Suicide risk prediction in social media using Machine Learning and Large Language Models

Alessandro Costanzo Ciano

Master of Science
School of Informatics
University of Edinburgh
2025

Abstract

Suicide remains a global public health crisis, particularly in low- and middle-income countries with limited mental health resources. This study addresses a multi-class learning problem: the automated detection of suicide risk levels in social media posts. Using a dataset created from the r/SuicideWatch subreddit during the COVID-19 pandemic, 2000 posts are analysed, 500 of which are labelled across four risk levels—indicator, ideation, behavior, and attempt. Hybrid approaches are proposed, combining fine-tuned RoBERTa transformers with prompted large language models (LLMs) including Gemini, DeepSeek, and Grok families. A two-stage methodology is employed: initial supervised fine-tuning and prompting establish baselines, followed by semi-supervised pseudo-labelling of unlabelled data using high-confidence LLM predictions to augment RoBERTa training. Finally, an ensemble integrates the final version of RoBERTa-large with two optimized LLMs, achieving a weighted F1-score of 79.4% on cross-validation and 76.9% on a held-out test set, outperforming individual components and aligning with top benchmarks from the IEEE BigData 2024 Cup Challenge. The findings demonstrate the efficacy of LLM prompting and iterative pseudo-labelling, where high-confidence consensus predictions allowed a smaller model such as RoBERTa to achieve results comparable to state of the art general purpose LLMs.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Alessandro Costanzo Ciano*)

Acknowledgements

I would like to express my sincere gratitude to Professor Kobi Gal for the support and mentorship throughout my research. I also thank my friends and family for their support throughout this process. This work is dedicated to them.

Table of Contents

1 Introduction **1**

2 Background **4**

2.1 From Statistical Models to Deep Learning 4

2.2 Advanced Transformer and LLM Approaches 5

3 Dataset and Models Employed **7**

3.1 Dataset 7

3.2 RoBERTa 8

3.3 General-purpose Large Language Models 9

3.3.1 Gemini 10

3.3.2 DeepSeek 11

3.3.3 Grok 12

4 Methodology **13**

4.1 Stage 1: Fine Tuning and Prompt Design 13

4.1.1 Initial RoBERTa fine-tuning 14

4.1.2 LLMs foundational prompting 16

4.1.3 Iterative Self-Correction by LLMs 17

4.2 Stage 2: Pseudo-Labelling and Ensemble 19

4.2.1 LLM Pseudo-Labelling 19

4.2.2 Iterative Pseudo-Labelling with RoBERTa 20

4.2.3 Ensemble 21

4.3 Evaluation Criteria 22

5 Results **23**

5.1 Stage 1 23

5.1.1 RoBERTa Initial Fine-tuning Performance 23

5.1.2 LLM Foundational Prompting Performance 25
5.1.3 Iterative Prompt Refinement via Self-Correction 27
5.2 Stage 2 29
5.2.1 Identifying the Best Pseudo-Labelling Approach 29
5.2.2 RoBERTa Performance with Pseudo-Labels 30
5.2.3 Ensemble Model Performance 32
5.3 Final Evaluation on Held-Out Test Set 33

6 Discussion **34**
6.1 Interpretation of Results and key Findings 34
6.1.1 RoBERTa Initial Fine-Tuning Performance 34
6.1.2 Foundational Prompting with LLMs 34
6.1.3 Iterative Prompt Refinement via Self-Correction 35
6.1.4 Semi-Supervised Learning with Pseudo-Labelling 36
6.1.5 Ensemble Model 36
6.2 Comparison with Existing Literature 37

7 Conclusions **39**

Bibliography **41**

A Pseudo-code **47**
A.1 Ordinal Loss 47

B LLM prompting **48**
B.1 Base prompts 48
B.2 Calculators prompts 50
B.3 Enhanced prompts 54

Chapter 1

Introduction

Suicide is a significant and persistent public health crisis, demanding urgent and innovative intervention strategies. Globally, more than 700,000 people die by suicide each year, making it a leading cause of death, particularly among young adults [1]. The true scale of the issue is even larger when considering the millions more who experience suicidal ideation or attempt suicide annually. The issue is especially prevalent in low and middle-income countries (LMICs), where access to traditional mental health services is often scarce and healthcare systems are already overburdened [2]. This disparity highlights a critical gap: the very regions with the highest burden have the fewest resources for conventional, one-on-one clinical screening and intervention. Therefore, the development of scalable, accessible, and computationally efficient tools for suicide risk detection is not merely an academic exercise but a potential solution to the issue. Such tools hold the potential to augment strained healthcare systems in LMICs and offer feasible, low-cost screening solutions for organizations and companies worldwide, representing a vital step toward preventing these tragic and often avoidable deaths.

Social media usage has become significantly widespread, with nearly two-thirds of the worldwide population being active on at least one platform. Carey et al. [3] examined adolescents who had been hospitalized and analysed their social media posts in the month prior to hospital admission, finding that nearly half of the participants posted about depression, dying, or death. Social media posts may serve as early warning signs, offering a possible route for targeted intervention before a crisis escalates.

