Interpretable Learning for Detection of Cognitive Distortions from Natural Language Text

Anonymous ACL submission

Abstract

We developed a technology that, based on a dataset annotated for cognitive distortions, builds an interpretable model capable of detecting cognitive distortions in natural language texts. The learning and detection technologies are based on structural pattern (N-gram) matching with the "priority on order" principle. We investigated and released two types of detection models: plain binary classification and a model based on a multi-class representation. We optimized the hyper-parameters of the models and achieved an accuracy of 0.92 and an F1 score of 0.95 in a cross-validation experiment. Additionally, we achieved over 1000 times higher performance and lower computational cost compared to LLM-based alternatives.

1 Introduction

In cognitive-behavioral therapy (CBT), cognitive distortions are identified as key indicators for monitoring a person's psychological state. These are systematic errors in thinking that occur unconsciously and automatically, leading to inaccuracies in judgment and irrational behavior, thereby influencing decision-making and information interpretation. The term was first introduced by Aaron Beck, the creator of CBT (Beck, 1963).

Cognitive distortions can arise from subjective beliefs, stereotypes, social influences, and emotional factors. They showed psychological issues such as depressive disorder (Bathina et al., 2021), anxiety disorder (Al-Mosaiwi and Johnstone, 2018), and post-traumatic stress disorder (PTSD) (Ouhmad et al., 2023). Moreover, they were found to be closely linked to historical events not only at the individual level but also at the societal and national levels. As demonstrated in Bollen et al. (2021), there has been a significant increase in the occurrence of cognitive distortions in literature since the year 2000.

With the development of technology and the rise of social networks and messaging apps, people now receive large amounts of textual information far more frequently than before. As mentioned earlier, cognitive distortions can arise due to social influence and stereotypes perpetuated in society. For example, if a person reads news every day that states, "everything is bad," "nothing will change," or "everyone around is foolish," they will inevitably start adopting these linguistic patterns in their thinking and speech over time. Therefore, it becomes essential to develop the ability to detect cognitive distortions in natural language text.

To date, the solutions found are not interpretable, require significant time or computational resources, and generally achieve a maximum accuracy of 0.6–0.9. Therefore, our goal was to develop a more efficient model without compromising accuracy. We developed a technology that, based on a dataset, built an interpretable model for detecting cognitive distortions in natural language texts, represented by lists of N-grams , corresponding to specific distortions. This model demonstrated performance higher than most of neural network-based solutions and comparable to the best of them, achieving an accuracy of 0.92 and an $F1$ score of 0.95. The model learning approach is based on analyzing the frequency of N-grams in texts, associated with target distortion. We used different metrics derived from the N-gram frequency counts, such as $TF - IDF$, mutual information and ones from the work by (Kolonin, 2022), as described in section 4. Additionally, we explored optimal values for the associated N-gram length and the threshold for filtering out irrelevant words and phrases.

The technology we developed for creating an interpretable model to detect cognitive distortions in text can become a universal and easily adaptable tool, applicable in various fields. For example, if we collect a dataset of posts from Twitter and Reddit on the topic of cryptocurrencies and use

our technology to create a model for detecting cognitive distortions in these texts, we could predict market movements with some accuracy, similar to the work of Raheman et al. (2022). Our technology could also be applied in psychology (Calvo et al., 2017) to support beginner psychologists in tracking cognitive distortions in clients. It is known that psychologists, depending on their CBT school, classify between 6 (Beck, 1976) and 50 (Boyes, 2013) cognitive distortions in their practice. Thanks to the flexibility of our technology, an interpretable model can be created for any classification system in use. If training is conducted on a dataset containing posts from people prone to anxiety and depression disorders, the resulting model could be used to track these negative states in individuals online and potentially warn about more severe depression and anxiety conditions.

2 Related Work

Currently, several studies addressed the problem of detection of cognitive distortions from natural language text. These works can be conditionally divided into those that solve only the binary classification task, and those that solve both binary and multi-class classification tasks. The most common approaches include logistic regression and neural network-based models, such as modifications of BERT.

In the study Simms et al. (2017), a logistic regression model was used with LIWC features selected using RELIEF. The resulting accuracy for binary classification was 0.73. Similarly, in Shickel et al. (2019), logistic regression was applied, but with TF-IDF-based features. This model achieved an accuracy of 0.9 and an F1 score of 0.88 for the binary classification task.

In (Shreevastava and Foltz, 2021), the authors applied a support vector machine (SVM) classifier using contextual embeddings extracted with a pre-trained S-BERT model. This approach achieved an F1 score of 0.79 for binary classification and 0.3 for multi-class classification. For the same task, the study (Singh et al., 2024) applied a Large Language Model (LLM), specifically LLAMA-7b, which achieved an accuracy of 0.84 and an F1 score of 0.80. Additionally, the studies (Tauscher et al., 2023) and (Wang et al., 2023) presented BERT-based models that reached F1 scores of 0.62 and 0.77, respectively. Finally, the RoBERTa-based model described in (Babacan et al., 2023) achieved the highest results to date for the binary classification task, with an accuracy of 0.973 and an F1 score of 0.951. This work was further extended to solve the multi-class classification task in (Babacan et al., 2025), where the model achieved accuracy = 0.95 and average F1 score = 0.95 on synthetic data. However, our model achieved comparable performance (accuracy = 0.92 and F1 score = 0.95), while also being interpretable and computationally efficient.

It is worth noting that most of the related studies did not provide access to the datasets they used, making it impossible to reproduce their results and directly compare them with ours. At the time of the study only two works mentioned above — (Shreevastava and Foltz, 2021) and (Babacan et al., 2023) — provided publicly available datasets. Therefore, in our further analysis, we focused on these two datasets.

3 Data

For the purpose of the study we selected two original datasets with labeled cognitive distortions in patient texts in English language found in the earlier works.

The "Binary" dataset (Babacan, 2023) created synthetically according to earlier study (Babacan et al., 2023) contains 3527 texts classified as either having some of the cognitive distortions (labeled as "Distortion") or not (labeled as "No Distortion"). The dataset is imbalanced, 74% of texts have distortions expressed and only 26% have no distortions.

The "Multi-class" dataset (Shreevastava, 2021) contains 2530 annotated sample texts of the patient's input annotated manually (Shreevastava and Foltz, 2021). Among the samples, 933 one are annotated as "No Distortion", remaining ones are annotated for having one or more distortions ("Personalization": 202, "Labeling": 203, "Emotional Reasoning": 169, "Fortune-telling": 210, "Magnification": 245, "Mind Reading": 295, "All-or-nothing thinking": 126, "Overgeneralization": 277, "Mental filter": 151, "Should statements": 135).

In our current study we address binary classification only, so both original datasets were consolidated into a single dataset, which we refer to as the combined dataset, consisting of 6057 texts. Among them, 4191 texts (69%) were labeled as containing some distortion — either an unspecific "Distortion" label or one or two specific distortions, as identified above. The remaining 1866 texts (31%) were

labeled as "No Distortion".

The combined dataset was used in two ways. First, it was used for "overfitting" experiment when the same model was used for both learning and testing — in order to make sure that our goal is reachable at all, sort out which selection metrics for the N-grams are practical and see what could be the "upper line" for accuracy and $F1$ measures. Second, it was split into three separate sections for error study and "cross-validation" experiment.

For the error study and "cross-validation" purposes, the combined dataset was divided into three parts based on triples: every first element of each triple was placed in the first split, the second in the second split, and the third in the third split. Moreover, for error assessment of accuracy and $F1$, each model was evaluated against every split independently. For cross-validation purposes, every model was trained and tested against different test and train corpora in three rounds. The first round involved training on the first and second splits and testing on the third, the second round — training on the first and third and testing on the second, and the third round — respectively. For every round, individual measures of accuracy and $F1$ were collected for error analysis. That means, each of the three rounds was based on 4038 texts in training set and 2019 texts in test set.

4 Methodology

Our goal was to develop an interpretable model capable of detecting cognitive distortions in text with high reliability and optimal performance. To achieve this, we decided to use an interpretable text classification algorithm based on structural pattern (N-gram) matching, applying the "priority on order" principle (Kolonin, 2022; Raheman et al., 2022). This principle means that N-grams of higher order (larger N) take precedence over N-grams of lower order (smaller N) that they contain. For example, if the tetragram ["not", "a", "bad", "thing"] is identified, then the bigrams ["bad", "thing"] and the unigram ["bad"] are disregarded.

The model we obtained using our technology consists of a set of dictionaries containing N-grams for respective distortions. The content of these dictionaries was obtained during the learning stage, while the accuracy and $F1$ scores were evaluated during the detection stage.

