# SINAI at eRisk@ CLEF 2024: Approaching the Search for Symptoms of Depression and Early Detection of Anorexia Signs using Natural Language Processing.

Notebook for the eRisk Lab at CLEF 2024

Alba María Mármol-Romero[1], Adrián Moreno-Muñoz[1], Pablo Álvarez-Ojeda[1], Karla María Valencia-Segura[2], Eugenio Martínez-Cámara[1], Manuel García-Vega[1] and Arturo Montejo-Ráez[1]

[1]Computer Science Department, SINAI, CEATIC, Universidad de Jaén, 23071, Spain
[2]Computer Science Department, Instituto Nacional de Astrofísica, Óptica y Electrónica, Mexico

**Abstract**

This paper describes the participation of the SINAI team in the eRisk@CLEF lab. Specifically, two of the proposed tasks have been addressed: i) Task 1 on the search for symptoms of depression, and ii) Task 2 on the early detection of signs of anorexia. The approach presented in Task 1 is based on the use of a two-step detection approach using a transformer-based model, while the approach for Task 2 is based on calculating perplexity using two transformer-based models trained with causal language modelling: one on positive user data and the other on negative user data. In Task 1, our team has been ranked in the 7$^{th}$ position out of a total of 9 participating teams, with a 0.562 score of precision at 10 in the majority ranking. In Task 2, our team placed the 6$^{th}$ out of a total of 10 participating teams in the F1 score and we reached the best overall values after only two writings.

**Keywords**

Early risk prediction, Anorexia detection, Symptoms of depression detection, Natural Language Processing, Transformers, Perplexity, Large Language Model

## 1. Introduction

The large amount of content shared daily on social media has made these platforms a significant source of data for the early detection of mental disorders and risky behaviours. The eRisk@CLEF 2024 lab [1, 2] focuses on furthering the development of computational systems able to early detect mental disorders such as depression, self-harm or eating disorders. In particular, the following three tasks have been proposed:

- **Task 1 - Search for symptoms of depression**. It consists of ranking sentences from a collection of user writings according to their relevance to a depression symptom. Then, the participants will have to provide rankings for the 21 symptoms of depression from the BDI Questionnaire. It is a continuation of Task 1 proposed for eRisk 2023 [3].
- **Task 2 - Early Detection of Signs of Anorexia**. It consists of sequentially processing pieces of evidence and detecting early traces of anorexia as soon as possible. It is a continuation of Task 2 proposed for eRisk 2018 [4] and Task 1 proposed for eRisk 2019 [5].
- **Task 3 - Measuring the severity of the signs of Eating Disorders**. Its aim is to estimate a user's level of disordered eating from his or her history of posts. For this purpose, each user has to fill in a standard eating disorder questionnaire (EDE-Q). This is a continuation of 2022 [6] and 2023's Task 3 [3].

*CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France*
amarmol@ujaen.es (A. M. Mármol-Romero); ammunoz@ujaen.es (A. Moreno-Muñoz); paojeda@ujaen.es (P. Álvarez-Ojeda); valencia.karla@inaoep.mx (K. M. Valencia-Segura); emcamara@ujaen.es (E. Martínez-Cámara); mgarcia@ujaen.es (M. García-Vega); amontejo@ujaen.es (A. Montejo-Ráez)
0000-0001-7952-4541 (A. M. Mármol-Romero); 0009-0000-8809-8804 (A. Moreno-Muñoz); 0009-0000-9026-6205 (P. Álvarez-Ojeda); 0000-0003-2796-6561 (K. M. Valencia-Segura); 0000-0002-5279-8355 (E. Martínez-Cámara); 0000-0003-2850-4940 (M. García-Vega); 0000-0002-8643-2714 (A. Montejo-Ráez)

In this context, our primary objectives extend beyond merely creating high-performing systems. We aim to comprehend the most effective methods and approaches applicable to similar scenarios. Rather than assembling numerous features and systems into an ensemble of predictors solely for achieving top rankings, our focus lies in identifying the best approaches aligned with our objectives: (1) the importance of ranking messages according to the different symptoms of the BDI Questionnaire described in Task 1, and (2) the design of online and monitoring tools, as is requested in Task 2.