This thesis presents a comprehensive study on automated suicide risk detection from social media posts, utilizing a dataset from the r/SuicideWatch subreddit during the COVID-19 pandemic—a period marked by heightened suicide rates. The primary objective is to accurately perform multi-class risk classification across four risk levels:

indicator, *ideation*, *behaviour*, *attempt*. To achieve this, the study develops and evaluates hybrid approaches combining fine-tuned transformer models, such as RoBERTa, with prompted general-purpose large language models (LLMs) like Gemini, DeepSeek, and Grok. These approaches leverage advanced prompting strategies, including zero-shot and chain-of-thought, alongside semi-supervised methods like pseudo-labelling to address data scarcity and class imbalance, culminating in ensemble models for improved performance.

The key hypothesis made in the research are the following:

HP1 *Advanced LLM prompting strategies, such as zero-shot, chain-of-thought, and iterative self-correction, can enable general-purpose LLMs to achieve strong baseline performance in suicide risk classification, even without task-specific fine-tuning;*

HP2 *Semi-supervised techniques, including pseudo-labelling of unlabelled data using high-confidence predictions, can significantly enhance the performance of compact models like RoBERTa, addressing challenges of scarce labelled data and class imbalance while improving computational efficiency for real-world deployments;*

HP3 *An ensemble that combines a RoBERTa model (trained on labelled and pseudo-labelled data) with optimized LLM prompts offers a powerful, hybrid framework. This approach can surpass the performance of individual models, but it sacrifices the efficiency of using RoBERTa alone.*

The research yields promising results. RoBERTa-large model alone, trained on both labelled and pseudo-labelled data, achieved a validation F1-score of 73.8%, in line with top-performing LLMs used alone. A ensemble model integrating the latter RoBERTa model with optimized LLM prompts reached a weighted F1-score of 79.4% on cross-validation and 76.9% on a held-out test set, outperforming individual components and aligning with top benchmarks from related competitions like the participants of IEEE BigData 2024 Cup Challenge. The main contributions of this thesis include harnessing LLMs for suicide risk detection through innovative prompt engineering and self-supervised pseudo-labelling techniques, enabling a compact model (RoBERTa) to achieve performances comparable to much larger systems, and integrating a powerful ensemble model to surpass individual component performance.

The remainder of the document is structured as follows: Chapter 2 provides background on the evolution of computational suicide risk detection, from traditional models

to advanced transformers and LLMs, alongside current challenges. Chapter 3 describes the dataset and the selected models (RoBERTa and various LLMs). Chapter 4 details the methodology, including a two-stage approach divided into initial fine-tuning/prompting and semi-supervised pseudo-labeling, culminating in an ensemble. Chapter 5 presents the empirical results across stages, with performance metrics and visualizations. Chapter 6 discusses the findings, interpretations, and comparisons to existing literature. Finally, conclusions and future directions are outlined in Chapter 7.

Chapter 2

Background

The evolution of computational methods for suicide risk detection has been primarily shaped by inherent challenges within the field. A primary issue is the scarcity of large, high-quality annotated datasets. Developing these datasets is resource-intensive, demanding substantial domain expertise to ensure accurate and reliable labelling. Another pervasive issue is severe class imbalance. Expressions of high-risk ideation, such as specific plans or recent attempts, are far less common than lower-risk or non-risk. This imbalance complicates model training, making it difficult for algorithms to learn meaningful representations of the critical, high-risk classes. Despite these challenges, research has advanced considerably. This progress is mostly due to two key factors: the increasing availability of digital data, particularly from social media, and concurrent advancements in natural language processing (NLP). These developments have facilitated a shift from traditional statistical models to more sophisticated deep learning architectures, significantly enhancing the capabilities of suicide risk detection systems.

The following sections are meant to contextualize the present study's focus on fine-tuning transformer-based models and prompting general-purpose LLMs.

2.1 From Statistical Models to Deep Learning

Early research into computational suicide risk detection largely depended on traditional machine learning techniques. Methods such as Support Vector Machines (SVMs) and logistic regression were employed to classify texts, often demonstrating a capacity to distinguish between suicidal and non-suicidal content. For instance, a notable study showed that an SVM could differentiate genuine suicide notes from fabricated ones with an accuracy comparable to that of human experts [4]. However, these initial models

were often constrained by their reliance on hand-crafted features. This dependency meant that their performance was heavily tied to the specific datasets they were trained on, limiting their ability to generalize to new and diverse sources [5]. This limitation highlighted the need for models capable of learning relevant features automatically from the data itself.

The advent of deep learning, and specifically transformer-based models like BERT (Bidirectional Encoder Representations from Transformers) [6], marked a significant shift in the field. Unlike their predecessors, transformers utilize self-attention mechanisms that allow them to weigh the importance of different words in a sentence and capture complex semantic relationships [7]. This capability is particularly well-suited for the subtle and often ambiguous language associated with mental health and suicide risk. Numerous studies [8][9] have demonstrated that models such as BERT and its variants consistently outperform traditional machine learning approaches in suicide risk detection tasks.