4.1 Learning

At this stage, we conducted tokenization and formed N-grams, which were then stored in the corresponding dictionaries. The selection of N-grams for the dictionaries was based on the following hyper-parameters:

Punctuation on/off — this parameter indicates the presence or absence of punctuation and special characters in the N-grams.

N-gram max length (N_{max}) — the maximum length of the considered N-grams. For performance reasons, this hyper-parameter takes values from 1 to 4 inclusive.

N-gram inclusion threshold (%) (IT) — the threshold for including key N-grams in the model's dictionaries. It ranges from 0% to 90%.

N-gram selection metric (SM) — the metric computed to rank N-grams for inclusion in a model based on IT applied to the metric value. The following metrics were involved, along with abbreviations used to refer to them in Figure 1.

G_g — Total count of N-gram g in the entire corpus.

D_d — Count of texts with distortion d per corpus.

G_d — Count of N-grams associated with distortion d.

G^u_g — Total unique count of N-gram g in the corpus (each N-gram counted once per text).

D_g — Count of distortions by N-gram g (from DG_{dg}).

GD_{gd} — Count of N-gram g associations with distortion d or N-gram frequency (F).

GD^u_{gd} — Unique count of N-gram g associations with distortion d per-text or unique frequency (UF).

DG_{dg} — Count of distortion d associations with N-gram g, $DG_{dg} = GD^u_{gd}$.

$TF\text{-}IDF = \frac{GD_{gd}}{G_d}$ — $TF\text{-}IDF$ normalized by total N-gram distortion associations.

$\bar{GD}_{gd} = \frac{GD_{gd}}{G_g}$ — Count of associations of N-grams g with distortion d, normalized by its count across the entire corpus, or "frequency normalized" (FN).

$\bar{GD}^u_{gd} = \frac{GD^u_{gd}}{G^u_g}$ — Count of unique associations of N-grams g with distortion d, normalized by its unique count across the entire corpus, or "unique frequency normalized" (UFN).

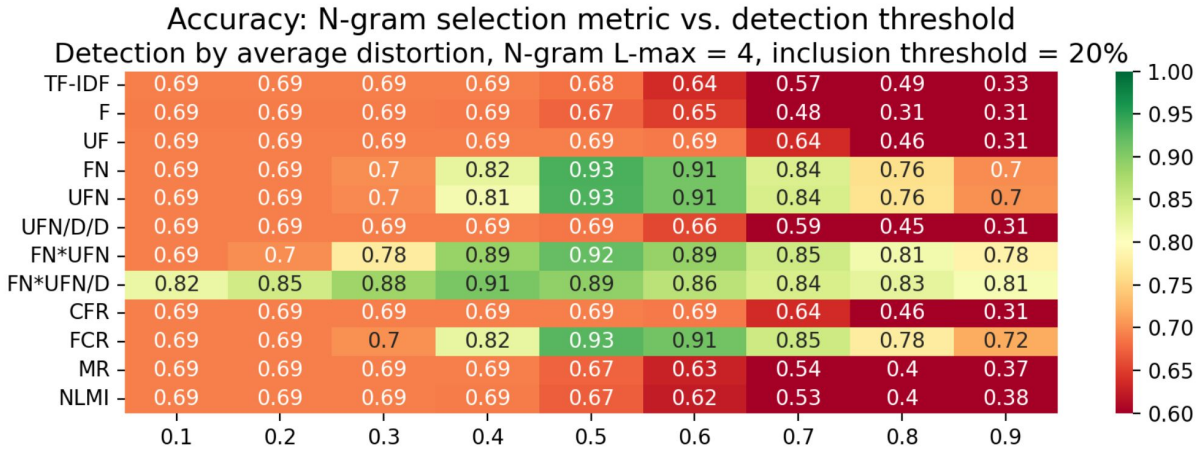

Figure 1: A heatmap illustrating the accuracy values for the selection metrics (SM) we considered, depending on the binary classification detection threshold (DT) ranging from 0.1 to 0.9. We examine the cross-validation experiment (multi-class view) with the detection function (DF) set to average, under fixed values of N-gram max length (N_{max}) = 4 and N-gram inclusion threshold (IT) = 20%.

$\frac{GD^u_{gd}}{D_d \cdot D_g}$ — "Unique frequency normalized" denominated by count of texts with given distortion in the corpus and distortions associated with N-gram ($UFN/D/D$).

$\bar{GD}_{gd} \cdot \bar{GD}^u_{gd}$ — Product of "frequency normalized" and "unique frequency normalized" ($FN * UFN$).

$\frac{\bar{GD}_{gd} \cdot \bar{GD}^u_{gd}}{D_g}$ — Previous metric additionally normalized by count of distortions by N-gram ($FN * UFN/D$).

$\frac{GD^u_{gd}}{\sum_d GD^u_{gd}}$ — Feature Category Relevance (FCR), according to Kolonin (2022).

$\frac{GD^u_{gd}}{\sum_g GD^u_{gd}}$ — Category Feature Relevance (CFR), according to Kolonin (2022).

$\frac{(GD^u_{gd})^2}{(\sum_d GD^u_{gd})(\sum_g GD^u_{gd})}$ — Mutual Relevance (MR), according to Kolonin (2022).

$\frac{(GD^u_{gd})^2}{D_d \cdot G^u_g}$ — Non-logarithmic Mutual Information ($NLMI$).

As described in section 3, we combined two datasets (binary and multi-class) into a single combined dataset, for which we conducted both an overfitting experiment and a cross-validation experiment. For each of the experiments, we solved the binary task (distortion/no distortion) in two ways. The first way, labeled as "binary view", corresponds to training a binary model. This means that at the learning stage, the model learns to recognize whether there is a distortion in the text or not. The second way, labeled as "multi-class view", corresponds to training a multi-class model. At the learning stage, the model learns to recognize 11 distortions (10 specific ones and one general "distortion") and "No Distortion", while binary classification is performed at the detection stage.

4.2 Detection

At this stage, we perform the detection of cognitive distortions based on the built model, which consists of dictionaries of associated N-grams, dictionary per distortion. Detection is performed considering the following hyper-parameters, applied to scores $\bar{C}$ computed according to Algorithm 1.

Logarithmic/non-logarithmic scaling (LS) – used or not used for scaling the numerical results of $\bar{C}$ before applying the threshold further.

Distortion detection threshold (DT) – sets the threshold for binary classification based on $\bar{C}$ value. This parameter is necessary because our model can determine the "intensity" of cognitive distortions $\bar{C}$ with a continuous value from 0 to 1. Therefore, to obtain a binary result, we set a threshold below which values are considered 0, and above which values are considered 1.

Detection function (DF) – used in the case of multi-class view and allows converting the results into a binary form in two ways: average – based on the average value across all cognitive distortions, any – based on at least one cognitive distortion.

The recognition function, which outputs the intensity of cognitive distortion in the text, is based

on the frequency of N-grams found in the text and the dictionaries of the trained model. Algorithm 1 shows how the frequency of N-grams is taken into account and the "priority on order" principle.

Algorithm 1 Priority on order in detection algorithm

Require: Input text T, cognitive distortions dictionaries $\mathcal{D}_1, \ldots, \mathcal{D}_k$ with N-grams up to $N_{\max}$
Ensure: Normalized metric scores $\bar{C}_1, \ldots, \bar{C}_k$

Tokenize T into sequence $S = [s_1, \ldots, s_l]$; let i denote the token position in S, and w the current N-gram starting at position i
Create mask $M = [1, \ldots, 1]$ of length l
Initialize counts $C_j = 0$ for each metric $j = 1, \ldots, k$
for $n = N_{\max}$ **to** 1 **do**
  for $i = 0$ **to** $l - n$ **do**
    if $\sum_{t=0}^{n-1} M[i+t] = n$ **then**
      $w \leftarrow (s_i, \ldots, s_{i+n-1})$
      $found \leftarrow$ false
      for $j = 1$ **to** k **do**
        if $w \in \mathcal{D}_j$ **then**
          $C_j \leftarrow C_j + n$
          $found \leftarrow$ true
        end if
      end for
      if $found$ **then**
        $M[i : i+n] \leftarrow 0$
      end if
    end if
  end for
end for
for $j = 1$ **to** k **do**
  if log-scaling (LS) enabled **then**
    $\bar{C}_j \leftarrow \frac{1}{2} \log_{10}(1 + 100 \cdot C_j / l)$
  else
    $\bar{C}_j \leftarrow C_j / l$
  end if
end for

4.3 Experimental Setup

As described earlier, we conducted four experiments: overfitting experiments (binary view and multi-class view) and cross-validation experiments (binary view and multi-class view). For each experiment, the learning stage generated the model dictionaries, and the detection stage performed cognitive distortion detection and calculated accuracy and $F1$ score. We performed a full grid search over all possible hyperparameter combinations to determine optimal values for each experiment.

