Representation Exploration and Deep Learning Applied to the Early Detection of Pathological Gambling Risks⋆

Notebook for the eRisk Lab at CLEF 2023

Xabier Larrayoz[1,†], Nuria Lebeña[1,*], Arantza Casillas[1] and Alicia Pérez[1]

[1]HiTZ Center - Ixa, University of the Basque Country (UPV/EHU), 20080 Donostia

Abstract

The aim of Task 2 (Early Detection of Signs of Pathological Gambling) from the CLEF 2023 eRisk Workshop is to analyze social media users' messages for early warning signs of pathological gambling. Given that Pathological Gamblers are a small set compared to the Control group, we propose the utilization of a neural network incorporating a customized loss function to effectively tackle the challenge of class imbalance. In our proposed loss function it is possible to adjust the penalty for false positives and false negatives, increasing the penalty for the critical false negatives. Our proposed solution demonstrates robustness, achieving one of the highest recall rates while maintaining a competitive precision. Furthermore, our system introduces a range of potential variations that warrant further investigation.

Keywords

Early risk prediction, Natural Language Processing, Class imbalance, Deep learning, Mental health

1. Introduction

The different editions of the eRisk workshop serve as a meeting point where different methodologies and practical approaches for early detection of different types of health risks are proposed. The various tasks focus on the textual analysis of posts and messages from social media users. Social media plays a crucial role in the early diagnosis of mental illnesses and other health problems. Many people use social media regularly posting their inner thoughts and daily experiences, sometimes showing different symptoms and signs of mental illnesses and therefore being indicative of potential mental disorders. Given that in many cases mental health is still perceived as taboo in our society, many patients don't receive the necessary medical care, as a consequence, social media emerges as a good data source to analyze mental illnesses and be able to help potential patients.

In this article, we present a system to address Task 2 of the eRisk Workshop 2023: Early Detection of Signs of Gambling Addiction [1]. The approach is based first on generating

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece

*Corresponding author.

†These authors contributed equally.

xlarrayoz001@ikasle.ehu.eus (X. Larrayoz); nuria.lebena@ehu.eus (N. Lebeña); arantza.casillas@ehu.eus (A. Casillas); alicia.perez@ehu.eus (A. Pérez)

vector representations of user messages through sentence embedding, and then on detecting positive messages using deep learning-based methods. Additionally, an original loss function is introduced to deal with imbalanced classes or cases where false negatives have a significant impact.

The rest of the article is structured as follows: Section 2 provides an overview of previous work related to the task considered and the techniques used in this study. Section 3 describes the tackled task, including the available dataset and evaluation metrics, while the developed system is presented in Section 4. Section 5 demonstrates the configuration of the variants used for the competition. The results obtained are compared with those of other participating systems and discussed in Section 6. Finally, Section 7 presents the main conclusions and future lines of work.

2. Related work

In recent years, the use of AI to address mental health-related tasks has gained greater presence. AI, and in particular NLP, has proven to be a powerful tool in the detection of mental disorders. In previous studies, NLP has been used on electronic health records to assist in the identification of suicidal behaviors [2, 3], achieving an accuracy of 0.47.

In this context, most of the methods used for the detection of mental disorders are based on traditional Machine Learning techniques, such as SVM, AdaBoost or Decision Trees. The recent interest in Deep Learning has shown a better performance [4]. However, as stated by Zhang et al. [4] a large part of the solutions proposed are concentrated on a few mental disorders. As one of the most prevalent disorders in the world, the absence of research on the identification of gambling addiction stands out.

The SMM4H [5], CLEF and CLPsych [6] competitions focus on the application of machine learning techniques in the field of mental health. In 2022, the winner of the CLPsych competition achieved 68.9% accuracy in detecting mood changes in tweets. Years earlier, in 2019, a similar rate was recorded in identifying users at risk of suicide based on their messages.

For several years, one of the tasks to be addressed in CLEF eRisk has been the early detection of pathological gambling risks. By sequentially processing user interactions on social networks, the system was to detect the first signs of pathological gambling as early as possible.

Very different approaches were used in the previous edition. The BLUE group, proposed to train a BERT classifier, using an additional dataset generated from some Reddit mental health communities. The UNED-NLP [7] team, participated with a system that relied on Approx Nearest Neighbors techniques to detect positive messages. The SINAI [8] group came up with a design based on language features. Using the last 50 user messages, a vector was obtained which was complemented with message features such as number of words, lexical diversity and sentence complexity. Finally it was passed through a feed-forward neural network (FFNN) model. Similarly, in our work we propose a FFNN feed by a semantic representation of sentences.

