific domain RoBERTa model, like MentalRoBERTa, with the non-domain RoBERTa model, the baseline of the competition (RoBERTa base).

Once we chose our pre-trained model, we performed an experiment that consisted of testing two fine-tuning processes: one with the Dataset 1 (RoBERTa-1) and the other with the Dataset 2 (RoBERTa-2); the second dataset is the one with data augmentation. Table 9 shows the configuration used in the fine-tuning process.

parameter	value
optimizer	AdamW
learning rate	7e-5
lr scheduler type	linear
weight decay	0.01
number of epochs	10
training batch size	16

Table 9
Parameters for the fine-tuning process.

Table 10 shows the results of each model on the development partition. The results show that the best model is RoBERTa-2, the one fine-tuned with data augmentation. In our participation, this model was used for *Run1* in Task 1.

	Data Augmentation	Precision	Recall	F1-score
RoBERTa-1	No	0.81	0.82	0.81
RoBERTa-2	Yes	**0.94**	**0.94**	**0.93**

Table 10
RoBERTa's result for Task 1 on development partition.

3.2.3. LongFormer Approach

As we said before, one of the most important disadvantages of BERT-like or RoBERTa-like models based on Transformers is the lack of capacity to handle large contexts. However, a variant of Transformers can handle large text called LongFormer [7].

LongFormer is the abbreviation for "Long-Document Transformer" and can process long contexts more efficiently than Transformer models, such as BERT or RoBERTa. LongFormer architecture shows the following characteristics:

- **New attention mechanism**: An efficient attention mechanism that uses a sliding window, where each token only attends to a fixed number of neighborhood tokens, reducing the complexity.
- **Global attention selection**: The architecture can select which tokens are globally attended and which are just attended locally.

The pre-trained model chosen was ***AIMH/mental-longformer-base-4096*** [17] a pre-trained LongFormer for the mental health domain. This model can be found in https://huggingface.co/AIMH/mental-longformer-base-4096.

As in with the RoBERTa model, we fine-tuned the LongFormer with the two datasets: Dataset 1 without data augmentation (**LongFormer-T1-1**), and Dataset 2 with data augmentation

(**LongFormer-T1-2**). We used the same fine-tuning parameters as in RoBERTa's experimentation; the configuration is in Table 9.

Table 11 shows the results of the experimentation, where **LongFormer-T1-2** (fine-tuned with data augmentation) achieves better performance than **LongFormer-T1-1** (fine-tuned without data augmentation). This model was *Run2* in our participation.

	Data Augmentation	Precision	Recall	F1-score
LongFormer-T1-1	No	0.83	0.84	0.83
LongFormer-T1-2	Yes	**0.95**	**0.95**	**0.94**

Table 11
LongFormer's results for Task 1 on development partition.

3.3. Task 2

The experimentation for Task 1 shows that the best system is the LongFormer-T1-2, so to take part in Task 2 we only chose this approach. We used the LongFormer pre-trained model as the base model, increased the number of samples of the competition dataset with data augmentation, changed the labels for the new ones, and fine-tuned the model. **LongFormer-T2** model was used for *Run0* in the second task.

The Table 12 shows the results of the fine-tuning process.

	Model	Precision	Recall	F1-score
LongFormer-T2	LongFormer	0.99	0.98	0.97

Table 12
LongFormer's results for Task 2 on development partition.

4. Runs

Table 13 summarizes the selected model for each run, also the development performance is shown.

	Task	Model	Precision	Recall	F1-score
Run0	1	SVM-4	0.79	0.76	0.76
Run1	1	RoBERTa-2	0.94	0.94	0.93
Run2	1	LongFormer-T1-2	**0.95**	**0.95**	**0.94**
Run0	2	LongFormer-T2	0.99	0.98	0.97

Table 13
Summary of the approaches chosen for each run. Also, the performance achieved by each system in the development partition.

The reason for choosing these models was to assess the importance of context in predicting mental illness. Each model has a different input length capability, which can handle larger or smaller context sizes.

On the one hand, BERT-like models performed better than SVMs in the first task, even though BERT-like models can handle less context than SVMs. On the other hand, LongFormer performed slightly better than BERT-like models in the first task since LongFormer can handle larger contexts.

4.1. Run Configuration

Besides, to select the model for each run, the classification systems contained additional parameters that needed to be set:

Task1:

- For every round in the competition, we used as the input classifier a new sample created combining the new message of the user with the previous ones.
- Each system has an initial context, in other words, we made our systems wait until the initial context was sufficiently large. This context was different in each system:
    - **SVM**: An initial context of 50 tokens after the pre-process.
    - **RoBERTa and LongFormer**: An initial context of 100 tokens.
- The RoBERTa and LongFormer system has a limit of tokens, when the system was full we just returned the last prediction made.