This work presents the participation of our research group, the SINAI[1] team in *Task 1: Search for symptoms of depression* and *Task 2: Early Detection of Signs of Anorexia.* The rest of the paper is organized as follows: sections 2 and 3 describe in detail our participation in task 1 and task 2, respectively. Each of them is divided into subsections in which, first, we introduce what the task consists of, the data provided and the evaluation measures used. Secondly, the system developed and the methodology used are presented. Thirdly, the experimental setup is detailed. Subsequently, the results obtained and a discussion on them are presented. Finally, Section 4 shows the conclusions obtained after the participation in the eRisk lab and the perspectives for future work.

## 2. Task 1: Search for symptoms of depression

### 2.1. Task description

This task focuses on searching signals of depression in users from social media. In this case, the systems had to read the sentences, which are embedded within the preceding and following context, and determine their relevance to each of the 21 symptoms in the BDI Questionnaire. Additionally, each symptom has its own ranking, which will correspond to the 1000 most relevant for each distinct symptom.

The provided training data is annotated at sentence level regarding its relevance to symptom query. Each sentence is annotated by three annotators, and the dataset provides a majority vote on ground truth and the unanimous or aggregated label. The test data is composed of 551,311 users with a total of 15,542,200 sentences which have an average number of words of 17.98.

### 2.2. System and methods

We have explored two different systems for this task. One submission is based on fine-tuning with data augmentation and uses a two-step detection approach (see Section 2.2.1) and the other is based on prompting to GPT-3 (see Section 2.2.2).

#### 2.2.1. Fine-tuned model with data augmentation

The SINAI_DR_majority_daug run trained a DistilRoBERTa base model[2] using the training data provided in its majority vote form and with data augmentation using the BDI-Sen dataset [7].

**Data** We used BDI-Sen which is a symptom-based dataset with relevant sentences that trace the presence of clinical symptoms, for the labelling they used three expert annotators consisting of a psychologist, a speech therapist, and a PhD student. This dataset contains a total of 18,510 sentences. The data in the BDI-Sen dataset is labelled from 0 to 4, where 4 represents the control class, while values from 0 to 3 indicate severity, with 0 being the mildest and 3 the most severe. We have modified these numbers as follows: we consider values 0, ñ, and 4 as unrelated (0), and values 1, 2, and 3 as related (1). Table 1 shows all the data from relevant sentences used in training.

After training the model, we computed the probability for each user to suffer each BDI Questionnaire symptom. We then ranked the scores from highest to lowest and selected the top 1,000 users with the highest probability for each symptom. Following this, we reapplied the previously trained model,

[1] https://sinai.ujaen.es/

[2] https://huggingface.co/distilbert/distilroberta-base

**Table 1**
This table presents relevant data for symptoms based on the BDI Questionnaire, the BDI-Sen dataset and the total data used for the train. The numbers represent the relevant sentences of each symptom in the respective datasets

| BDI Item | data majority voting | BDI-Sen data | Total |
|---|---|---|---|
| Sadness | 319 | 1485 | 1804 |
| Pessimism | 334 | 1317 | 1651 |
| Past Failure | 304 | 1162 | 1466 |
| Loss of Pleasure | 207 | 1214 | 1421 |
| Guilty Feelings | 143 | 920 | 1063 |
| Punishment Feelings | 50 | 1145 | 1195 |
| Self-Dislike | 288 | 1203 | 1491 |
| Self-Criticalness | 174 | 1095 | 1269 |
| Suicidal Thoughts or Wishes | 349 | 1007 | 1356 |
| Crying | 320 | 1068 | 1382 |
| Agitation | 155 | 1132 | 1287 |
| Loss of Interest | 168 | 1143 | 1311 |
| Indecisiveness | 141 | 1141 | 1282 |
| Worthlessness | 144 | 1165 | 1309 |
| Loss of Energy | 204 | 1117 | 1321 |
| Changes in Sleeping Pattern | 351 | 948 | 1299 |
| Irritability | 155 | 1153 | 1308 |
| Changes in Appetite | 224 | 998 | 1222 |
| Concentration Difficulty | 141 | 1057 | 1198 |
| Tiredness or Fatigue | 222 | 1091 | 1313 |
| Loss of Interest in Sex | 159 | 974 | 1133 |