3. Dataset

The dataset provided in this edition, is the combination of the data used in the two previous editions (eRisk 2021 and 2022). It is composed of a number of XML files, each of which contains a number of posts made by 4,427 users of social networks. It includes 2,298,412 messages overall. Labels are user level and they design them as either gamblers (1) or control users (0).

In Figure 1, it is evident that both editions exhibit a significantly imbalanced dataset, indicating a non-uniform data distribution. Specifically, the proportion of players classified as gambling addicts in their respective sets does not exceed 7% or 4%. Previous research in supervised classification has highlighted class imbalance, referred to as class skew or class imbalance, as a critical factor that significantly impedes the capacity of inference algorithms to learn and accurately generalize the minority class.

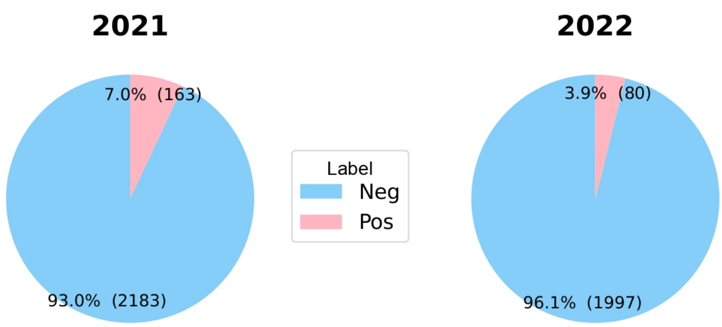

Figure 1: Class distribution of the training sets made available in eRisk 2021 and 2022. Pos label stands for Pathological Gambler and Neg for Control subjects.

In order to enhance the message quality during the training process, a filtering approach was employed. Specifically, users with fewer than 10 messages and the top 20 users with the highest presence in the dataset were excluded. This strategic exclusion aimed to mitigate the potential noise introduced by these users, thus improving the overall training efficacy.

The test set officially employed to assess the challenge, described in Table 1, is sent to the participants iteratively through a connection to a server. The total number of users is 2174, of which 103 are compulsive gamblers.

Table 1
Task 2 Pathological gambling. Main statistics of test collection for task 2 early detection of signs of pathological gambling

	Pathological Gamblers	*Control*
Num. subjects	103	2071
Num. submissions (posts & comments)	33.7×10^3	1.07×10^6
Avg num. of submissions per subject	327	516
Avg num. of days from first to last submission	≈ 675	≈ 878
Avg num. words per submission	28.9	20.5

4. Methods

Given a sequence of consecutive l posts post messages written by subject k, denoted as $\mathbf{t}_k = (t_k^1, t_k^2, \ldots, t_k^l)$ with t_k^j being the j-th post (i.e. text) in the succession, the aim is to get a subject-level classification label ($\hat{u}_k$) to distinguish Pathological Gamblers from Control subjects. To that end, in our approach each post (t_k^j) is processed and a post-level label (c_k^j) computed by means of the architecture presented in Figure 2. With this information is attained, next, the user-level label. The processes involved and training strategies applied are detailed in the following sections.