Task2:

For the second task, we combined the best model from Task 1 (LongFormer-T1-2) and the one fine-tuned specifically for Task 2 (LongFormer-T2). The first model was used to discriminate between negative cases and positive cases. If the sample was detected as positive, then the LongFormer-T2 was used to predict the context.

5. Results

5.1. Task 1

Table 14 shows the results achieved by our teams in Task 1. The structure of the Table 14 is the following: rows refer to each run and a special row which refers to the highest values of the competition. The systems in the competition were ranked using the Macro-F1 score (last column).

	Model	Accuracy	Macro-P	Macro-R	Macro-F1
Run0	SVM	0.848	0.840	0.838	0.833
Run1	RoBERTa	0.850	0.853	0.845	0.840
Run2	LongFormer	**0.890 (1)**	**0.875 (1)**	**0.880 (1)**	**0.874 (1)**
Highest	-	0.890	0.875	0.880	0.874

Table 14
Results for the 3 runs on Task 1. *Highest* refers to the highest values achieved in the competition. The values inside the parenthesis indicate our position in the ranking.

Table 14 shows how the best system is the **Run 2**, which refers to the **LongFormer-T1-2**: pre-trained LongFormer fine-tuned with the data augmentation. This run achieved the first position in the competition. The only two runs that beat the Baseline were our Run1 and Run2, indicating the importance of appropriate data selection.

Although the best runs used a model base in Transformers, the run with SVM achieves a similar result, only 1% less than *Run1*. This indicates that classical approaches like SVMs continue to be useful in detecting mental illnesses because of their ability to handle large contexts. Therefore, SVMs still well-fitted in situations with low computational resources.

5.2. Task 2

Table 15 shows the results for the Task 2. Our run was fifth in the official ranking, based on Macro-F1 score.

	Accuracy	Macro-P	Macro-R	Macro-F1
Run0	0.065 (4)	0.262 (3)	0.177 (5)	0.208 (5)
Highest	0.077	0.358	0.508	0.268

Table 15
Results for the Task 2. *Highest* refers to the highest values achieved in the competition. The values inside the parenthesis indicate our position in the ranking.

As can be seen from Table 15, the results obtained by our system in the competition are not as good as those obtained in the development partition, which might indicate that the model was overfitted during the fine-tuning process. Further analysis is needed to find the source of the low generalization capabilities of the developed model.

5.3. Carbon emission

One of the main goals of the competition is to identify systems that complete tasks with minimal resource consumption[1]. This will help them pinpoint technologies that can operate on mobile devices or personal computers and those with the lowest carbon emissions. Therefore, we include the following information:

- Total time to process (in milliseconds)
- Kg in CO_2 emissions.

Using the provided script, which utilizes the CodeCarbon API [18] to calculate emissions, we present our team's computer configuration in Table 16. This table details the types and quantities of CPUs and GPUs employed, as well as the total RAM used. We present the results for the **LongFormer-T1-2 Run 2**.

Measurements	Values
CPU_Count	24
GPU_Count	1
CPU_Model	12th Gen Intel(R) Core(TM) i9-12900K
GPU_Model	NVIDIA GeForce RTX 4090
RAM_Total_Size	128 GB
Country_ISO_Code	ESP

Table 16
Computer configuration

Figure 1 illustrates the variation in emissions and duration during the experimentation. A direct correlation exists between each measurement, indicating that rounds with longer durations emitted more CO_2. Since every round utilized the same models and configurations, the primary factor influencing emissions was the length of the round and the accumulated context of the user.

(a) Emissions of CO_2 (Kg) of each round   (b) Duration (milliseconds) of each round
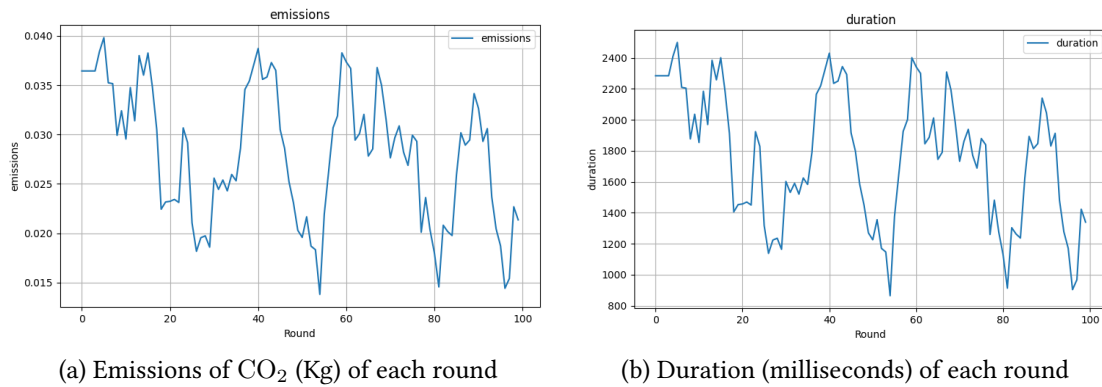
Figure 1: Emissions and Durations Graphs