focusing only on users with high scores. Then we selected the top 1,000 sentences based on their ratings. By filtering at the user level first and then at the sentence level, this two-step procedure allowed us to refine the selection further and ensure the most relevant content was found for each symptom. We trained the model to perform a search for the 21 symptoms using multi-label regression on a compute node equipped with a single NVIDIA-A100 GPU with 40 GB of memory. To achieve this, we searched the best hyper-parameters with Optuna [8]. We trained the model over 10 epochs, set the learning rate to 1e-05 and a weight decay of 0.007, we also organized the training process into batches of 16 and used the AdamW optimizer.

### 2.2.2. Prompting GPT-3.5

The second proposed approach is based on Large Language Models (LLM) and based on the search for the 8 symptoms evaluated in the PHQ-8 (Patient Health Questionnaire-8) [9]: (a) Feeling down, sad, or hopeless. (b) Not enjoying things you used to like. (c) Trouble sleeping or sleeping too much. (d) Feeling tired all the time. (e) Changes in appetite, either eating too much or too little. (f) Feeling worthless or like a failure. (g) Difficulty concentrating on things. (h) Moving slowly or feeling restless and fidgety. This approach provides a solid foundation, as these symptoms can be indicative of underlying emotional conditions. By targeting these symptoms, a starting point is established to explore the 21 symptoms of the BDI, as many of them may be interconnected. Since the PHQ-8 only assesses 8 symptoms while the BDI assesses 21, with the assistance of a physician, we conducted a symptom mapping, which can be observed in Table 2.

The approach consisted of two steps: The first focused on generating 15 synthetic phrases with the generative model GPT-3.5 [10] for each of the 8 symptoms evaluated in the PHQ-8. The prompts used to generate these phrases were designed with the assistance of medical staff, aiming for as much detail as possible to provide GPT-3.5 with accurate and comprehensive information, thus simulating a real interview scenario. Additionally, the prompts were tailored to obtain phrases related specifically to

**Table 2**
Mapping of symptoms between the 8 PHQ-8 symptoms and the 21 BDI symptoms.

| PHQ-8 item | BDI item |
|---|---|
| Little interest or pleasure in doing things | Loss of Pleasure, Loss of Interest, Loss of Interest in Sex |
| Feeling down, depressed, or hopeless | Sadness, Pessimism, Suicidal Thoughts or Wishes, Crying, Irritability |
| Trouble falling or staying asleep, or sleeping too much | Changes in Sleeping Pattern |
| Feeling tired or having little energy | Tiredness or Fatigue |
| Poor appetite or overeating | Changes in Appetite |
| Feeling bad about yourself — or that you are a failure or have let yourself or your family down | Past Failure, Guilty Feelings, Punishment Feelings, Self-Dislike, Self-Criticalness, Worthlessness |
| Trouble concentrating on things, such as reading the newspaper or watching television | Indecisiveness, Concentration Difficulty |
| Moving or speaking so slowly that other people could have noticed? Or the opposite — being so fidgety or restless that you have been moving around a lot more than usual | Agitation, Loss of Energy |

depressive signs. The second step aimed to find relevant phrases in the user's history. We utilized the synthetic phrases generated by GPT-3.5 as anchors to search for similar phrases in the user's history using semantic similarity. This strategy combines the power of artificial intelligence with the richness of user contextual information.

Once we obtained phrases per symptom, we computed a phrase vector using a pre-trained transformer from SentenceTransformer [11]. Once we have obtained representations for synthetic phrases, we proceed with a semantic search within the user histories. The ranking of phrases was determined as follows: for each user history phrase, we computed the cosine similarity with each of the symptom-related phrases generated by GPT-3.5, then summed this similarity across the 15 phrases for each symptom, and obtained a score. This score was assigned to each phrase in the history, thus determining the relevance of each phrase for each symptom.