2.2 Advanced Transformer and LLM Approaches

Recent efforts have concentrated on further leveraging transformer architectures, which have proven highly effective in handling complex linguistic nuances, while optimizing their efficiency through innovative techniques. One key approach is the fine-tuning of pre-trained transformer models that have already been exposed to domain-specific language, such as text from mental health forums [10] [11]. This method has shown promise in improving classification accuracy.

Another emerging strategy involves using foundational models through prompt engineering, which can achieve high accuracy even with limited training data [12][13]. For instance, Pasch and Cutura [14] demonstrated that zero-shot prompting with a refined prompt for a general-purpose LLM like GPT-4 could outperform fine-tuned models like RoBERTa on the same dataset chosen for this dissertation. zero-shot prompting allows models to perform tasks without any prior examples or fine-tuning on that specific task, relying solely on the information provided in the prompt. Additionally, to address the need for interpretability in predictions, recent works in suicide risk detection have incorporated chain-of-thought (CoT) prompting with reasoning models [15]. These approaches enable LLMs to provide step-by-step reasoning, justifying their classifications and enhancing transparency, which is crucial in sensitive domains like mental health. CoT prompting guides the model to break down complex problems into

intermediate steps, generating a series of logical reasoning steps before arriving at a final answer. Despite the general success of these prompting strategies, it's important to acknowledge that in sensitive domains like mental health, zero-shot approaches still carry the risk of hallucinations or require extremely careful prompt design to mitigate potential harm, even if they offer high accuracy in some settings [16] [17] [18] [19].

Building on prompt engineering, automated self-refinement techniques enable LLMs to iteratively improve prompts by analysing previous versions, examples, and performance metrics such as accuracy or confusion matrices. These processes have not been employed yet in this field and might be promising. Notable methods include Automatic Prompt Engineer (APE), which generates and ranks candidate prompts for optimal performance [20], and Optimization by PROmpting (OPRO), where LLMs optimize prompts through natural language descriptions of past solutions and scores [21]. Similarly, approaches like PromptBreeder and GrIPS refine prompts through self-generated evolutionary mutations or gradient-free edits based on performance feedback [22] [23].

Finally, given the limited availability of labelled data in specialized contexts, semi-supervised learning techniques are gaining traction [24][25]. One of these techniques, is pseudo-labelling: a model generates "pseudo-labels" for unlabelled data, which are then used as if they were true labels to further train the model. The generation of pseudo-labels can be done by the same model that is subsequently retrained [26], or by a different model. For example, Nguyen and Pham [27] employed a combination of two LLMs and one BERT-based model for the pseudo-labelling process. They then retrained one of the LLM models with the augmented dataset and subsequently combined various models in a final ensemble. Although pseudo-labelling carries inherent risks, such as the propagation of errors from noisy labels or the potential for reinforcing biases present in the initial, limited labelled dataset, which is particularly pertinent for sensitive mental health data, its potential to significantly improve model performance in data-scarce scenarios makes it a promising approach [28], [29].

Chapter 3

Dataset and Models Employed

This chapter outlines the core components of the research: the dataset for suicide risk detection in social media posts and the selected models. The selection of each was guided by the core objective of accurately classifying suicide risk in social media text, addressing the key challenge of a small, imbalanced set of labelled data sourced from the IEEE BigData 2024 Cup Challenge. This dataset, originating from the r/SuicideWatch subreddit, necessitates specialized modelling and evaluation strategies.

The research employs two types of models. First, RoBERTa, a specialized transformer, is chosen for its efficiency and fine-tuning capabilities. Second, several state-of-the-art, general-purpose LLMs—including Gemini, DeepSeek, and Grok families—are used both as powerful classifiers and as ”teacher” models for a pseudo-labelling strategy to expand the training data.

3.1 Dataset

The dataset used was constructed for the IEEE BigData 2024 Cup Challenge on Suicide Ideation Detection [30]. Composed of 2000 training and 200 testing samples, it was originally obtained from another dataset composed of 139,455 posts from 76,186 users, published in the subreddit r/SuicideWatch from 01/01/2020 up to 31/12/2021 ([31]). This timeframe was selected specifically to cover the COVID-19 pandemic, which produced a significant increase of suicide rates. Out of the 2000 training posts, 500 were annotated each with four increasing levels of risk to commit suicide: *indicator*, *ideation*, *behaviour*, *attempt*. The annotation criteria for each level is reported in Table 3.1 and was approved by domain experts who followed the Columbia-Suicide Severity Rating Scale [31]. In the original competition, each developed model was subjected

to a primary evaluation with 100 test posts, followed by a final evaluation using the remaining 100.