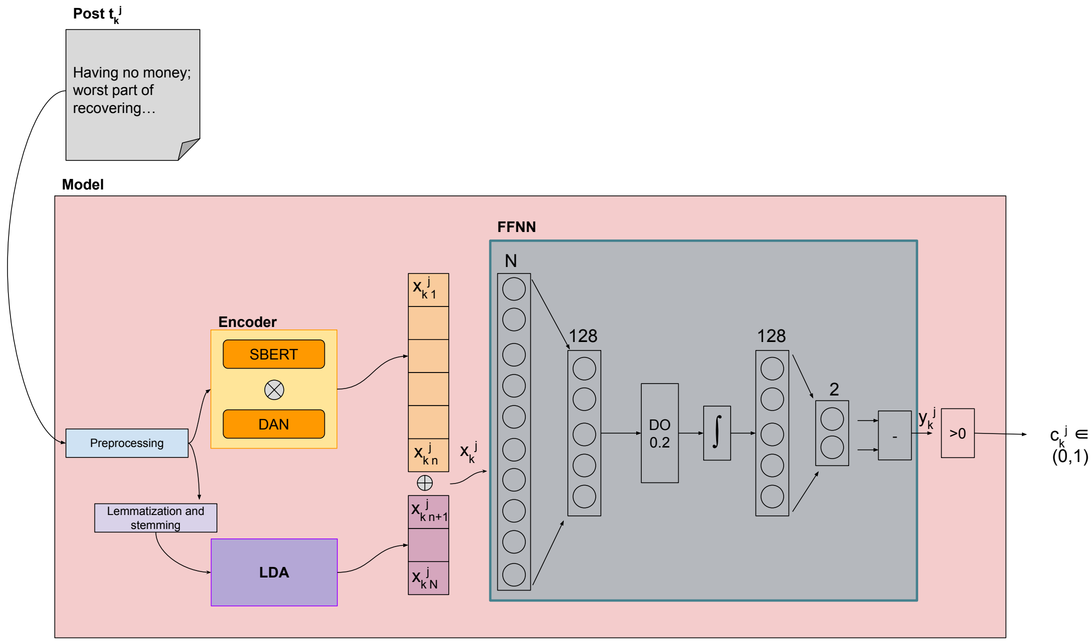

Figure 2: Design of the model for the binary post-level classification. The post j of a user k (t_k^j) is represented as a numeric array ($x_{k1}^j, \ldots, x_{kN}^j$). The input of the FFNN network is the concatenation of the vector generated by the Encoder and the LDA model. The output of the FFNN (c_k^j) is the estimated post-label.

4.1. Text pre-processing

Each of the posts has two parts, a title and a body. We combined both parts to create a single message. We conducted the following steps in order to preprocess the posts:

- Conversion to lowercase
- Characters cleaning
- Stopwords removal

Lemmatization and stemming were also incorporated prior to LDA in order to improve the generated topics. Table 2 shows the versions of a text after applying different preprocessing techniques. The last version would be necessary in the case of applying LDA.

Table 2
Comparison of the original text and the preprocessed versions, where Preprocessing refers to applying the transformations mentioned in the previous list, and Lem. & Stem. generates the base form.

Original	**Preprocessing**	**Lemmatization & stemming**
Having no money; worst part of recovering	money worst part recovering	money worst part recover

4.2. Post-level vector representation

We need to get each text-post, t_k^j, converted into a fixed-size numeric vector, $\mathbf{x}_k^j = (x_{k1}^j, x_{k2}^j, \dots, x_{kN}^j) \in \mathbb{R}^N$ that would serve as the input to the FFNN. Note that $t_k^j \in \Sigma^*$ being Σ the input vocabulary. Two main strategies were explored in order to get a numeric representation ($\mathbf{x}_k^j$) given a post (t_k^j): encoding and LDA. We can either use just one strategy or both and make use of the concatenated representation leading, thus, to a longer vector representation, as in (1), in which the vectorization by means of the encoder led to an $N_{encoder} = n$-dimensional array, $v_{encoder}(t_k^j) = (x_{k,1}^j, x_{k,2}^j, \dots, x_{k,n}^j)$ and the LDA yielded an $N_{LDA} = N - n + 1$ dimensional array $v_{LDA}(t_k^j) = (x_{k,n+1}^j, x_{k,n+2}^j, \dots, x_{k,N}^j)$.

The text, represented as an array of fixed size with the dimension (N) depending on the representation used (either encoding or LDA or both). The resulting vector representation $\mathbf{x}_k^j \in \mathbb{R}^N$ is, indeed, the input for the classifier.

$$
\begin{aligned}
v : \Sigma^* &\longrightarrow \mathbb{R}^N \\
t_k^j &\longrightarrow v(t_k^j) = (x_{k,1}^j, x_{k,2}^j, \dots, x_{k,n}^j, x_{k,n+1}^j, x_{k,n+2}^j, \dots, x_{k,N}^j) = \mathbf{x}_k^j
\end{aligned}
\tag{1}
$$

In what follows, details are given of each strategy explored to obtain the representation, the encoder in section 4.2.1 and LDA in section 4.2.2.