## 2.3. Results and discussion

The organizers conducted a labelling process involving three annotators who manually selected the relevant writings. This process led to the creation of two evaluation schemes: majority voting (requiring agreement from 2 out of 3 annotators) and unanimity (requiring agreement from all 3 annotators).

**Table 3**
Results of SINAI team for Task 1 in ranking-based evaluation (majority voting). Results are reported according to the metrics Average Precision (AP), R-Precision (R-PREC), Precision at 10 (P@10) and NDCG

| Run | $AP$ | $R-PREC$ | $P@10$ | $NDCG$ |
|---|---|---|---|---|
| SINAI DR majority daug | 0.064 | 0.107 | 0.562 | 0.174 |
| GPT3-Insight-8 | 0.008 | 0.024 | 0.200 | 0.044 |

**Table 4**
Results of SINAI team for Task 1 in ranking-based evaluation (unanimity).

| Run | $AP$ | $R-PREC$ | $P@10$ | $NDCG$ |
|---|---|---|---|---|
| SINAI DR majority daug | 0.046 | 0.098 | 0.362 | 0.150 |
| GPT3-Insight-8 | 0.001 | 0.009 | 0.052 | 0.014 |

It is clear from Table 3 and Table 4 that the system obtains better results in the ranking with a majority vote than in the ranking with a unanimous vote. While AP, R-Precision, and NDCG measurements show only slight variations at P@10, a significantly larger shift is apparent.

We argue that this phenomenon stems from model overfitting, which occurs when there is a disparity in the representation of related and unrelated instances during the training phase.

# 3. Task 2: Early detection of signs of anorexia

## 3.1. Task description

This task focuses on early risk detection of signs of anorexia by processing posts from social media in strict order of publication. The participant systems had to read the posts (from several subjects) in the order in which they were created, process them and generate a response in order to get the next posts.

The task is faced from two different perspectives: as a binary decision problem, and as a ranking (regression) decision problem. As a binary decision problem, posts have to be labelled as positive (label 1, i.e. risk detected) or negative (label 0, no risk detected). The earlier the system detects an addiction, the better, as it is reflected with the ERDE and $F_{latency}$ metrics proposed by the organizers and used to evaluate the systems, along with the well-known precision, recall and F1 scores. As a ranking decision problem, instead of assigning 0 or 1 labels, a score of the estimation of the risk of suffering such a disorder is computed. Different metrics as the ones used for information retrieval are considered to evaluate this second view of the task (P@10 or NDCG, among others).

The test data is composed of 366,886 posts by 784 different subjects in total: 92 subjects labelled as positive in risk detection of signs of anorexia, and 692 subjects not labelled as at risk.

## 3.2. Systems and methods

We submitted a system based on the calculation of perplexity. For that, we trained two transformer-based models with causal language modelling, one with data from positive users and the other with data from negative users.

Nonetheless, though it was not submitted, another approach was performed, consisting of creating and employing emotion vectors, relying on several English lexicons, and training various classical Machine Learning (ML) algorithms with 80% of the provided observations of the dataset.

For the training and experiments conducted, only the data provided by the organisers consisting of data from the 2018 and 2019 editions was used. In total, the training set has 1,245 subjects, with 134 subjects labelled as positive for suffering the risk of anorexia and 1,111 subjects labelled as negative. Furthermore, the total amount of training data they give amounts to 799,186 writings. For both approaches, we split the data provided by organizers into training (996 subjects of whom 103 are labelled as positive and a total of 639,824 texts) and valid (249 subjects of whom 31 are labelled as positive and a total of 159,362 texts) datasets. From now on, whenever we refer to training data, we will refer to this subset.