The dataset was acquired by contacting the organizers and required signing a Data Usage Agreement (DUA) from The Hong Kong Polytechnic University. This DUA permits a non-exclusive license to use the dataset solely for non-commercial, educational, or research purposes only, with restrictions (e.g., prohibiting identification of research subjects), all of which were strictly observed. However, 100 out of the 200 test posts were not provided. Further analysis revealed that the 500 training posts did not exactly match those from the competition, with slight variations in class proportions. Additionally, only 30 of the 100 provided test posts were unique (i.e., absent from the training or unlabelled data). Despite this, the class proportions (as detailed in Table 3.1) closely resembled those in the training set, indicating they likely share the same distribution. For this study, training and evaluation was carried on all the 2000 posts provided and the 30 unique test posts were reserved as a held-out test set.

3.2 RoBERTa

RoBERTa, which stands for a Robustly Optimized BERT Pretraining Approach, is an open-source enhancement of the BERT (Bidirectional Encoder Representations from Transformers) model. The original BERT model [6] is a transformer-based architecture pre-trained on a large corpus of text to understand language context from both left and right directions simultaneously. As illustrated in Figure 3.1, this was achieved by training it on two distinct tasks. The first involved masking some percentage of the input tokens and then predicting those masked tokens, while the second involved choosing which sentence followed the other in between two distinct ones.

RoBERTa builds upon BERT by modifying key pre-training strategies [32]. The modifications include training on a significantly larger dataset, removing the next sentence prediction objective from the pre-training, and dynamically changing the masking pattern applied to the training data. These changes allow RoBERTa to achieve improved performance on various natural language understanding tasks. For specific applications such as classification, the model is adapted by fine-tuning the pre-trained architecture on a task-specific dataset, as done in this research, which employed RoBERTa in both large (355 million parameters) and small (125 million parameters) version. The choice is justified by previous efforts for the IEEE BigData 2024 Cup Challenge, which obtained the best non-LLM models results with it. [14] and [26] respectively demonstrated strong

Table 3.1: Detailed definition of different suicide risk levels and distribution of annotated training and test set posts.

Risk level	**Definition**	**Posts (Train)**	**Posts (Test)**
Indicator (IN)	The post content has no explicit suicidal expression or has explicit expression of resolved suicidal issues.	129 (25.8%)	6 (20.0%)
Ideation (ID)	The post content has explicit suicidal expression but there is no plan or tendency to commit suicide.	200 (40.0%)	12 (40.0%)
Behaviour (BR)	The post content has explicit suicidal expression and there is a plan or tendency to act out self-harm or suicide, or mention of historical experience of self-harm behaviour.	132 (26.4%)	10 (33.3%)
Attempt (AT)	The post content has explicit expression concerning a recent suicide attempt, or mention of historical experience of a suicide attempt.	39 (7.8%)	2 (6.7%)

performance with a weighted F1-score of 71% and 75% for a reasonably small model.

Despite the relative ”smallness” and ”age” of RoBERTa compared to the latest massive LLMs, there is still significant potential to enhance its performance. One promising approach is through pseudo-labelling the unlabelled data, making use of a combination of the most recent, general LLMs.

3.3 General-purpose Large Language Models

General-purpose Large Language Models have been improving significantly in recent years, with a variety of models being constantly released, offering a wide range of capabilities and performance levels. For this research, a selection of cutting-edge LLMs were employed to pseudo-label the training data for training RoBERTa, but also as standing models themselves. Each of the employed families and versions are reported

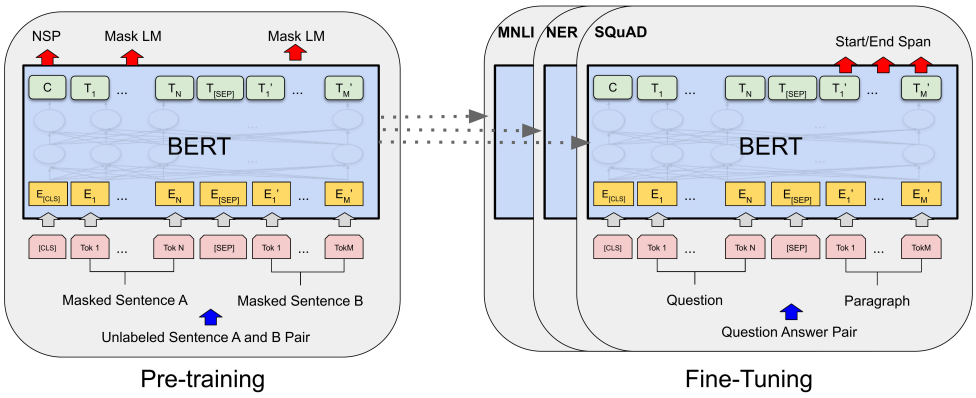


Figure 3.1: Illustration of BERT's pre-training (left) and fine-tuning (right) processes from the original paper of Devlin et al. [6]. During pre-training, the model learns bidirectional context from unlabelled sentence pairs by masking and predicting random words (Masked LM) and determining if one sentence follows the other (Next Sentence Prediction). In fine-tuning, the pre-trained parameters initialize the model for specific tasks, such as question answering on paired inputs, by adding minimal output layers and adjusting all weights on labelled data, enabling effective adaptation to various NLP applications with little architectural alteration.

below.