4.2.1. Post-level encoder

We wondered whether Dynamic Aggregation of Network (DAN) or Sentence-BERT (SBERT) would generate a better semantic representation of the texts. Models such as the Universal Sentence Encoder (USE) [9], Sentence-BERT (SBERT) [10] and Transformer-based Pretrained Language Models (PML), allow a complex and global representation taking into account word interactions and relations. SBERT is a variation of the traditional BERT [11] model that incorporates Siamese and triplet lattice structures. Such structures enable learning the similarities and contrasts between various inputs.

USE, on the other hand, bases its architecture on convolutional and recurrent neural networks. There are two USE variants, the most widely used is based on transformers. Nevertheless the variant known as the Dynamic Aggregation of Network (DAN) that makes use of the dynamic aggregation of networks approach to enhance the outcome has shorter computing time.

We represented the posts using DAN and SBERT in order to compare their performance. The vector of size ($N_{encoder}$) generated by DAN encoder is 512 while SBERT works with a vector of size 384.

4.2.2. Topic modeling

Latent Dirichlet Allocation (LDA) is a probabilistic model able to identify the latent topics in the posts. It enables to extract the topic distribution of each post and we used it as additional features to represent the post. We configured LDA to extract 20 topics from the posts, leading to a representation $\mathbf{x}_k^j \in \mathbb{R}^{20}$.

4.3. Post-level classification

The estimated post-label-confidence $y_k^j \in \mathbb{R}$ is an intermediate scalar obtained in our system interpreted as the confidence score that the post t_k^j contains traits of language related to Pathological Gambling. When y_k^j takes positive values, the intermediate post-level label c_k^j is assigned to class 1; otherwise, it is assigned to class 0, that is, $g(z) = sign(z)$.

Note that the $f(\cdot)$ transformation is attained by means of the FFNN network. These processes are formally summarized in (2) and graphically depicted in Figure 2.

$$
\begin{array}{rcccl}
g \circ f : \mathbb{R}^N & \longrightarrow & \mathbb{R} & \longrightarrow & \{0, 1\} \\
\mathbf{x}_k^j & \longrightarrow & f(\mathbf{x}_k^j) = y_k^j & \longrightarrow & g(y_k^j) = c_k^j
\end{array} \tag{2}
$$

With regard to the practical details of the FFNN network, we would like to mention that it includes two layers and contains a total of 49538 to 65922 parameters, depending on the encoder employed, respectively, SBERT or DAN. If LDA is used, there are 2560 additional parameters.

In what the training process is concerned, the FFNN was trained on 5 epochs and setting the learning rate to 5×10^{-5}. Besides, in the training stage the dropout was set to 0.2 and AdamW optimizer was employed as it was proven effective [12]. An iterative training approach was implemented where, in each epoch, all the messages from a user are sequentially processed to update the parameters.

4.4. Subject-level classification

User gold label u_k is the user label (either 'Control' or 'Pathological Gamblers') for k-th subject as in the gold-standard, that is, the expected label for the subject. With the information attained at post-level (as stated in section 4.3), the user-level label is estimated ($\widehat{u_k}$). The performance of the system is, indeed, assessed based on the difference between predicted ($\widehat{u_k}$) and expected (u_k) subject-level labels.

Note, however, that there is a subtlety in the arrangement of the task: not all the post of the user k are given jointly, instead, the posts are presented to the system in sequentially, one by one in their turn. That is, for user k in the time-stamp 1 we merely count on the post t_k^1, while by the l-th time-stamp we would have seen a sequence of l posts, $(t_k^1, t_k^2, \ldots, t_k^l)$. For each post the system must provide a user-level assessment. Hence, by the l-th round, the system has provided a sequence of l outputs $\widehat{\mathbf{u}}_k = (\widehat{u}_k^1, \widehat{u}_k^2, \ldots, \widehat{u}_k^l)$.

By the time the system needs to compute current user-level label $\widehat{u}_k^{l+1}$, it counts with current post t_k^{l+1} and all the history:

- past sequence of posts and, inherently, their corresponding encoding: $\mathbf{t}_k = (t_k^1, t_k^2, \ldots, t_k^l)$ and $(v(t_k^1), v(t_k^2), \ldots, v(t_k^l))$
- past sequence of computed post-level confidence scores: $\mathbf{y}_k = (y_k^1, y_k^2, \ldots, y_k^l)$
- past sequence of estimated post-level labels: $\widehat{\mathbf{c}}_k = (c_k^1, c_k^2, \ldots, c_k^l)$

All this information is available and can be employed to generate $\widehat{u}_k^{l+1}$ as stated in (3).

$$\begin{array}{rcl} h : \Sigma^* \times (\Sigma^*)^l \times \mathbb{R}^l \times \{0,1\}^l & \longrightarrow & \{0,1\} \\ (t_k^{l+1}, \mathbf{t}_k, \mathbf{y}_k, \widehat{\mathbf{c}}_k) & \longrightarrow & h(t_k^{l+1} | \mathbf{t}_k, \mathbf{y}_k, \widehat{\mathbf{c}}_k) = \widehat{u_k}^{l+1} \end{array} \quad (3)$$

In our approach, however, the user-level label $\widehat{u}_k^{l+1}$ is estimated as in (4).