### 3.2.1. Emotion vectors approach

**Data pre-processing** Each entry is pre-processed by removing URLs, line breaks, text only containing blank spaces or empty strings and texts whose number of tokens was lower or equal to 10 (this high

threshold was chosen due to the large size of the texts of the dataset). Though for the following system used in this task titles will not be considered, in this case, if a record does not have text, the title will be used as such, avoiding a large number of empty entries to be removed. Table 5 shows the size of the datasets used

**Table 5**
Data available in the datasets developed for training emotion vectors approach for Task 2.

| Dataset | Texts | Words (mean) | Words (std) |
|---|---|---|---|
| Positive | 37,183 | 64.35 | 116.85 |
| Negative | 500,732 | 37.88 | 77.58 |

**Methodology** First of all, it is necessary to introduce the employed algorithms for this approach. We have used the default hyperparameter values of the methods used from the scikit-learn library [12]. In particular, the ML methods used were:

- Logistic regression with a maximum of 1000 iterations.
- RandomForest using 20 decision trees and seed with value 45.
- MLP with a maximum of 1000 iterations.
- SVM (default).
- Decision tree (default).
- KNN (default).
- SGD with a limit of 1000 iterations, stopping criteria with value $1 \times 10^{-4}$ and 45 as a seed value.

In order to conduct the experiment, it is necessary to obtain valid English emotion lexicons. The employed lexicons in our experiments are NRC [13], ESN [14], DPM (DepecheMood) [15], DPM with normalized term frequency, and the intersection of words appearing in DPM and NRC using DPM emotion values. Each word has a value for each one of the considered emotions, which are Ekman [16] basic ones (anger, disgust, joy, sadness, surprise and fear) plus "don't care" for DPM and finally, anticipation, negative and positive polarity and trust for NRC.

Regarding the training of the algorithms, the messages are grouped by user. Therefore, the generated array of each user will contain information about all the messages sent by the person. In the test phase, messages are not grouped all together by user and the arrays are re-generated each round, one message per user each lap.

To build the emotion vectors, the text for each observation will be tokenized by word and every time a word in the text also belongs to the lexicon currently being worked with, it is registered alongside its emotion values (if the word appears more than once, the values for that word are registered as many times as occurrences in the text there are). Therefore, the final result for each message's feature vector will be a list of 4 sublists (maximum, minimum, average and standard deviation of the values of each gathered emotion) each one of those containing many positions equal to the number of "measured" emotions a lexicon has. Once the list of lists is created, the resulting data structure for each round is generated getting the maximum, minimum, average and standard deviation between the newly created array and the one from the previous round, being this the feature we will use in our classification experiments.

**Results and discussion** The results were obtained using the created validation set and measured utilizing common ML evaluation metrics. Before presenting the table, the runs that populate are presented:

- Run 0: normalized DPM with KNN as best performing algorithm.
- Run 1: ESN with KNN as best performing algorithm.

- Run 2: intersection of words in NRC and DPM with normalized DPM emotion values being KNN the best performing algorithm.
- Run 3: intersection of words in NRC and DPM with DPM emotion values being Random forest the best performing algorithm.
- Run 4: intersection of words in NRC and DPM with DPM emotion values being stochastic gradient descent the best performing algorithm.

The runs were launched in an ordinary computer (i7-10700 CPU and 32 GB of memory). The 5 best results ordered by Macro-F1 are shown in Table 6.

**Table 6**
Results of SINAI team for Task 2: Emotion vectors approach.

| Run | $P$ | $R$ | $F1$ | $ERDE_5$ | $ERDE_{30}$ | $latency_{tp}$ | $speed$ | $latency_w F1$ |
|---|---|---|---|---|---|---|---|---|
| 0 | 0.5447 | 0.5668 | 0.5471 | 0.1487 | 0.1425 | 31.0 | 0.4513 | 0.0440 |
| 1 | 0.5457 | 0.5876 | 0.5349 | 0.1598 | 0.1598 | 178.0 | 0.0014 | 0.0001 |
| 2 | 0.5152 | 0.5277 | 0.5015 | 0.1547 | 0.1326 | 30.0 | 0.4658 | 0.0847 |
| 3 | 0.5182 | 0.5346 | 0.5012 | 0.1563 | 0.1563 | 199.0 | 0.0006 | 0.0001 |
| 4 | 0.5000 | 0.5000 | 0.4990 | 0.1371 | 0.1371 | - | 0.0 | 0.0 |

Regarding the obtained score, DPM (DepecheMood) seems the most useful lexicon out of all those tested. Indeed, combined with NRC terms, it populates 4 positions within the top 5 best runs. Also, the K-nearest-neighbors algorithm provides the 3 best results regarding the Macro-F1 metric. What is more, it is worth mentioning the proficient results in ERDE5 and ERDE30 being this approach capable of detecting true positives acceptably fast. A future improvement involves a hyperparameter search to find the best possible tuning for the employed algorithms.