3.3.1 Gemini

The Gemini family of models, developed by Google, represents a significant leap in multimodal and long-context reasoning [33]. For this study, models from both the Gemini 2.0 and the more recent Gemini 2.5 series were utilized, enabling effective analysis of extended textual contexts in social media posts for subtle indicators of suicide.

- **Gemini 2.0** is Google's first publicly available LLM able to challenge competitors. While it didn't surpass the best models such as the ones developed by OpenAI and Anthropic in all metrics, it was notable for its lightweight design and efficiency. It built upon the foundational architecture of its predecessors, offering a balance of speed and capability that made it suitable for a wide array of NLP tasks. Key models in this family included `Gemini 2.0 Flash` and `Gemini 2.0 Flash-Lite`.

- **Gemini 2.5** introduces further enhancements, with `Gemini 2.5 Pro` being the

most powerful model in this line. It is a thinking model, capable of reasoning through its thoughts before responding, resulting in enhanced performance and improved accuracy. It possesses state-of-the-art performance in reasoning, coding, and multimodal understanding, which aids in interpreting nuanced emotional cues and multimodal data (e.g., text with images) to detect suicide risk. The family also includes more cost-effective versions like `Gemini 2.5 Flash`, which are optimized for low-latency applications while still maintaining a high level of performance.

The most powerful of these, Gemini 2.5 Pro, demonstrates impressive capabilities across a range of benchmarks. For instance, in the Massive Multitask Language Understanding (MMLU) benchmark, a key measure of a model's general knowledge and problem-solving abilities, Gemini 2.5 Pro achieves a score of 90%. In mathematical reasoning, as measured by the AIME benchmark, it scores a notable 92.0% [33], facilitating affordable yet accurate identification of suicide-related language in large datasets.

3.3.2 DeepSeek

The DeepSeek models, developed by DeepSeek AI, have garnered significant attention for their open-source nature, impressive performance, and cost-effectiveness [34]. This research utilized both the V3 and the specialized R1 models, whose architectures are detailed by Zhao et al. [35].

- **DeepSeek V3** has been trained on a massive and diverse dataset. It is designed to excel at a wide range of natural language and coding tasks.
- **DeepSeek R1** is a model specifically fine-tuned for complex reasoning. It builds upon the V3 architecture but incorporates "thinking" capabilities, where the model explicitly reasons through a problem before providing an answer. This makes it particularly well-suited for tasks that require logical deduction and in-depth analysis.

The primary motivation for including DeepSeek was its remarkable combination of low cost and high performance, even with a relatively small model size compared to other leading ones. It obtained a score of 79.8 on AIME and 90.8 on MMLU. Its efficiency and strong general-purpose performance made it an ideal candidate for

generating high-quality pseudo labels in a cost-effective manner, thereby supporting a scalable suicide risk detection pipeline.

3.3.3 Grok

The Grok family of models is developed by xAI and is designed to provide a deep understanding of language. For this work, the ”mini” and ”normal” variants of Grok 3 were used.

- **Grok 3 Mini** is a more compact and efficient version of the Grok 3 model. It is designed to offer a good balance between performance and resource consumption, making it suitable for a wide range of applications where speed and cost are important factors.
- **Grok 3** represents the full-scale, powerful version of the model. It was trained in largest supercomputing cluster on the planet, which at the time of training contained 200,000 H100 GPUs.

The most powerful in this series, Grok 3, has demonstrated strong performance on various benchmarks. It achieves a score of 81.5% on the MMLU benchmark and 93.3% on the AIME, showcasing its advanced problem-solving capabilities [36]. Despite the Grok family’s impressive language and problem-solving abilities, its application to mental health remains largely unresearched. This gap justifies the choice to use Grok, offering a unique opportunity to provide novel insights into its potential in this critical area.

Chapter 4

Methodology

This chapter presents the methodology employed for developing and evaluating automated suicide risk classification systems for social media posts. The research adopts a two-stage approach designed to make use of both labelled and unlabelled data, and was structured to progressively build upon initial models to enhance classification performance.

The first stage establishes baseline performance through traditional supervised learning approaches, employing both transformer-based models, trained exclusively on the labelled dataset, and general-purpose LLMs.

The second stage builds on the models developed in the first stage to implement a pseudo-supervised learning approach and a final ensemble to maximize performance.

To evaluate performance on the imbalanced dataset, the weighted F1-score is the primary metric, supplemented by the macro-averaged F1-score and accuracy. Statistical reliability is ensured through bootstrapping and cross-validation.

The following sections detail the specific implementations, experimental designs, and evaluation procedures employed in each stage of the research.

4.1 Stage 1: Fine Tuning and Prompt Design

This stage employs both fine-tuning and prompting approaches, using exclusively the labelled dataset.