That is, we computed the user-level label relying merely on the current post-level label and disregarding the historic information. Needless to say, future efforts could be devoted to leverage $h(\cdot)$ exploiting all the information available.

$$\widehat{u_k}^{l+1} = h(t_k^{l+1} | \mathbf{t}_k, \mathbf{y}_k, \widehat{\mathbf{c}}_k) = g(f(v(t_k^{l+1}))) = \widehat{c}_k^{l+1} \quad (4)$$

4.5. Silver-standards explored as post-level reference

Since gold-labels are provided at user level and we trained our FFNN using posts, we needed to get the posts labeled for the training. That is, in the training stage the estimated post-label confidence (y_k^j) must be compared to a desired or expected confidence ($\mathbf{y'}_k$) the underlying issue rests on the fact that the post-level confidence is not given. This silver standard is shown in Figure 3 As the reference post-label confidence, $\mathbf{y'}_k = (y'^1_k, \ldots, y'^l_k)$, in this work we explored two alternative silver-standard assignation strategies:

- User-based message labeling (UBL): Consists of assigning each post the label of the user. That is, if a user is positive all it's posts will be labeled as positive, that is, all the components in this array, y'^l_k, equal to u_k.
- Approximated Nearest Neighbors (ANN): Posts are labeled using an iterative labeling approach. First, each post is labeled with the label of the user. Then using ANN technique, the labels are reassigned, giving to the closer posts the same labels, getting $\mathbf{y'}_k$ as in [7].

The heuristic post-level references employed have a deep impact in the training stage and, needless to say, should be selected carefully. Future work can be addressed in alternative reference assignment strategies.

4.6. Loss function across post-level labels to improve user-level label

As mentioned in section 3, the data, far from being uniform, exhibit a noticeable imbalance. As a consequence, the neural network can become biased and achieve low precision in the minority class. To address this problem, there are various strategies to alter the class proportion through over-sampling and under-sampling. Following an approach closer to assigning class weights, we have chosen to apply a loss function based on cross-entropy during the training of the neural network. The strategy followed in our work is sketched in Figure 3.

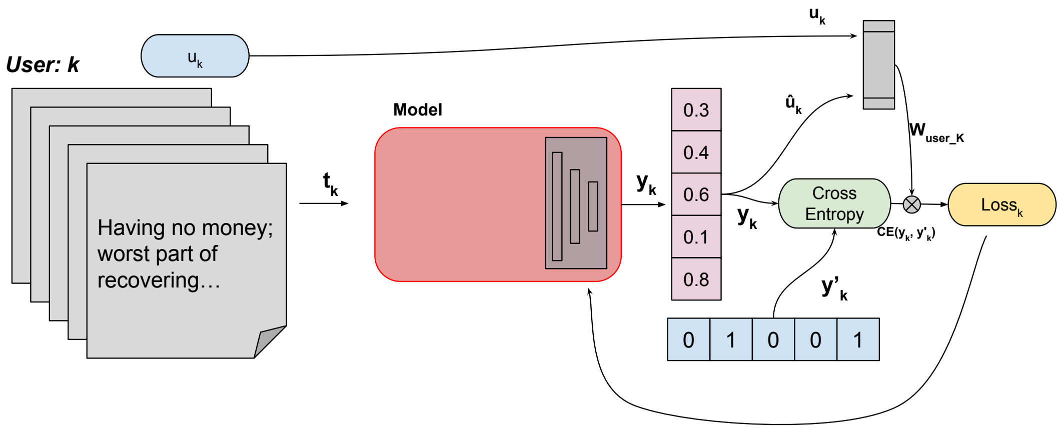

Figure 3: System training approach. Cross Entropy is employed to compute the loss function and update the model for a given user. A zoom in the model was given in Figure 2.