### 3.2.2. Perplexity approach

**Data pre-processing** We apply a pre-processing of the training dataset which consists of only keeping the 'text' field from each writing user. Thus, we ignore titles of publications that do not have content in the text field because we suppose that if a writing does not have text then it does not contain valuable information. Again URLs, non-alphanumeric characters and blank spaces are removed. Moreover, we removed empty writings (when a text contains only punctuation, spaces or '[removed]' target) and writings whose text contains less than four tokens. Once texts have been cleaned, we split the training dataset based on the label values. So we had two different datasets: one with texts coming from subjects labelled as positive (label = 1) and another with texts coming from subjects labelled as negative (label = 0). Table 7 shows some statistics about the datasets developed to train the systems. Of the 639,824 texts we had before the processing was applied in the training dataset we developed, we now have 411,498 texts.

**Table 7**
Data available in the datasets developed for the training of perplexity approach for Task 2

| Dataset | Texts | Words (mean) | Words (std) |
|---|---|---|---|
| Positive | 28,521 | 59.26 | 88.94 |
| Negative | 382,977 | 39.24 | 77.97 |

**Methodology** Once we have the data prepared, we proceed to train the models. We selected the GPT-2[3] model because it is freely available for download from HuggingFace[4] and is straightforward

[3]https://huggingface.co/openai-community/gpt2
[4]https://huggingface.co/

to use and train. Additionally, GPT-2 performs well with the English language, which is the primary language of our dataset and can also handle other languages. We trained two models:

- positive_model: using texts from subjects labelled as positives
- negative_model: using texts from subjects labelled as negatives

Our main goal is to see which model finds a given text more familiar. We do this by calculating the perplexity for each text using both models. The model with the lower perplexity value indicates the text is more similar to its training data.

Next, we tested the models with our validation dataset. We ran experiments to find a good threshold for making predictions because the two models were trained on different amounts of data, making their perplexity values not directly comparable. We calculated the difference between the two perplexity values and checked if this difference was less than a certain threshold. We tested thresholds from 0 to 1,000 and then focused on 0 to 100 to find the best one. Then, if the perplexity from the positive_model is lower than that from the negative_model, the subject is classified as positive. However, a subject can also be classified as positive if the difference between the perplexities is less than a specified threshold, even if the positive_model perplexity is higher. This formula 1 determines whether a subject is labelled positive.

$$\text{ppl\_pos} < \text{ppl\_neg} \quad \text{or} \quad \text{dif} < \text{threshold} \tag{1}$$

Where:

$$\begin{aligned} \text{ppl\_pos} &: \text{Perplexity computed with the positive\_model} \\ \text{ppl\_neg} &: \text{Perplexity computed with the negative\_model} \\ \text{dif} &: \text{Difference between ppl\_pos and ppl\_neg} \\ \text{threshold} &: \text{Threshold for accepting the difference} \end{aligned}$$

Additionally, we also explored the role of context in our experiments: (1) We tested predictions using just one round of text from each subject and (2) we also tested by combining texts from multiple rounds for each subject. If the combined text is over 500 words, older texts are removed stack-like until the total word count is under 500.

We applied the same pre-processing to each text as we did to the training data. We labelled the subject as negative if a text was too short or invalid. The scores we report are based on the perplexity values from the positive_model.

**Experiments** For the testing phase, the experiment set up was:

- Run 0: $text$ refers to the text provided for the subject at each round.
  $ppl(model^+, text) < ppl(model^-, text) \longrightarrow positive$
- Run 1: $text$ refers to the text provided for the subject at each round.
  $ppl(model^+, text) < ppl(model^-, text)$ *OR* $ppl(model^+, text) - ppl(model^-, text) < 45 \longrightarrow positive$
- Run 2: $text$ refers to the text provided for the subject at each round.
  $ppl(model^+, text) < ppl(model^-, text)$ *OR* $ppl(model^+, text) - ppl(model^-, text) < 60 \longrightarrow positive$
- Run 3: $text$ refers to the concatenation of texts provided for the subject.
  $ppl(model^+, text) < ppl(model^-, text) \longrightarrow positive$
- Run 4: $text$ refers to the concatenation of texts provided for the subject.
  $ppl(model^+, text) < ppl(model^-, text)$ *OR* $ppl(model^+, text) - ppl(model^-, text) < 45 \longrightarrow positive$