Three complementary approaches are explored: first, fine-tuning RoBERTa with different loss functions (cross-entropy, ordinal, and differentiable F1-score) optimized through stratified cross-validation; second, systematic LLM prompting by zero-shot, chain-of-thought (COT); and third, an automated prompt refinement process where

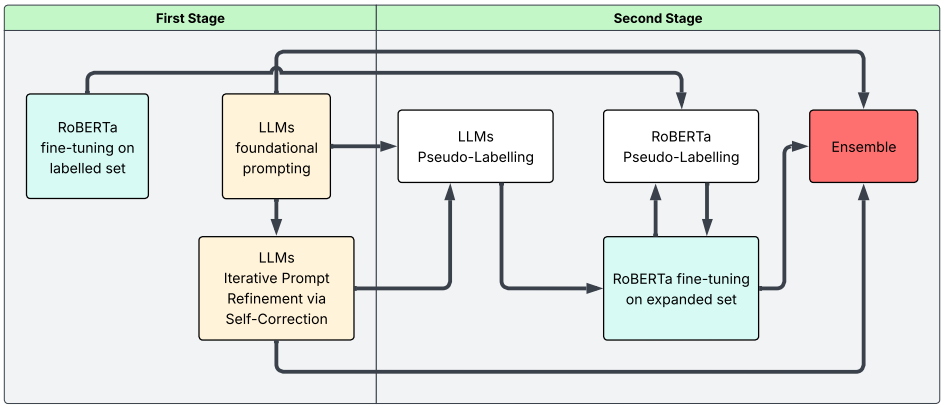


Figure 4.1: Process flow for the two stage approach followed. During the first stage RoBERTa is initially finetuned on the labelled set and LLMs are prompted with zero-shot and CoT. The CoT version is subsequently used in an iterative prompt refinement process. Stage two employs the best performing approaches from the first stage pseudo-label the unlabelled data, expanding the training dataset and allowing further fine-tunings of RoBERTa. Finally, the best models are incorporated into an ensemble to maximize performance.

LLMs analyse their own performance to improve classification accuracy.

The outputs serve two purposes: establishing baseline performance metrics and generating high-quality predictions for pseudo-labelling unlabelled data in the second stage. The systematic evaluation ensures that only the most reliable models contribute to expand the training dataset.

4.1.1 Initial RoBERTa fine-tuning

An initial fine-tuning of RoBERTa was carried using the labelled data only (500 posts). In order to identify the best fine-tuning setup for the model various combinations of different loss functions and training arguments were tested using 5-fold cross-validation. These folds were stratified to maintain the initial class proportions, as done by [27]. The fine-tuning for this and the subsequent chapters' RoBERTa models was conducted on a single NVIDIA RTX 3080 GPU with 10 GB of VRAM.

To make the training process more efficient, Low-Rank Adaptation (LoRA) was employed. LoRA is a parameter-efficient fine-tuning technique that freezes the pre-trained model weights and uses trainable rank decomposition matrices for the layers of the transformer architecture [37]. This significantly reduces the number of trainable

parameters and memory requirements. The LoRA hyper-parameters were selected as rank=16, alpha=32, and dropout=0.05 based on empirical tuning that balanced model adaptability and computational efficiency on the available GPU. The adaptation was specifically applied to the attention mechanism's.

Although the maximum number of tokens for a dataset post is 2930, Suzuki [24] shows that the majority consist of 200 or fewer tokens, which is why the input token length was limited to 512, further reducing memory usage and computational load.

Three losses were tested: the first loss function employed was the cross-entropy loss. This is arguably the most common and reliable loss function for classification tasks, particularly in deep learning [38][39][40]. For multi-class classification, it is often combined with a softmax activation function on the model's output to convert raw scores into probabilities [38], as done in this research.

Secondly, an ordinal loss was implemented. The underlying idea was to transform the problem into an ordinal regression task, given the inherently ordered nature of the classification classes. This approach assumes that misclassifying a sample by a larger margin (e.g., predicting class 1 when the true class is 5) should be penalized more severely than a smaller margin (e.g., predicting class 2 when the true class is 1). The pseudocode for the OrdinalTrainer is reported in Appendix A.1.

This implementation utilizes torch.nn.BCEWithLogitsLoss, which is suitable for multi-label binary classification or, in this context, for an ordinal setup where each "label" can be interpreted as a binary decision point along the ordered scale. To enable this, the ground truth labels are encoded as cumulative multi-hot vectors —for example, a class 2 label in a 5-class problem would be encoded as [1, 1, 0, 0].

Finally, the third loss function chosen was inspired by research aimed at directly optimizing the F1-score, the primary metric. Since the F1-score is non-differentiable, making it unsuitable for direct gradient-based optimization, the Macro Double Soft F1 loss was adopted, as introduced by [27]. The paper states that optimizing with this loss function can lead to performance gains compared to conventional choices like cross-entropy, as demonstrated in their experiments.