Based on the learning studies discussed above for two types of experiments ("overfitting" and "cross-validation") and two types of models ("binary view" and "multi-class view"), we selected the respective best-performing models for each experiment, relying on accuracy and $F1$ measures.

For the two cross-validation experiments (binary view and multi-class view), we created two "joint" models based on the best-performing models. These "joint" models included only the N-grams that were selected above the DT threshold across all three splits, separately for binary view and multi-class view.

4.3.1 Model Benchmarking

Based on the learning studies discussed above for two kinds of experiments ("overfitting" and "cross-validation") and two kinds of models ("binary view" and "multi-class view") we selected respective best winning models for each of the experiments, relying on accuracy and $F1$ measures.

These models were compared with different detection hyper-parameters against suite of baseline and alternative models. The baseline models were used to provide reference "bottom line" accuracy and $F1$ measures for the imbalanced dataset used for testing, including $Const(True)$ providing always positive assessments, $Const(False)$ — always negative ones, $Random$ — random true of false assessments. We also explored our own baseline model created relying on N-grams presented by Bollen et al. (2021) and later used by Arinicheva and Kolonin (2025).

In addition to that, we compared our models against large language models (LLM), namely LLAMA 3.2 (3B) (Grattafiori et al., 2024), QWEN 2 (7B) (Yang et al., 2024), QWEN 2.5 (7B), and QWEN 2.5 (14B) (Qwen et al., 2025) deployed locally. Detection of a cognitive distortion presence in a $text$ using LLM was performed by query "Be concise. Does this text have cognitive distortions in it "$text$"?" and analysis if the response starts with case-insensitive "yes".

The benchmarking was done in two rounds. First, the full data set was used to find semi-optimal detection hyper-parameters (LS, DF, and DT) for our own models — baseline and learned. Second, the models with the best hyper-parameters selected during the first round were explored on the three separate splits of the entire dataset discussed in section 3 against all "bottom line" baseline models

and alternative LLM ones.

5 Results and Discussion

5.1 Learning Results

Based on all experiments for model learning — both "overfitting" and "cross-validation" applied to the "binary view" and "multi-class view" models, the following results can be stated.

Including or excluding punctuation had no significant impact on accuracy and $F1$ measures. However, visual analysis of N-grams revealed that those containing punctuation lacked interpretability or meaningful contribution. Therefore, for model benchmarking and practical deployment, we only considered punctuation-less models and recommend them for future use due to their higher clarity and interpretability.

Exploration of selection metric has revealed category of metrics practical for N-gram selection, such as $FN, UFN, FN*UFN, FN*UFN/D$, and FCR, providing the best accuracy and $F1$ with nearly the same absolute values at the same detection threshold values, as shown in Figure 1. The FN metric was selected for the final model because it achieved the highest accuracy and $F1$ while remaining the least computationally expensive (one division of two sums per N-gram).

The study of N-gram length (N_{max}) was conducted for values up to and including 4. In most overfitting and cross-validation experiments, optimal accuracy and $F1$ occurred at $N_{max} = 3$, with only marginal gains at $N_{max} = 4$ (see Figure 2). The sole exception was the "binary view" cross-validation experiment, which also peaked at $N_{max} = 3$.

Analysis of the inclusion threshold (IT) during the learning phase revealed the expected inverse relationship with the detection threshold (DT) used in the detection stage. The higher the IT, the fewer indicative N-grams were included in the models, which required a lower DT during detection, as shown in Figure 3 for the DT range of 0.2–0.6. For the "overfitting" experiments, optimal IT lay between 80% and 90% for the binary view and 50% for the multi-class view. For "cross-validation" experiment, optimal IT ranged from 50% to 70% for the binary view and from 20% to 30% for the multi-class view. That is, "multi-class view" generally required a lower inclusion threshold.

For all experiments, the model configurations of hyper-parameters in Table 1 were considered the best for further use.

5.2 Models Benchmarking

Evaluation of all models selected above with all sets of detection hyper-parameters led to the following observations.

Use of logarithmic scaling LS is indicated as $LS = log$ if turned on and $LS = no\ log$ if turned off. As expected, it had no major impact on the best accuracy and $F1$ measures reached, it just affects the required optimal level of detection threshold (DT). At the same time, we found that "binary view" models are less sensitive to DT when $LS = log$, whereas "multi-class view" models — with $LS = no\ log$.

The use of different detection functions (DF) applied to "multi-class view" models showed, in most cases, similar best values of accuracy and $F1$. An exception was observed with the "conservative" ("joint") model, where the best results were obtained using the $RF = avg$ ("average") function.

Table 2 present the best selected models, including baseline, our models, and LLM-s with the best combinations of hyper-parameters selected based on the above considerations with mean percentage error (MPE) of measurement determined on basis of three cross-validation splits. Our models are coded with prefix "Ours" with 4-letter abbreviations described below, with recognition threshold RT in parentheses.

First letter — B or N — indicates whether the model was a baseline built using the N-gram dictionaries from (Bollen et al., 2021) and the algorithmic framework of (Raheman et al., 2022) (B), or a new model learned during this study (N).

Second letter — B or M — indicates the "binary view" (B) or "multi-class view" (M).

Third letter — L or N — indicates logarithmic scaling, either $LS = log$ (L) or $LS = no\ log$ (N).

Fourth letter — N or V — indicates the detection function, either $RF = any$ (N), or $RF = avg$ (V) which can apply to "multi-class view" models only.

The most representative results in Table 2 make it possible to consider few classes of models with respective accuracy and $F1$ levels.

The bottom line baseline used the $Const(True)$ model (always "positive") which provides target measures of accuracy at 0.692 and $F1$ at 0.818 due to imbalance of the data set, so this can be considered as minimally acceptable level for the model.

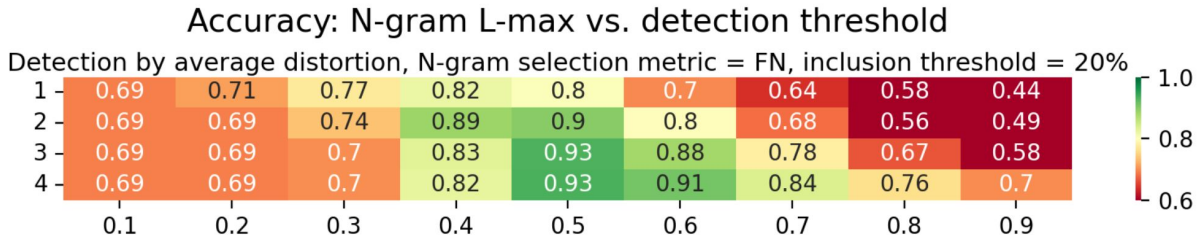

Figure 2: A heatmap illustrating the accuracy values for the N-gram max length (N_{max}) we considered, depending on the binary classification detection threshold (DT) ranging from 0.1 to 0.9. We examine the cross-validation experiment (multi-class view) with the detection function (DF) set to average, under fixed values of selection metrics (SM) = FN and N-gram inclusion threshold (IT) = 20%.

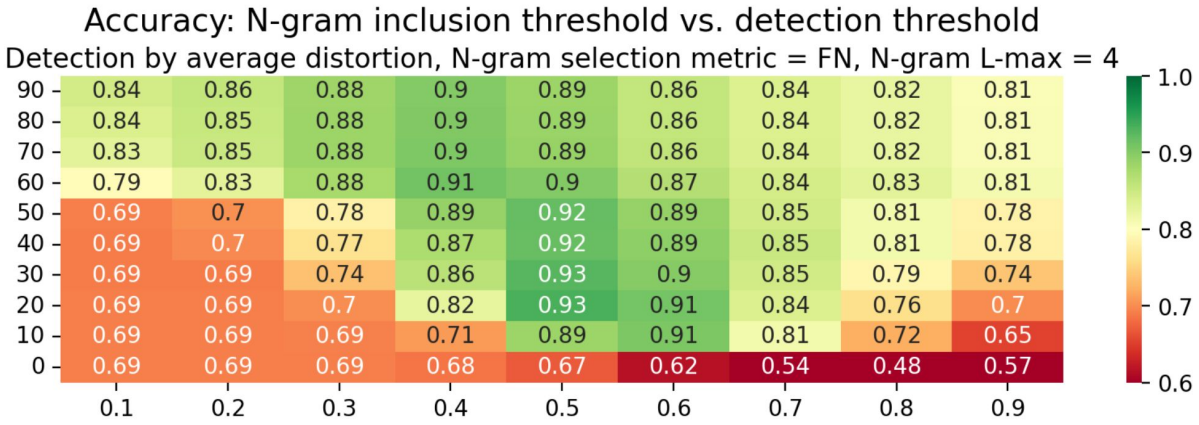

Figure 3: A heatmap illustrating the accuracy values for the N-gram inclusion threshold (IT) we considered, depending on the binary classification detection threshold (DT) ranging from 0.1 to 0.9.We examine the cross-validation experiment (multi-class view) with the detection function (DF) set to average, under fixed values of selection metrics (SM) = FN and N-gram max length (N_{max}) = 4.