Both the model trained with positive texts and the model trained with negative texts are trained with 3 epochs, 0.01 weight decay, 1e-05 learning rate and batch size equal to 8. A maximum length of 512 tokens per document was set and the optimizer used was AdamW. All experiments in the pre-evaluation and evaluation phases were run on a compute node equipped with a single Tesla-V100 GPU with 32 GB of memory.

### 3.3. Results and discussion

From the reported results provided by the organizers, we have extracted our scores, which are shown in Tables 8 and 9.

**Table 8**
Results of SINAI team for Task 2 (perplexity approach) in decision-based evaluation. In bold the highest possible score obtained is marked.

| Run | $P$ | $R$ | $F1$ | $ERDE_5$ | $ERDE_{50}$ | $latency_{tp}$ | $speed$ | $latency_w F1$ |
|---|---|---|---|---|---|---|---|---|
| 0, 1, 2 | 0.21 | 0.92 | 0.34 | 0.10 | 0.07 | 3.0 | 0.99 | 0.34 |
| 3, 4 | 0.12 | **1.00** | 0.21 | 0.13 | 0.10 | 2.0 | **1.00** | 0.21 |

**Table 9**
Results of SINAI team for Task 2 (perplexity approach) in ranking-based evaluation.

| | 1 writing | | | 100 writing | | | 500 writing | | | 1000 writing | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Run | $P@10$ | $NDCG@10$ | $NDCG@100$ | $P@10$ | $NDCG@10$ | $NDCG@100$ | $P@10$ | $NDCG@10$ | $NDCG@100$ | $P@10$ | $NDCG@10$ | $NDCG@100$ |
| 0, 1, 2 | 0.00 | 0.00 | 0.07 | 0.00 | 0.00 | 0.02 | 0.00 | 0.00 | 0.02 | 0.00 | 0.00 | 0.03 |
| 3, 4 | 0.00 | 0.00 | 0.07 | 0.10 | 0.07 | 0.06 | 0.00 | 0.00 | 0.07 | 0.00 | 0.00 | 0.07 |

The first thing that is observable when looking at Table 8 is that there is no difference between runs 0, 1 and 2 (without contexts) and runs 3 and 4 (with contexts). So, the threshold estimated in the training phase is not valid for the test phase. We may think the test data is too far from the data used in training or that we have over-adjusted the thresholds we have set. Therefore the only observable differences that can be discussed lead us to use context (concatenation of messages) or not to use context (taking into account only the current text).

The approaches based on contexts locate all positive users in the test set, the rest come close to achieving this goal. In addition, the prediction that only takes into account the current text is more accurate than with context but needs more messages in the detection of true-positive subjects.

Table 9 shows us that the scores sent are not valid values for this task.

Deep pre-processing of the training data should have been done to obtain better results, as some texts labelled as positive (because they came from a subject labelled as positive) were very similar to texts labelled as negative. For example, the positive_model was trained with the positively labelled sentence "They say dogs look like their owners....", which tokenised with the GPT-2 model has a very high cosine similarity to the negatively labelled sentence "Dogs are so much more cool than people." used to train the negative_model. We selected GPT-2 due to its established performance metrics and accessibility. However, we acknowledge that the use of a more recent and advanced model, such as GPT-3 or GPT-4, could potentially enhance performance and yield more impressive results. Future iterations of our research will consider integrating these more advanced models to improve accuracy and overall outcomes. Re-labelling of the messages, as for example was done in Fabregat et al. [17] in another edition of a similar task, as well as a search for hyperparameters and testing with other LLMs is also proposed as future work.