The hyper-parameters, including 5-fold cross-validation, a learning rate of 5e-5, 100 epochs, batch size of 2 with 8 gradient accumulation steps, 100 warm-up steps, and the AdamW optimizer (a variant of Adam that decouples weight decay from the gradient update [41]), were chosen through empirical testing on the limited labelled dataset and RTX 3080 GPU constraints, ensuring stable convergence and optimal performance.

To leverage the entirety of the labelled data, an additional, alternative version of

RoBERTa-large was fine-tuned during this first stage. This specific model was trained on the complete set of 500 labelled posts, without being split into separate training and validation sets. The selection of its loss function, hyperparameters, and number of training epochs was based on the best results observed during the earlier cross-validation experiments, using the weighted F1 score as the primary selection criterion. The only exception was the warm-up steps. For this full-dataset fine-tuning, a warm-up schedule of $N_{\text{training examples}}/4$ steps (which equated to 125 steps for the full training set) was chosen. This approach follows the recommendations from [42], which suggests adjusting warm-up steps based on factors like dataset size, model architecture, and optimizer.

4.1.2 LLMs foundational prompting

A variety of prompting strategies were employed to assess the classification capabilities of several LLMs. Specifically, models from the Gemini family were accessed via the google-generative-ai package, while DeepSeek and Grok models were utilized through their OpenAI-compatible endpoints using the openai package. All models were executed with carefully selected hyperparameters chosen through empirical testing, shown in Table 4.1. For non-thinking, zero-shot prompts, a deterministic settings was employed across all models, while for COT reasoning prompts, model-specific configurations were applied.

Table 4.1: LLM Prompting Hyperparameter Configurations

Prompt Type	**Model**	**Temperature**	**Top P**	**Top K**
Non-thinking	All	0	1	1
Chain-of-Thought	Gemini	0.4	0.95	64
Chain-of-Thought	DeepSeek	0.5	0.95	N/A
Chain-of-Thought	Grok	0.4	0.95	N/A

A critical consideration involves limiting the generated tokens. The relative parameter was set to 4096 tokens for Grok and DeepSeek, and 8192 for Gemini models-substantially higher than typical classification outputs required. This design choice served two purposes: First, it suited verbose reasoning chains in CoT prompts without truncation. Second, and most importantly, it prevented safety filter triggers that consistently occurred when lower token limits were specified, even with the zero-shot prompts. Truncated outputs are often deemed harmful by filters because they may interrupt the

model's self-correction process, leaving partial unsafe reasoning exposed without the full context that resolves potential violations [43].

As a foundational approach, a zero-shot prompting strategy was established. This initial prompt, referred to as the *Base* prompt, was constructed directly from the class definitions outlined in Table 3.1. This represents the most straightforward method, providing the model with a direct request, the descriptions of each risk category, and constraints for the output format, without offering any examples.

In parallel, a CoT approach was implemented. This more advanced strategy augmented the zero-shot prompt by instructing the model to first articulate a detailed thinking process before providing the final classification. The response format was adjusted to accommodate this explicit reasoning step. This CoT prompting strategy was applied exclusively to LLMs specifically designed with inherent reasoning capabilities, ensuring they could effectively handle such structured thinking processes and benefit from them. Both the zero-shot and the CoT versions are included B.1.

A data-processing pipeline was subsequently developed to parse the outputs from both the zero-shot and CoT methods, accurately extracting the final classification labels.

The work of [14] was then incorporated. They investigated various non-thinking (zero-shot) prompting options for GPT-4, GPT-4-mini, and GPT-4-turbo. Their most effective prompt, which will be referred as the *Calculators* prompt, was also adopted and tested in this study (see Appendix B.2). Additionally, to ensure methodological consistency, a CoT version of their prompt was created, modifying it in a similar fashion to how the *base* prompt was adapted into its "thinking" counterpart (also included in Appendix B.2).

The adoption of the *Base* and *Calculators* prompts was ultimately guided by a performance comparison conducted using Gemini 2.5 Flash, a choice justified by its lightweight, robustness and high performance shown during preliminary test. In light of this evaluation, the *Calculators* version was selected for both non-thinking and thinking configurations, and subsequently evaluated across all selected models.

4.1.3 Iterative Self-Correction by LLMs

To enhance the performance of suicide risk classification from social media posts, a meta-learning approach was employed, using the LLMs themselves to refine the *Calculators* prompt. Inspired by works such as by Yang et al. [21] and Zhao et al. [44], the core hypothesis was that these models could identify their operational patterns,

identifying shortcomings in their reasoning, and subsequently adapt the prompt to mitigate these deficiencies. This method of ”self-correction” was implemented by providing the thinking models with a performance overview. This overview included the original thinking prompt, a confusion matrix, and other relevant metrics (see section 4.3) derived from the initial classification task. To provide insights on specific strength and issues, examples of successful and failed classifications for each risk category were also included. However, due to the limited context window of the models, the number of these examples was restricted: Gemini 2.5 Flash and DeepSeek R1 received one example of a success and one of an error per class, while Grok-3-mini, with a slightly larger capacity, was provided with two of each. The inclusion of the lengthy ”thinking” process in the input was considered crucial for the models to analyse their reasoning, despite the constraint it placed on the number of examples.