With the sequence of posts from user k a sequence of confidence scores is computed by the model, post by post and $\mathbf{y}_k$ obtained. With this, as mentioned in section 4.3, a sequence of post-level labels are computed, that is, $\widehat{\mathbf{c}}_k$. In the training stage, the system estimates a user-level label taking into account all the post-level labels, as in (5), meaning that a positive post-level label in the sequence suffices to classify the user as positive. In the training stage, the user-level label is estimated comprising all the posts-level labels from the user. The estimated user-level label can be compared to the ground-truth provided (u_k) to update the model in the training stage.

$$\widehat{u_k} = \begin{cases} 1, & \text{if } \exists i \quad 1 \leq i \leq l \quad : \quad c_k^i = 1 \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

In the training stage, the sequence of labels computed ($\mathbf{y}_k$) are compared with the silver-labels proposed as reference (that is, the $\mathbf{y'}_k$ labels presented in section 4.5). This comparison is quantitatively seized as the loss by means of the Cross Entropy Loss function, $H(\mathbf{y'}_k, \mathbf{y}_k)$, as implemented in PyTorch [12]. The user-dependant loss is seized by a weight factor that allows for a penalty as in (6).

$$Loss_k = W_{user_k} \cdot H(\mathbf{y'}_k, \mathbf{y}_k) \tag{6}$$

Our training approach does not penalize equally false positives and false negatives, indeed the weight factor employed by our team is given in (7).

$$W_{user_k} = \begin{cases} 4 & \text{if } \widehat{u_k} = 0 \wedge u_k = 1 \\ 2 & \text{if } \widehat{u_k} = 1 \wedge u_k = 0 \\ 1 & \text{if } \widehat{u_k} = u_k \end{cases} \tag{7}$$

In the experiments, a penalty of 2 was used for false positives and 4 for false negatives.

The critical false negatives were penalized by doubling the loss in those cases where the system incorrectly predicted a positive instance. These values (1, 2 and 4) have been determined through a sensitivity analysis and based on the objective of prioritizing false positives over false negatives in an attempt not to miss Pathological Gamblers.

There is room for improvement in the training stage. On the one hand, the user-level labeling strategy, i.e. (5), for instance, could be computed taking the time-stamp into account and not just the sequence of post-labels, however, the function proposed is computationally cheap and suited to tackle class-imbalance. On the other hand, the loss function and the penalty weight.

5. Experimental framework

5.1. Practical details

Preliminary experiments were conducted combining all the variants mentioned in the methodological framework, varying: the post level vector representation (involving SBERT, DAN and LDA as mentioned in section 4.2); alternative silver-standard approaches to get a post-level ground-truth (UBL and ANN as detailed in section 4.5); the data-set partition employed to train the system (exploring the 2021 or 2022 and sets as mentioned in section 3). user-penalty weights in the training stage (developed in section 4.6). These preliminary experiments were conducted on a partition of the provided data split by year and also combined (see Figure 1). These experiments led us to select the parametrization e.g. the penalty weights presented in expression 7 and the same modified loss function. We found that DAN outperformed SBERT and this is why we, eventually, discarded that encoding.

As a result, in our team, a total of 5 runs were submitted with the configurations detailed in Table 3. Given the limitation on the number of runs to submit, greater diversity in configurations was prioritized over variants using SBERT. Additionally, a combined model was used, which utilizes all the mentioned variants to create a single result. The decision made by the combined model is determined through an OR operation. Accordingly, it is, both necessary and sufficient, for one of the variants to estimate a positive user-level label for this combined approach to estimate positive. T he motivation behind this combined approach is to leverage the Recall even at the expense of certain false positives, which are preferable to false negatives.

5.2. Results

In Table 4, we can see some of the best results from the competition in the main task. This allows us to compare our proposed system, Xabi_EHU, with other submissions. Among all the system variations, the first configuration has achieved competitive scores in all metrics, with a strong

Table 3
Submitted Runs: Description of the configurations explored. The second column refers to the encoding strategy (explained in section 4.2.1), LDA refers to the incorporation of LDA in the post-level vector representation (as in 4.2.2), Label indicates the labeling used in training (Sec. 4.5), and finally the edition of the training set used (Sec. 3).