Both LLM models that we explored (LLAMA and QWEN) showed target measures about $1-4\%$ higher than the bottom baseline, comparable with the average MPE level 1.5%.

Our baseline model $OursBMLV(0.4)$ provided target measures $3-6\%$ higher than LLM, with difference exceeding the average MPE.

Our models learned from the "overfitting" experiment ("binary" $OursNBLN(0.6)$ and "multi-class" $OursNMNN(0.6)$) provided upper limit with accuracy and $F1$ at 1.0 but we are cautious recommending them for practical use because of their "over-fitted" nature, still providing the models for reference.

Ours "joint" ("binary" $OursNBNN^{*}(0.1)$ and "multi-class" $OursNMNN^{*}(0.5)$) models obtained based on "cross-validation" models ($OursNBLN^{**}(0.7)$ and $OursNMNV^{**}(0.2)$) deliver consistently high accuracy at $0.91-0.96$ and $F1$ at $0.93-0.97$ which makes it possible for us to recommend the "joint" models for practical purposes, still having the models reviewed and edited, if needed, by a human expert.

Comparing the run-time performance across the models, we also found that average time required to process single text by LLM models was taking from 1 to 7 seconds given our computing resources in possession, as shown in Table 3. At the same time, any of other models, including ours, was taking less than 1 millisecond.

6 Conclusion

We explored the interpretable text classification algorithm based on structural pattern (N-gram) matching with the "priority on order" principle, applied to the task of detection of cognitive distortions in natural language texts. This approach achieved a practically reasonable level of accuracy, exceeding the level reached by interpretable models and comparable to that of neural network-based solutions. Moreover, the interpretable nature of the algorithm makes it possible to report or highlight specific text fragments or figures of speech in the

Model	View	SM	N_{max}	IT,%
Overfitting	Binary	FN	3	90
Overfitting	Multi-class	FN	4	50
Cross-validation	Binary	FN	2	60
Cross-validation	Multi-class	FN	4	20

Table 1: For each experiment (model-view), the best learning hyper-parameter values are provided, at which the maximum accuracy and F1 score were achieved.

Model	Accuracy	MPE(Accuracy),%	F1	MPE(F1),%
Const(True)	0.69	1.9	0.82	0.5
Const(False)	0.31	4.3	0.00	0.0
Random	0.49	5.1	0.57	0.8
OursBMLV(0.4)	0.80	1.9	0.85	0.6
OursNBLN(0.6)	0.90	0.2	0.99	0.1
OursNMNN(0.6)	1.00	0.1	1.00	0.0
OursNBNN*(0.1)	0.96	1.9	0.97	0.5
OursNMNN*(0.5)	0.92	1.5	0.94	0.5
OursNBLN**(0.7)	0.92	1.2	0.94	0.3
OursNMNV**(0.2)	0.94	1.3	0.95	0.4
llama3.2:3b	0.71	1.2	0.83	0.3
qwen2:7b	0.74	2.4	0.81	0.6
qwen2.5:7b	0.73	1.6	0.82	0.4
qwen2.5:14b	0.73	1.6	0.82	0.4

Table 2: For each of our models with a specific set of hyper-parameters and for each LLM model applied to our task, we present the best accuracy and F1 score values, as well as the corresponding mean percentage errors (MPE).

Model	seconds/text
Const(True)	0.0001
Const(False)	0.0001
Random	0.0001
OursBM (baseline, multi-class)	0.0006
OursNB (new, binary)	0.0002
OursNM (new, multi-class)	0.0005
LLM:llama3.2:3B	1.08
LLM:qwen2:7B	2.03
LLM:qwen2.5:7B	1.51
LLM:qwen2:14B	7.10

Table 3: Runtime performance for different models.

text being explored for validation by an expert.

We developed and tested a new learning algorithm capable of creating dictionaries of structured text patterns (N-grams). These dictionaries were used in the algorithm based on the "priority on order" principle, which we also formalized in this study. This made it possible to build two types of models capable of solving the binary classification problem in two ways — plain "binary" classification and "multi-class view" one. The latter solves "multi-class" problem first and then makes "binary" decision on the basis of the former.

We created and tested two interpretable models to detect the presence of cognitive distortions in natural language text in English, as described above. Both models achieved accuracy and $F1$ values exceeding 0.91, outperforming other interpretable models and comparable to neural network-based counterparts. The latter "multi-class" model can be also used for recognition of specific distortions individually, but evaluation of its performance "per se" is planned for future work.

We found that our solution, besides being interpretable, explainable and transparent, also delivers more than 1000 time greater run-time performance and lower computational cost than locally deployed LLM-based alternatives.

Future work will focus on extending our approach to other languages, primarily Chinese and Russian, and on training models capable of detecting emotional, social, and thematic nuances in text, with the goal of building complete interpretable tools for psychological diagnosis, treatment and monitoring.

7 Limitations

The primary goal of our work was to explore the possibility to develop technology for learning interpretable models for text classification, detection of cognitive distortions in the natural language text in particular. That means, even if we admit high accuracy of our models, we still leave the possibility of incorrect decisions based on them due to insufficiency or limitations in the English training corpora that we used in order to learn these models. We also admit that the accuracy obtained with given dataset may be lower with the other datasets. That means, to validate and improve the reliability and performance of our solution, further work may get required, including evaluation of our models and technology on larger corpora and datasets in languages other than English.

7.1 Data Limitations

The datasets referenced in section 3 are both limited by size and cover only the English language, and the first "binary" dataset is generated synthetically by Babacan et al. (2025). These datasets do not contain any information regarding demographics, gender or age. This may limit the practical applicability of the models inferred in this paper, so more reliable models and models for other languages can be built based on our technology based on additional and richer datasets, including datasets in different languages.

In order to ensure reliability of our work, we performed cross-validation using three different test/train splits based on the same corpus and compared the outcomes. We also created "conservative" model which was based on unification of all three partial data sets based on respective splits.

To ensure the robustness of our study, we evaluated all baseline and "conservative" models on three independent test splits, and our models learned independently on the three training splits were evaluated against the corresponding test splits, so we collected the accuracy values and $F1$ measures for the three corresponding runs for each model. These three runs were used to calculate the average values and the mean percent error (MPE) values shown in Table 2, so error bars can be drawn on the corresponding plots.

Both original datasets were unbalanced or biased in a sense that there were more texts labeled as having distortions in them compared to texts that were labeled as having no distortions. Furthermore, different distortion types were represented by varying numbers of texts in the dataset, as presented in section section 3. This would affect our target evaluations of accuracy and $F1$ due to fundamental nature of these metrics. To address this problem, we decided not to to balance them removing some texts or generating some extra synthetic texts. Instead, we preserved original datasets and performed model learning and model detection relying on them "as is". However, we computed "bottom line" evaluations for accuracy and $F1$ using functions such as $Const(True)$ (always "positive", assuming presence of distortion), $Const(False)$ (always "negative", assuming absence of distortion), and $Random$ (randomly "positive" or "negative") and used them for comparison with results provided by real prediction models being evaluated. That is, the competing model have not just exceed the other model in terms of higher accuracy and $F1$, but it mist have these evaluations substantially higher than provided by the "bottom line" evaluations, as shown in Table 2.

Given the combined dataset imbalance of 69% "positive" vs. 31% "negative" labels (see section 3), the "bottom line" evaluations according to Table 2 are 0.692 for accuracy and 0.818 for $F1$ with an error percentage of around 2% for most of the other models. We therefore expected that only those models whose accuracy and $F1$ values exceeded the "bottom line" measures by more than 2% could be considered practically usable.

At the beginning of this study, the only English datasets we identified were the two we found. We acknowledge that this may limit our work and plan to evaluate our solution on additional corpora in the future, including those in languages other than English. Since the focus of our study is the cognitive behavioral therapy, no other subject domains were involved in the study.