Figure 2 displays the cumulative energy consumption of each component. The GPU is the highest energy-consuming component, accounting for approximately 83% of the total energy usage. The CPU follows, consuming 16.5%, while RAM accounts for only 0.5% of the total energy consumption.
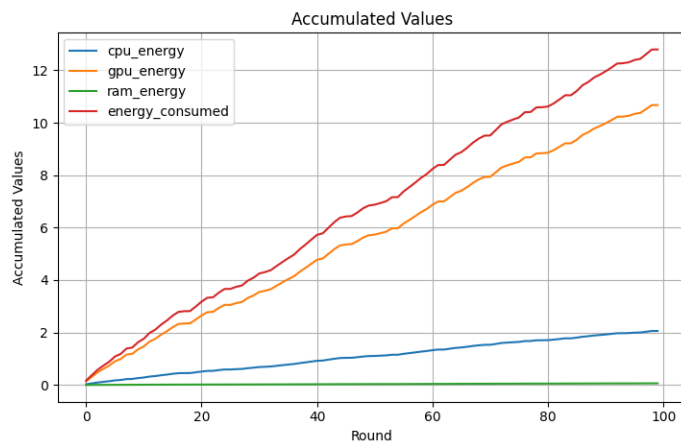
Figure 2: Accumulated values of energy (kWh) during the rounds

6. Conclusion

In this paper, we have presented the participation of the ELiRF-VRAIN team in the shared tasks of MentalRiskES at IberLef 2024. In addition to testing classic classification models and state-of-the-art transformer models, our team's most innovative contribution was using LongFormer models to expand the context for making the decision and increase the training corpus through data augmentation.

The results obtained support the correctness of our proposal, being the only team to exceed the baseline presented by the organization of the shared task.

For future work, two lines of improvement are identified. On the one hand, try to improve early detection so that the system does not need as much initial context to make the right decision; on the other hand, use Explainable Artificial Intelligence (XAI) techniques to better understand the system's behavior.

Acknowledgments

References

[1] L. Chiruzzo, S. M. Jiménez-Zafra, F. Rangel, Overview of IberLEF 2024: Natural Language Processing Challenges for Spanish and other Iberian Languages, in: Proceedings of the

Iberian Languages Evaluation Forum (IberLEF 2024), co-located with the 40th Conference of the Spanish Society for Natural Language Processing (SEPLN 2024), CEUR-WS.org, 2024.

[2] World Health Organization, Mental disorders, 2022. URL: https://www.who.int/news-room/fact-sheets/detail/mental-disorders, accessed: 2024-05-15.

[3] World Health Organization, Mental disorders fact sheet, 2022. URL: https://www.who.int/news-room/fact-sheets/detail/mental-disorders, accessed: 2024-05-21.

[4] A. M. Mármol-Romero, A. Moreno-Muñoz, F. M. P. del Arco, M. D. Molina-González, M.-T. Martín-Valdivia, L. A. Ureña-López, A. Montejo-Ráez, Overview of mentalriskes at iberlef 2024: Early detection of mental disorders risk in spanish, Procesamiento del Lenguaje Natural 73 (2024).

[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] 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.

[7] 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.

[8] A. M. Mármol Romero, A. Moreno Muñoz, F. M. Plaza-del Arco, M. D. Molina González, M. T. Martín Valdivia, L. A. Ureña-López, A. Montejo Ráez, MentalRiskES: A new corpus for early detection of mental disorders in Spanish, in: N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, N. Xue (Eds.), Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), ELRA and ICCL, Torino, Italia, 2024, pp. 11204–11214. URL: https://aclanthology.org/2024.lrec-main.978.

[9] N. Reimers, Easynmt: A simple interface to state-of-the-art machine translation models, 2020. URL: https://github.com/UKPLab/EasyNMT, accessed: 2024-05-15.

[10] J. Tiedemann, S. Thottingal, OPUS-MT — Building open translation services for the World, in: Proceedings of the 22nd Annual Conferenec of the European Association for Machine Translation (EAMT), Lisbon, Portugal, 2020.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] AIMH, Mentalroberta: A robustly optimized bert pretraining approach for mental health, 2024. URL: https://huggingface.co/AIMH/mental-roberta-large, accessed: 2024-05-15.

[16] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, A. M. Rush, Transformers: State-of-the-art natural language processing, 2020. URL: https://arxiv.org/abs/1910.03771. arXiv:1910.03771.

[17] AIMH, Mentallongformer: A long-document transformer model for mental health, 2024. URL: https://huggingface.co/AIMH/mental-longformer-base-4096, accessed: 2024-05-15.

[18] CodeCarbon, Codecarbon: Track and reduce your carbon emissions from machine learning workloads, https://mlco2.github.io/codecarbon/index.html, 2024. Accessed: 2024-05-15.