## 4. Conclusions and future work

This paper describes our participation as the SINAI team in Task 1 and Task 2 of the eRisk@CLEF 2024 edition. The former is the continuation of the first edition in 2023 and consists of ranking sentences from a collection of user writings according to their relevance to a depression symptom, while the latter is the continuation of the 2018 and 2019 editions and consists of sequentially processing pieces of evidence and detect early traces of anorexia as soon as possible.

For Task 1, we have developed two different approaches: one is based on a transformer-model training with a two-step ranking, and the other is based on prompting GPT-3. As can be seen in the results, the pre-trained model with data augmentation works better than an LLM. In the two-step approach, the results are not entirely satisfactory, possibly owing to the model overfitting that could have been conducted with the dataset. For Task 2, we developed two different approaches, the emotion vectors approach and perplexity approach. We submitted the last one and achieved the best scores for recall and speed metrics. In future work, we plan to perform error analysis for both tasks to identify the main weaknesses of our systems and apply another preprocessing of the data.

## Acknowledgments

This work has been partially supported by projects CONSENSO (PID2021-122263OB-C21), MODERATES (TED2021-130145B-I00), SocialTOX (PDC2022-133146-C21) funded by Plan Nacional I+D+i from the Spanish Government.

## References

[1] 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, Springer International, Grenoble, France, 2024.

[2] J. Parapar, P. Martín-Rodilla, D. E. Losada, F. Crestani, Overview of erisk 2024: Early risk prediction on the internet (extended overview), in: Working Notes of the Conference and Labs of the Evaluation Forum CLEF 2024, Grenoble, France, September 9th to 12th, 2024, CLEF 2024, CEUR Workshop Proceedings, 2024.

[3] J. Parapar, P. Martín-Rodilla, D. E. Losada, F. Crestani, Overview of erisk 2023: Early risk prediction on the internet, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2023, pp. 294–315.

[4] D. E. Losada, F. Crestani, J. Parapar, Overview of erisk 2018: Early risk prediction on the internet (extended lab overview), in: Proceedings of the 9th International Conference of the CLEF Association, CLEF, 2018, pp. 1–20.

[5] D. E. Losada, F. Crestani, J. Parapar, Overview of erisk 2019 early risk prediction on the internet, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction: 10th International Conference of the CLEF Association, CLEF 2019, Lugano, Switzerland, September 9–12, 2019, Proceedings 10, Springer, 2019, pp. 340–357.

[6] P. Martın-Rodilla, D. E. Losada, F. Crestani, Overview of erisk 2022: Early risk prediction on the internet, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction: 13th International Conference of the CLEF Association, CLEF 2022, Bologna, Italy, September 5–8, 2022, Proceedings, volume 13390, Springer Nature, 2022, p. 233.

[7] A. Pérez, J. Parapar, A. Barreiro, S. Lopez-Larrosa, Bdi-sen: A sentence dataset for clinical symptoms of depression, in: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 2996–3006.

[8] T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama, Optuna: A next-generation hyperparameter optimization framework, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2019.

[9] K. Kroenke, T. W. Strine, R. L. Spitzer, J. B. Williams, J. T. Berry, A. H. Mokdad, The phq-8 as a measure of current depression in the general population, Journal of affective disorders 114 (2009) 163–173.

[10] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, Advances in neural information processing systems 33 (2020) 1877–1901.

[11] 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.

[12] 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, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[13] S. M. Mohammad, P. D. Turney, Nrc emotion lexicon, National Research Council, Canada 2 (2013) 234.

[14] S. Poria, A. Gelbukh, E. Cambria, A. Hussain, G.-B. Huang, Emosenticspace: A novel framework for affective common-sense reasoning, Knowledge-Based Systems 69 (2014) 108–123.

[15] J. Staiano, M. Guerini, Depechemood: a lexicon for emotion analysis from crowd-annotated news, arXiv preprint arXiv:1405.1605 (2014).

[16] P. Ekman, An argument for basic emotions, Cognition & emotion 6 (1992) 169–200.

[17] H. Fabregat, A. Duque, L. Araujo, J. Martínez-Romo, Uned-nlp at erisk 2022: Analyzing gambling disorders in social media using approximate nearest neighbors., in: CLEF (Working Notes), 2022, pp. 894–904.