7.2 Methodology Limitations

Even though the technology that we develop can provide non-binary classification for multiple classes at once for any subject domain, we intentionally reduced the scope and objective of this study to practical application for detection of the fact that some cognitive distortions are present in given text. That is, only the binary classification problem is attacked in this study. The first reason of that is substantial imbalance of representation of different cognitive distortions discussed in section 3 and subsection 7.1. The second reason is

desire to solve one problem first and move to the next problem after that, so the next study that we plan will be dedicated to evaluation of true multi-classification capabilities of our solution.

We used cross-validation on three splits of the combined dataset, as described in section 3, in order to prevent overfitting and estimate levels of error for both learning and detection stages. The average values and errors for accuracy and $F1$ are computed based on different rounds of learning and detection on different dataset split arrangements. The error values are comparable to differences between the "bottom line" of $Const(True)$ ("always positive") and all LLM models, while the accuracy and $F1$ provided by the best configurations of our models appear substantially higher.

There may be more possible feature selection metrics beyond what we described in our study. Some of them were explored in the preliminary phase of our work and were not included in the article due to their low performance and intent to make the presentation compact and clean. Some others may have escaped our attention and may be included in future work.

While the possibility of either logarithmic or non-logarithmic scaling is a property of the detection algorithm presented in subsection 4.2, the search for hyper-parameters was done using only the logarithmic setting. This was done under the assumption and our experience that scaling mostly affects the detection threshold and not specific to the model being learned itself. However, based on the best models found in the course of the "overfitting" and "cross-validation" experiments for "binary view" and "multi-class view" models, we performed extra search for hyper-parameters, including the scaling. This search has shown that using non-logarithmic scaling with lower detection threshold can provide accuracy about 1% higher than it was possible with logarithmic scaling and higher detection threshold.

We did not explore N-gram lengths above 4 because the results showed only minor improvement when increasing them from 3 to 4, and because an earlier study (Kolonin, 2022) found that N greater than 3 was not practical.

We used accuracy and $F1$ measures to evaluate model performance in our study, however accuracy was selected as a primary measure because of being more "contrast" for the search of the best hyper-parameters purpose. That means, given the combined dataset imbalance, the "bottom line" of accuracy was 0.69 and of $F1$ was 0.82 (Table 2), so the former appeared 1.5 times more contrastive than the latter. However, the Pearson correlation value of 0.93 between both target measures appears high enough to justify our decision, which is confirmed by the nearly linear relationship between accuracy and $F1$ as shown in Figure 4.

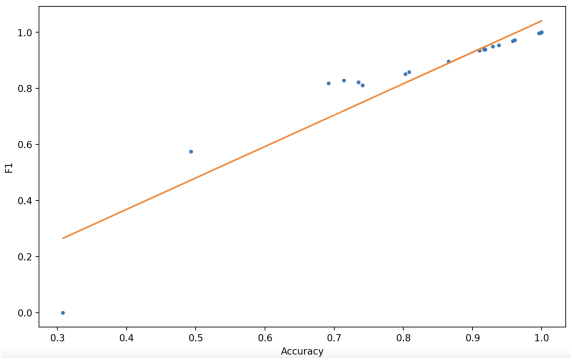

Figure 4: Relationship between accuracy and $F1$.

Given the limits applied on available computing and infrastructure resources, we limited power and scale of LLM models used for comparison with our solution, as discussed in the following section.

7.3 Infrastructure Limitations

The most powerful device that we had in possession was MSI Raider GE77HX 12UGS notebook with 12th Gen Intel(R) Core(TM) i7-12800HX 2.00 GHz, 32.0 GB RAM, 23.9 GB GPU NVIDIA GeForce RTX 3070 Ti Laptop GPU. It allowed us to run all performed experiments, including those involving LLM. Although the GPU had enough memory to run models with 3 and 7 billion parameters in GPU memory, the model with 14 billion parameters did not fit in GPU memory, so RAM was used, which slowed down the evaluation process several times, as shown in Figure 5 and Table 3.

Given that the goal of our work was to provide a solution that would work on sensitive data obtained by psychologists and psychotherapists, we were evaluating possibilities to run this solution locally on premises owned by professional specialists. So, even if theoretically we might obtain higher accuracy based on modern LLM models hosted in the cloud, we consciously limited the computer power down to what can be afforded by conventional practitioner.

In the end our study revealed that using our technology makes it possible to achieve substantially higher accuracy with computing costs more than 1000 times lower compared to LLM. For instance,

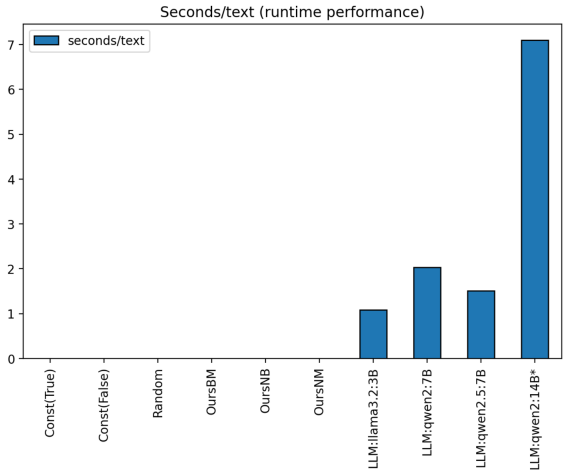

Figure 5: Runtime performance for different models, average seconds/text, based on Table 3.

as shown in Figure 5, average detection (inference) time when using LLM with 3-7 billion parameters took 1-2 seconds per single text, using LLM with 14 billion parameters needed 7 seconds, while using our model this time was lower than 1 millisecond. That means that our model is much less sensitive to infrastructure limitations than evaluated LLM models, still providing better accuracy and being interpretable and explainable.

7.4 Third-party Code Limitations

The related work by Babacan et al. (2023, 2025) refers to high $F1$ score comparable to ours, however the referred publications do not provide reproducible code artifacts, so we were not able to test them on our dataset to compare with our results.

8 Ethical Considerations

8.1 Social Good Awareness

Following the Association of Computing Machinery (ACM) Code of Ethics and Professional Conduct (`https://www.acm.org/code-of-ethics`), our work is targeted toward contributing to society and to human well-being.

In particular, we address the problem of explainable, interpretable, transparent and trusted AI, applied to the domain of psychological help and treatment is the area of cognitive behavioral therapy. This is achieved by means of delivering a technology for learning interpretable models for the detection of cognitive distortions in natural language texts. Such models, learned programmatically at first, can be further inspected and adjusted by human experts to ensure that no misclassification can take place. This also addresses the ACM Code of Ethics objective for improving the overall transparency of the scientific process.

As the ACM Code of Ethics encourages being fair and taking action not to discriminate any category of people, we are achieving this fairness and inclusiveness from two perspectives, as follows.

First, we anticipate that our approach taken for the English language, relying on English testing and training corpora, can be adopted to build respective models for languages other than English, including low-resource languages, to enable development of CBT applications for different linguistic cultures.

Second, we show the computational efficiency of our approach which makes it possible to build CBT applications for massive use at low cost in any geographical region including those without access to expensive high-bandwidth network infrastructure and high-performance computing equipment.

The positive impact of our work can be broadened if the solution we suggested could be adopted in other applications involving classification of natural language texts for wide range of business domains.

Specifically, it can potentially be used to improve the social well-being and increase the online security by applying our models to detect manipulative communications in online media since one of the referenced studies reported causal connections between distorted (presumably manipulative) communications in social and online media and the behavior of financial markets (Kolonin et al., 2023).

Another possible application improving the well-being of society at scale may involve the study of the cognitive state of entire population or online community by means of monitoring available online communications to detect significant bursts of increase in the cognitive distortions detected in response to economic and political developments, like it was studied by Bollen et al. (2021), but performed in real time.

8.2 Potential Risks

The risks of employing any technology of cognitive distortions detection are similar to risks of sentiment analysis, explored in depth earlier (Karoo and Chitte, 2023; Denecke and Gabarron, 2024). The major risks can be primarily enumerated as risk of impact of mistake, risk of misuse or improper use, and risk of privacy violation as discussed below.

8.2.1 Impact of Mistake

If we were developing non-interpretable model used for detection of cognitive distortions, the major risk would be its misuse so the false positive detections would effect in incorrect diagnostics of the cognitive distortions. However, since our model is based on interpretable and human-readable patterns, any user of the model can inspect it and have its decisions explained, so the risk can be taken under control. Moreover, a professional user can even adjust the model manually having the risk eliminated.

If the solution we provide is used to learn new models on insufficient or biased training data sets, or relying on the models that we present in this study blindly, without proper inspection, tuning and adjustment, false positive cognitive distortion diagnosis on behalf of psychologist or psychotherapist could result in inappropriate treatment. At the same time, while conventional non-interpretable solutions make this problem impossible to address, we make it possible to review, adjust and fine-tune the interpretable models manually by expert.