Run	DAN vs SBERT	LDA involved	Label	Train
0	DAN	No	UBL	2021
1	DAN	Yes	UBL	2021
2	DAN	No	ANN	2021
3	DAN	No	UBL	2021 ∪ 2022
4	$OR_{i=0}^{3}(Run_i)$			

dominance in Recall. In an environment where false negatives have a significant impact, having a high Recall is of great interest in real-world applications, compensating for small differences in other measures. On the other hand, as seen during development, the use of LDA has resulted in a general performance loss for the system, while training based on more precise labeling has led to a decrease in the system's precision. Overall, these results suggest that there are multiple effective approaches to addressing the problem. Note that, while somehow simplistic, the UBL approach resulted in sensitive.

Table 4
Decision-based evaluation for Task 2. Our Team is denoted as Xabi_EHU and the details of each run were presented in Table 3.

Team	Run	P	R	F1	$ERDE_5$	$ERDE_{50}$	latencyTP	speed	latency-weighted F1
ELiRF-UPV	0	**1.000**	0.883	**0.938**	**0.026**	**0.010**	**4.0**	**0.988**	**0.927**
Xabi_EHU	0	0.846	**0.961**	**0.900**	**0.030**	**0.012**	**8.0**	**0.973**	**0.875**
Xabi_EHU	1	0.89	0.864	0.877	0.035	0.017	12.0	0.957	0.839
Xabi_EHU	2	0.79	0.913	0.847	0.036	0.015	13.0	0.953	0.807
Xabi_EHU	3	0.829	0.942	0.882	0.033	0.013	12.0	0.957	0.844
Xabi_EHU	4	0.756	**0.961**	0.846	0.031	0.013	**8.00**	**0.973**	0.823
UNSL	2	0.752	0.854	0.800	0.048	0.013	14.0	0.949	0.759
BioNLP-IISERB	0	**0.933**	0.68	0.787	0.038	0.037	62.0	0.766	0.603
NLP-UNED-2	1	**0.957**	0.883	**0.919**	0.034	0.016	13.0	0.953	**0.876**

Regarding the additional task of user ranking, Table 5 shows the performance of the top teams in that task. As can be seen, prioritizing better Recall has led to inferior performance compared to other teams. Without additional information, using the class probability itself as the user's risk level may not be a good indicator for what the competition intended.

Table 5
Ranking-based evaluation for Task 2. Our team is denoted as Xabi_EHU.

		1 writing			100 writing			500 writing			1000 writing		
Team	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
Xabi_EHU	0	1.00	1.00	0.57	1.00	1.00	0.50	0.80	0.88	0.41	0.90	0.94	0.41
OBSER-MENH	0	1.00	1.00	0.64	1.00	1.00	0.55	1.00	1.00	0.48	1.00	1.00	0.50
ELiRF-UPV	0	1.00	1.00	0.59	1.00	1.00	0.91	1.00	1.00	0.95	1.00	1.00	0.94
UNSL	1	1.00	1.00	0.57	1.00	1.00	0.78	1.00	1.00	0.67	1.00	1.00	0.70
NLP-UNED-2	3	1.00	1.00	0.59	1.00	1.00	0.92	1.00	1.00	0.95	1.00	1.00	0.93

6. Conclusions

Our participation focuses on early detection of signs of pathological gambling. Given a sequence of posts in a sequence, one by one, the aim is to estimate, employing as fewer posts as possible, the subject-level label (either Control or Pathological Gambler). With the data-set highly skewed, and being the target Pathological Gambler the minority group, we struggled to find robust models and focused on maximizing the recall not to miss pathological gamblers. However, this led us to the creation of a specific loss function. The data imbalance has also prevented us from using ML-based architectures, as any attempts to generalize the data and achieve satisfactory performance had failed. This effect can be observed in the metric values separated by classes, where results close to 1 are obtained for the negative class, unlike the case of the positive class. Our approach was designed with the aim to avoid over-fitting. Basically, the runs submitted employed DAN encoder and an FFNN. The training was enhanced by means of a loss function defined by user. The user-level label is estimated by means of a post-level label, and this strategy required us to figure out, heuristically, the post-level label reference to train the system. Data imbalance has been a challenge throughout the development process and, as expected, the model obtained better results for the majority class (Control users).

The proposed system has achieved competitive performance in the tasks of binary classification and ranking-based classification.

Since the model development has focused exclusively on the main task of binary classification, the performance on the ranking task is lower.