8.2.2 Misuse

Any solution for psychological treatment or, specifically, for psychological diagnostics like ours may be misused. It can be misused by professional users like psychologists and psychotherapists not using it carefully with proper validation and control. Also, it can be misused by non-professional users trusting the results of the diagnostics too much or drawing non-professional and misleading conclusions from the results of such diagnostics.

While we anticipate that our solution can increase the performance and reliability of psychological diagnostics and even make self-diagnostics possible, the care should be taken by professional users, validating the diagnosis. Moreover, the non-professional users having access to our solution and decided to use it for self-diagnostics should refer to professionals to confirm the diagnosis and make sure about the need for any treatment.

Given that our solution makes it possible to learn models for cognitive distortions detection on any corpus, it can also be misused if the model is trained on insufficient or invalid training corpora and then used without any form of control, leading to incorrect diagnostics. However, the benefit of our solution, compared to others in this domain, is that the interpretable nature of the model makes it possible to inspect the quality of the model by professionals before using it for practical diagnostic purposes, which eliminates the risk with proper care.

8.2.3 Privacy Violation

The use of our technology by psychologists or psychotherapists based on informed consent obtained from the client appears to be a fair use case. Also, its use by governmental and business entities to monitor the public sources of online and social media in order to detect distorting trends appears legitimate. However, using it in respect to proprietary textual data obtained violating human privacy, adds extra value to the collected data for violators which can increase the harm to the data privacy subjects. However, this seems to be no different with any other sort of processing of the human private data collected in an inappropriate way and it has to be prevented by conventional security and legal means.

8.3 Scientific Artifacts

8.3.1 Datasets

The "binary" dataset, created synthetically, is publicly available (Babacan, 2023). The license is not specified in the dataset files or online metadata, however we contacted the authors (Babacan et al., 2025) and they confirmed that it is released under the MIT license, so our use of it can be considered fair.

"Multi-class" dataset contains the 2530 annotated samples of the patient's input annotated manually (Shreevastava and Foltz, 2021) and available online (Shreevastava, 2021). The license is not indicated in the dataset files or online metadata, however the data set is available online for four years and it was referenced in multiple latest publications (Shreevastava and Foltz, 2021; Shreevastava, 2021; Babacan et al., 2023; Babacan, 2023; Babacan et al., 2025) so we treat possibility of its use as fair.

Both datasets are published on the machine learning sites and have metadata and supplementary information indicating their purpose intended for machine learning purposes, so our use of them may be considered as intended.

The manual study of referenced datasets revealed that neither identification of individual people is possible nor offensive content is found, so no ethical issues may be anticipated. Since the original datasets were anonymized in the earlier studies that provided them (Shreevastava and Foltz, 2021;

Shreevastava, 2021), no extra anonymization effort was considered as necessary for us.

Even though we temporarily create unified datasets for training and testing on the basis of the two datasets referenced above in run-time, we do not build or release an artifact out of it, so no extra licensing on top of the existing regulations is required for those who decide to reproduce our work using the same datasets.

8.3.2 Models

The model files for cognitive distortions detection, along with the code we developed during the study and its release, are licensed under the MIT License, with no limitations on intended use, except for unlawful activities.

The model files used as our baseline models are manually created based on data published in public work by Bollen et al. (2021) in a format aligned with the design referenced in work by Raheman et al. (2022), which references the model data under MIT license. Moreover, the same model files were used in subsequent study by (Arinicheva and Kolonin, 2025) earlier.

The model files do not contain any offensive content or information that can be used for identification of individual people, because they are derived from the training datasets that have no such content or information either.

8.4 Human Annotators and Participants

No human annotators, crowd-workers or any other human participants, except the authors, were involved in our research, because all test and train data that we were using were available as described in section 3.

8.5 Use of AI assistants

No use of any AI assistants (like ChatGPT or Copilot) was involved in our research, involving coding and manuscript writing.

Acknowledgments

This work was supported by a grant for research centers, provided by the Anonymized.

References

Mohammed Al-Mosaiwi and Tom Johnstone. 2018. In an absolute state: Elevated use of absolutist words is a marker specific to anxiety, depression, and suicidal ideation. *Clinical Psychological Science*, 6:529–542.

Anna Arinicheva and Anton Kolonin. 2025. Diagnosis of cognitive distortions in public, group, and personal text communications. In *Advances in Neural Computation, Machine Learning, and Cognitive Research VIII*, volume 1179, pages 337–344. Springer Nature Switzerland.

Babacan. 2023. autotrain-data-cognitive_distortions (revision 4bc1d87). Online dataset at Hugging Face, visited 25-March-2025.

Babacan. 2024. combined_synthetic_cognitive_distortions (revision 9995a75). Online dataset at Hugging Face, visited 25-March-2025.

Hakkı Halil Babacan, Yahya Beyitoğlu, and Ramazan Oğuz. 2023. Creating a clinical psychology dataset with synthetic data: Automatic detection of cognitive distortions classified with nlp. Available at SSRN: urlhttps://ssrn.com/abstract=4582307 or urlhttp://dx.doi.org/10.2139/ssrn.4582307.

Hakkı Halil Babacan, Ramazan Oğuz, and Yahya Kemal Beyitoğlu. 2025. Creating a clinical psychology dataset with synthetic data: Automatic detection of cognitive distortions classified with nlp. *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, 37(1):83–92.

Krishna C. Bathina, Marijn ten Thij, Lorenzo Lorenzo-Luaces, Lauren A. Rutter, and Johan Bollen. 2021. Depressed individuals express more distorted thinking on social media. *Nature Human Behaviour*, page 458–466.

Aaron T. Beck. 1963. Thinking and depression. I. idiosyncratic content and cognitive distortions. *Arch Gen Psychiatry*, 9(4):324–333.

Aaron T. Beck. 1976. *Cognitive Therapy and the Emotional Disorders*. International Universities Press, Inc., Madison, CT.

Johan Bollen, Marijn ten Thij, Fritz Breithaupt, Alexander T. J. Barron, Lauren A. Rutter, Lorenzo Lorenzo-Luaces, and Marten Scheffer. 2021. Historical language records reveal a surge of cognitive distortions in recent decades. *PNAS*, 118(30):e2102061118.

Alice Boyes. 2013. 50 common cognitive distortions. Blog post, *Psychology Today*, "In Practice" series. Accessed: 2025-05-02.

Rafael Calvo, David Milne, Sazzad Hussain, and Helen Christensen. 2017. Natural language processing in mental health applications using non-clinical texts. *Natural Language Engineering*, 23(5):649–685.

Kerstin Denecke and Elia Gabarron. 2024. The ethical aspects of integrating sentiment and emotion analysis in chatbots for depression intervention. Online at https://doi.org/10.3389/fpsyt.2024.1462083, visited 25-March-2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Krishna Karoo and Vikas Chitte. 2023. Ethical considerations in sentiment analysis: Navigating the complex landscape. *International Research Journal of Modernization in Engineering Technology and Science*, 05.

Anton Kolonin. 2022. High-performance automatic categorization and attribution of inventory catalogs. *arXiv:2202.08965*.

Anton Kolonin, Ali Raheman, Mukul Vishwas, Ikram Ansari, Juan Pinzon, and Alice Ho. 2023. Causal analysis of generic time series data applied for market prediction. In *International Conference on Artificial General Intelligence, Lecture Notes in Computer Science (LNAI, volume 13539)*, pages 30–39.

Nawal Ouhmad, Romain Deperrois, Wissam El Hage, and Nicolas Combalbert. 2023. Cognitive distortions, anxiety, and depression in individuals suffering from ptsd. *International Journal of Mental Health*, 53(4):336–352.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Ali Raheman, Anton Kolonin, Igors Fridkins, Ikram Ansari, and Mukul Vishwas. 2022. Social media sentiment analysis for cryptocurrency market prediction. *arXiv:2204.10185*.

Benjamin Shickel, Scott Siegel, Martin Heesacker, Sherry Benton, and Parisa Rashidi. 2019. Automatic detection and classification of cognitive distortions in mental health text. *2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE)*, pages 275–280.

Sagarika Shreevastava. 2021. Cognitive distortion detection dataset (version 1). Online dataset at Kaggle, visited 25-March-2025.

Sagarika Shreevastava and Peter Foltz. 2021. Detecting cognitive distortions from patient-therapist interactions. In *Proceedings of the Seventh Workshop on Computational Linguistics and Clinical Psychology: Improving Access*, pages 151–158, Online. Association for Computational Linguistics.

Taetem Simms, Clayton Ramstedt, Megan Rich, Michael Richards, Tony R. Martinez, and Christophe G. Giraud-Carrier. 2017. Detecting cognitive distortions through machine learning text analytics. In *2017 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 508–512, Los Alamitos, CA, USA. IEEE Computer Society.

Gopendra Singh, Soumitra Ghosh, Asif Ekbal, and Pushpak Bhattacharyya. 2023. Decode: Detection of cognitive distortion and emotion cause extraction in clinical conversations. In *Advances in Information Retrieval*, pages 156–171, Cham. Springer Nature Switzerland.

Gopendra Vikram Singh, Sai Vardhan Vemulapalli, Mauajama Firdaus, and Asif Ekbal. 2024. Deciphering cognitive distortions in patient-doctor mental health conversations: A multimodal LLM-based detection and reasoning framework. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22546–22570, Miami, Florida, USA. Association for Computational Linguistics.

Stanislav Sochynskyi. 2021. Automated cognitive distortion detection and classification of reddit posts using machine learning. Master's thesis, University of Tartu. Chair of Natural Language Processing, Supervisor: Kairit Sirts, PhD.

Justin S Tauscher, Kevin Lybarger, Xiruo Ding, Ayesha Chander, William J Hudenko, Trevor Cohen, and Dror Ben-Zeev. 2023. Automated detection of cognitive distortions in text exchanges between clinicians and people with serious mental illness. *Psychiatric Services*, 74(4):407–410. Epub 2022 Sep 27.

Bichen Wang, Yanyan Zhao, Xin Lu, and Bing Qin. 2023. Cognitive distortion based explainable depression detection and analysis technologies for the adolescent internet users on social media. *Frontiers in Public Health*, Volume 10 - 2022.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 43 others. 2024. Qwen2 technical report. *Preprint*, arXiv:2407.10671.

Xuejiao Zhao, Chunyan Miao, and Zhenchang Xing. 2017. Identifying cognitive distortion by convolutional neural network based text classification. *International Journal of Information Technology*, 23(1):1–12. © 2017 Singapore Computer Society. Author version, accepted for publication.

A Computational Experiment

A.1 Results of Model Benchmarking

In subsection 5.2, we describe the evaluation of all models using all detection hyper-parameters. Figure 6 and Figure 7 present a visualization of this evaluation.

In fact, in Figure 6 and Figure 7 it is seen that the same models are providing the best top accuracy and $F1$ measures with different detection thresholds. For instance, the model "Ours new (multi-class)" at the bottom of the referenced figures with detection by "any" distortion provides the highest accuracy of 0.92 and the highest $F1$ of 0.94 at different detection thresholds — 0.9 for logarithmic ("log") and 0.5-0.6 for non-logarithmic ("no log").

Figure 8 and Figure 9 present the accuracy and $F1$ measures for the best selected models, including the baseline models, our models, and LLM-based models, with the optimal combinations of hyper-parameters selected based on Table 1 and considerations from subsection 5.2. Error bars in these plots are visualized around the mean values calculated together with the mean percent error (MPE) shown in Table 2 based on three independent runs on the corresponding splits.

A.2 Computational Environment and Cost

The total computational budget for the entire study was approximately two months for each of the two computer notebooks: 1) MSI Raider GE77HX 12UGS notebook with 12th Gen Intel(R) Core(TM) i7-12800HX 2.00 GHz, 32.0 GB RAM, 23.9 GB GPU NVIDIA GeForce RTX 3070 Ti Laptop GPU; 2) MacBook Pro with 2.9 GHz 6-Core Intel Core i9, Radeon Pro 560X 4GB Intel UHD Graphics 630 1536 MB, 32 GB 2400 MHz DDR4. The final run time, using the former device, for each of the eight Python Jupyter notebooks of the learning experiment, including the full hyperparameter search space, was between 12 and 73 hours, depending on the type of experiment, with an average of about 48 hours. The run time for LLM evaluation Jupyter notebook, using the same device, was 20 hours. The run time for the final detection experiment and model comparison across all interpretable models on the same device was 16 minutes.

A.3 Model Parameters and Size

The baseline model created based on earlier work (Bollen et al., 2021; Raheman et al., 2022; Arinicheva and Kolonin, 2025) consisted of 14 thousand N-grams with N in range 1-4, representing 12 cognitive distortions ("catastrophizing", "emotional-reasoning", "dichotomous-reasoning", "fortune-telling", "overgeneralizing", "disqualifying-positive", "labeling", "personalizing", "magnification", "mental-filtering", "mindreading", "should-statement"), emotional and ("positive", "negative"), and rude speech.

The models created in the course of our study contain N-grams with N in range 1-4, representing 10 cognitive distortions ("All-or-nothing_thinking", "Emotional_Reasoning", "Fortune-telling", "Labeling", "Magnification", "Mental_filter", "Mind_Reading", "Overgeneralization", "Personalization", "Should_statements"), according to Shreevastava and Foltz (2021); Shreevastava (2021) and unclassified distortions according to Babacan et al. (2023); Babacan (2023); Babacan et al. (2025). The total number of N-grams in each model is shown in Table 4.

Model	N-grams
Overfitting, "binary view"	74
Cross-split, "binary view"	22
Cross-joint, "binary view"	15
Overfitting, "multi-class view"	341
Cross-split, "multi-class view"	323
Cross-joint, "multi-class view"	88

Table 4: Numbers of N-grams per model in thousands. Cross-split model numbers are given as average individual split across three different models obtained on respective splits. Cross-joint model means the "joint" model created as intersection of all N-grams per individual splits.)

A.4 Data Files

The following data files were used in the course of this study or were generated based on its result.

1. *./data/corpora/English/distortions/halilbabacan* — "Binary" dataset, according to Babacan (2023); Babacan et al. (2023, 2025)

2. *./data/models/distortions/ours* — baseline interpretable model created based on earlier work (Bollen et al., 2021; Raheman et al., 2022; Arinicheva and Kolonin, 2025)

3. *./data/models/distortions/overfitting_combined* — interpretable models created in the course of our study during the "overfitting" experiments

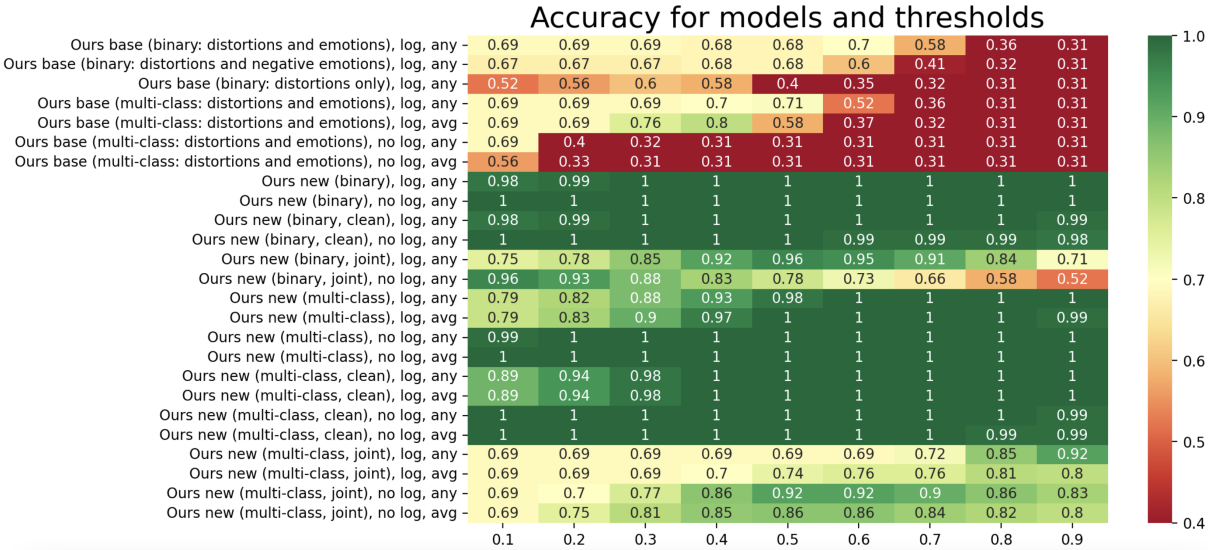

Figure 6: A heatmap illustrating the accuracy values for all our experiments with models obtained using all possible combinations of detection hyper-parameters.