Needless to say, there is room for improvement in the proposed approach. We feel motivated to keep exploring variants to improve the user-level label estimation employed all the pieces of information available at each stage re-defining (3) not in the simple manner we did in (4). There are other core-issues to bear in mind, such as the definition of a reference for the post-level confidence that are worth exploring. In any case, the approach proposed is versatile. This same design can be extrapolated to other mental disorders, even with texts in other languages, using the corresponding encoder. The weights of the modified loss function will vary according to the class balance, but with an unaltered loss function, the model could be competitive.

Acknowledgments

This work was elaborated within the framework of LOTU (TED2021-130398B-C22) funded by MCIN/AEI/10.13039/501100011033, European Comission (FEDER), and by the European Union "NextGenerationEU"/PRTR. Besides, this work was partially funded by the Spanish Ministry of Science and Innovation (DOTT-HEALTH/PAT-MED PID2019-106942RB-C31); by the Basque Government (IXA IT-1570-22, Predoctoral Grant PRE-2022-1-0069) and Ikasiker grants published in the 11/07/2022 BOPV; and by EXTEPA within Misiones Euskampus 2.0.

References

[1] J. Parapar, P. Martín-Rodilla, D. E. Losada, F. Crestani, Overview of erisk 2023: Early risk prediction on the internet, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction: 14th International Conference of the CLEF Association, CLEF 2023, Springer International Publishing, Thessaloniki, Greece., 2023.

[2] N. J. Carson, B. Mullin, M. J. Sanchez, F. Lu, K. Yang, M. Menezes, B. L. Cook, Identification of suicidal behavior among psychiatrically hospitalized adolescents using natural language processing and machine learning of electronic health records, PloS one 14 (2019) e0211116.

[3] B. L. Cook, A. M. Progovac, P. Chen, B. Mullin, S. Hou, E. Baca-Garcia, Novel use of natural language processing (nlp) to predict suicidal ideation and psychiatric symptoms in a text-based mental health intervention in madrid, Computational and mathematical methods in medicine 2016 (2016).

[4] T. Zhang, A. Schoene, S. Ji, S. Ananiadou, Natural language processing applied to mental illness detection: a narrative review, NPJ Digital Medicine 5 (2022). doi:10.1038/s41746-022-00589-7.

[5] D. Weissenbacher, J. Banda, V. Davydova, D. Estrada Zavala, L. Gasco Sánchez, Y. Ge, Y. Guo, A. Klein, M. Krallinger, M. Leddin, A. Magge, R. Rodriguez-Esteban, A. Sarker, L. Schmidt, E. Tutubalina, G. Gonzalez-Hernandez, Overview of the seventh social media mining for health applications (#SMM4H) shared tasks at COLING 2022, in: Proceedings of The Seventh Workshop on Social Media Mining for Health Applications, Workshop & Shared Task, Association for Computational Linguistics, Gyeongju, Republic of Korea, 2022, pp. 221–241. URL: https://aclanthology.org/2022.smm4h-1.54.

[6] A. Tsakalidis, J. Chim, I. Bilal, A. Zirikly, D. Atzil Slonim, F. Nanni, P. Resnik, M. Gaur, K. Roy, B. Inkster, J. Leintz, M. Liakata, Overview of the clpsych 2022 shared task: Capturing moments of change in longitudinal user posts, in: Proceedings of the Eighth Workshop on Computational Linguistics and Clinical Psychology, Association for Computational Linguistics, Seattle, USA, 2022, pp. 184–198. doi:10.18653/v1/2022.clpsych-1.16.

[7] 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: Conference and Labs of the Evaluation Forum, 2022.

[8] A. M. Mármol-Romero, S. M. Jiménez-Zafra, F. M. Plaza-Del-Arco, M. D. Molina-González, M.-T. Martín-Valdivia, A. Montejo-Ráez, Sinai at erisk@clef 2022: Approaching early

detection of gambling and eating disorders with natural language processing, in: CEUR Workshop Proceedings, volume 3180, CEUR-WS, 2022, pp. 961–971.

[9] D. Cer, Y. Yang, S. yi Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, Y.-H. Sung, B. Strope, R. Kurzweil, Universal sentence encoder, 2018. arXiv:1803.11175.

[10] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, 2019. arXiv:1908.10084.

[11] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[12] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, Pytorch: An imperative style, high-performance deep learning library, in: Advances in Neural Information Processing Systems 32, Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.