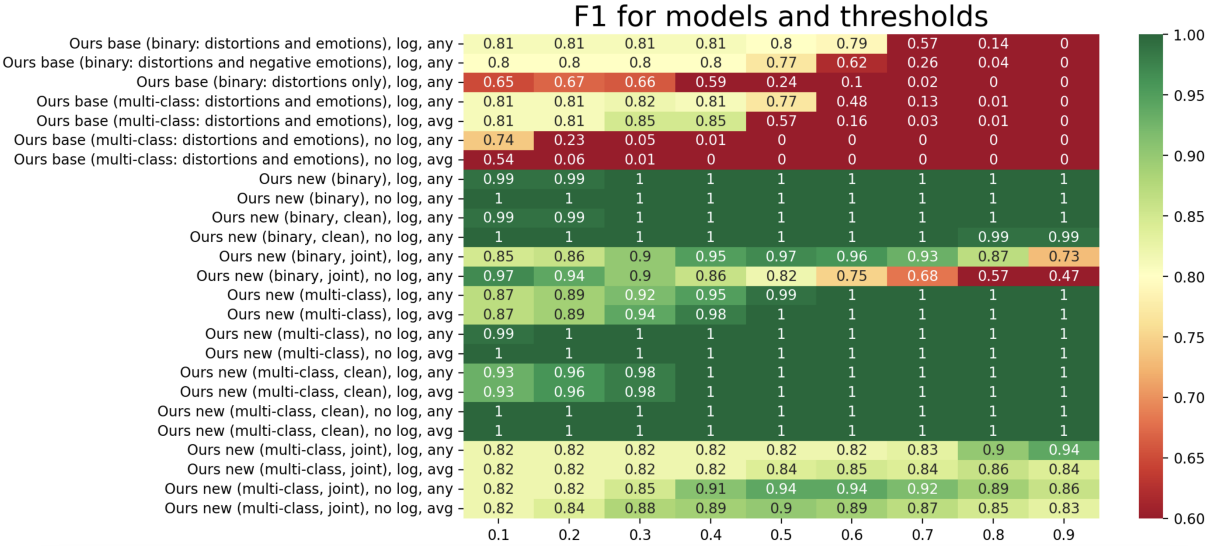

Figure 7: A heatmap illustrating the F1 score values for all our experiments with models obtained using all possible combinations of detection hyper-parameters.

4. *./data/models/distortions/split_combined* — interpretable models created in the course of our study during the "cross-validation" experiments

A.5 Code

The following code is supplied in the Anonymized repository and can be used to reproduce the results of our study and to extend the experiments. Python 3.11.11 was used for all experiments with external dependencies identified in the *requirements.txt* file with their respective versions. The following list details the code residing in the *./papers/distortions_binary_2025/* folder, in the order of proceedings needed to use the code to reproduce our results.

1. *requirements.txt* — list of dependencies to be installed under Python 3.11.11 environment, such as using *venv* and *pip*
2. *a_api.py, learn.py, plot.py, text.py, util.py* — program modules used by the following notebooks
3. Jupyter notebooks for learning experiments
   (a) *overfitting_combined*.ipynb* — overfitting experiment with no punctuation removed (initial study)

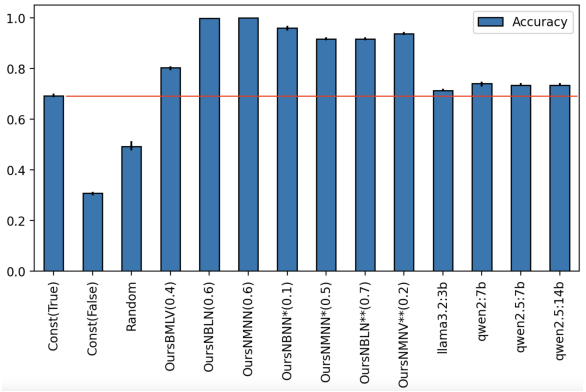

Figure 8: Bar plots with error bars illustrating the best accuracy values for our models with different sets of hyper-parameters compared to LLM-based models, based on Table 2, with red "bottom line" drawn at the level $Const(True)$ model.

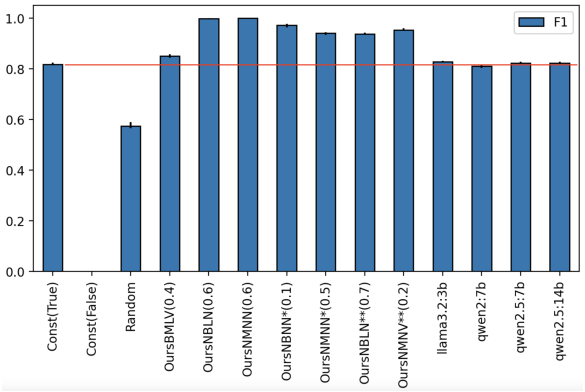

Figure 9: Bar plots with error bars illustrating the best $F1$ score values for our models with different sets of hyper-parameters compared to LLM-based models, based on Table 2, with red "bottom line" drawn at the level $Const(True)$ model.

(b) *overfitting_combined*cleaned.ipynb* — overfitting experiment with punctuation removed (cleaner and final results)

(c) *split_combined*.ipynb* — split cross-validation experiment with no punctuation removed (initial study)

(d) *split_combined*cleaned.ipynb* — split cross-validation experiment with punctuation removed (cleaner and final results)

4. *comparing_llms.ipynb* — Jupyter notebook for detection experiment using LLMs, saving the intermediate results to file *llm_evaluation_results* using *pickle* format and module

5. *comparing_models.ipynb* — Jupyter notebook for detection experiment comparing ours models against baseline and LLMs

B Related Work Overview

Table 5 below contains a detailed summary of the related work discussed in section 2.

Study	Dataset	Model	Accuracy	F1	Description
Detecting Cognitive Distortions Through Machine Learning Text Analytics (Simms et al., 2017)	Not present	Logistic Regression (LIWC + RELIEF features)	0.73	—	Binary classification; interpretable model
Automatic Detection and Classification of Cognitive Distortions in Mental Health Text (Shickel et al., 2019)	Not present	Logistic Regression (TF-IDF features)	0.90	0.88 (binary), 0.68–0.45 (multi-class)	Binary and multi-class classification; interpretable, high-performance model
Identifying Cognitive Distortion by Convolutional Neural Network Based Text Classification (Zhao et al., 2017)	Not present	Word2Vec (CBOW) + CNN	—	—	Multi-class classification; non-interpretable model
Automated cognitive distortion detection and classification of Reddit posts using machine learning (Sochynskyi, 2021)	Not present	FastText (binary), SVM + TF-IDF (multi-class)	—	0.71 (binary), 0.23 (multi-class)	Binary and multi-class classification
Automated Detection of Cognitive Distortions in Text Exchanges Between Clinicians and People With Serious Mental Illness (Tauscher et al., 2023)	Not present	Bidirectional encoder representations from transformers (BERT)	—	0.62	Binary and multi-class classification
Detecting Cognitive Distortions from Patient-Therapist Interactions (Shreevastava and Foltz, 2021)	Shreevastava (2021)	SVM + S-BERT embeddings	—	0.79 (binary), 0.3 (multi-class)	Binary and multi-class classification
Cognitive distortion based explainable depression detection and analysis technologies for the adolescent internet users on social media (Wang et al., 2023)	Not present	Bidirectional encoder representations from transformers (BERT)	—	0.78	Multi-class classification

DeCoDE: Detection of Cognitive Distortion and Emotion cause extraction in clinical conversations (Singh et al., 2023)	Not present	Multimodal, multi-task deep learning model (text, audio, visual features)	0.76	0.74	Binary classification
Creating a Clinical Psychology Dataset with Synthetic Data: Automatic Detection of Cognitive Distortions Classified with NLP (Babacan et al., 2023)	Babacan (2023)	RoBERTa	0.97	0.95	Binary classification
Diagnosis of Cognitive Distortions in Public, Group, and Personal Text Communications (Arinicheva and Kolonin, 2025)	Shreevastava (2021) and Babacan (2023)	Aigents	—	0.78 (binary), 0.25 (multi-class)	Binary and multi-class classification; interpretable, high-performance model
Deciphering Cognitive Distortions in Patient-Doctor Mental Health Conversations: A Multimodal LLM-Based Detection and Reasoning Framework (Singh et al., 2024)	Not present	LLM (LLAMA-7b)	0.84	0.80	Binary classification
Creating a Clinical Psychology Dataset with Synthetic Data: Automatic Detection of Cognitive Distortions Classified with NLP (Babacan et al., 2025)	(Babacan, 2024)	RoBERTa	0.95 (multi-class)	0.95 (multi-class)	Binary and multi-class classification
Ours	(Shreevastava, 2021) and (Babacan, 2023)	Ours	0.92	0.95	Binary classification; interpretable, high-performance model

Table 5: Overview of related work.