Here follows the prompt used.

```
I am trying to improve a prompt for classifying suicide risk from
    social media posts into four categories: Indicator, Ideation,
    Behavior, and Attempt.
Here is my current prompt:{{prompt}}
---
Here are my confusion matrix and some other metrics for my current
    prompt: {{metrics}}
---
Here are some examples of failures and successes with my current
    prompt: {{examples}}
---
Based on my prompt, the confusion matrix and these examples, provide
     a new prompt that will improve classification performance.
```

Listing 4.1: Enhancing prompt with examples of success and failure

The iterative improvement process began with the best-performing model from each of the three families: Gemini 2.5 Flash, DeepSeek R1, and Grok-3-mini. Each of these models’ initial prompts was then subjected to an enhancement process through itself. Following this, the newly refined prompts were used to re-evaluate the models’ classification performance. In a second iteration, a further attempt was made to enhance the already improved DeepSeek R1 prompt using its own evaluation results. This, however, resulted in a decline in performance. A similar self-improvement attempt was made with Gemini 2.5 Flash, where the model was tasked with refining its prompt based on its own output, which also led to underperformance compared to the previous

iteration.

Further iterations and variations (e.g., cross-model enhancement or error-only examples) were tested but led to no improvements.

4.2 Stage 2: Pseudo-Labelling and Ensemble

The second stage of this process transitions from supervised to semi-supervised learning by integrating pseudo-labelling of unlabelled data, a technique where a model trained on a small, labelled dataset predicts labels for a much larger, unlabelled one. The best models from the first stage act as the ”teacher,” generating high-fidelity pseudo-labels for vast quantities of otherwise unlabelled text, even for complex linguistic patterns. These predicted labels are then treated as ground truth to augment the training data for the RoBERTa model. The underlying assumption is that the initial model's high-confidence predictions are largely correct, and their inclusion helps refine RoBERTa's decision boundaries and improve its generalization. This process effectively expands the labelled dataset without the high cost and time of manual annotation, holding the potential for a smaller, specialized model like RoBERTa to reach or even surpass the performance of state-of-the-art LLMs, while being orders of magnitude less computationally expensive. Finally, an ensemble combining the refined RoBERTa with selected LLMs is employed to achieve more robust and accurate classifications through weighted aggregation of predictions.

4.2.1 LLM Pseudo-Labelling

To generate the initial set of pseudo-labels, a few random samples were manually reviewed to confirm they came from the same distribution as the labelled set. Following this, the 1500 unlabelled social media posts were processed by the best-performing LLMs identified in the first stage, considering only instances where all the models agreed on the label and discarding the rest. Specifically, the four models were utilized: Gemini 2.5 Flash non-thinking, DeepSeek R1 thinking, Grok 3 mini thinking, and Grok 3 mini non-thinking. Various combinations of such models were evaluated on the training set to identify the configurations that yielded high confidence pseudo-labels while maintaining an appropriate agreement and class proportions. The detailed performance metrics, including agreement percentages and class distribution analyses for these LLM combinations, are presented in Section 5.2 of the Results chapter. Based

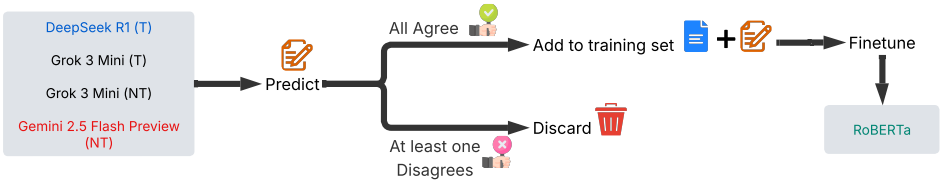


Figure 4.2: Schema of the fine-tuning of RoBERTa using pseudo-labels generated from LLMs only. On the left, the first block comprising the 4 chosen LLMs predicts the labels. If all modes agree, the example is added to the training set otherwise discarded. Finally, RoBERTa is fine-tuned again on the enhanced training set.

on these findings, the combination of all four models was ultimately chosen, as it achieved almost a 70% agreement rate, strong overall accuracy, and class distributions closely aligned with those in the labelled training set. The resulting pseudo-labelled posts were then incorporated into each of the five data splits used for cross-validation, and RoBERTa was subsequently fine-tuned again on this augmented dataset. Consistent with the optimal configuration identified in the first stage, Cross-Entropy loss, a learning rate of 5e-5 and a warmup schedule of $N_{\text{training examples}}/4$ were used. A schema of such process is depicted in Figure 4.2.

4.2.2 Iterative Pseudo-Labelling with RoBERTa

In a further effort to refine the quality of the pseudo-labels, an additional layer of filtering was introduced (Figure 4.3). In this case the condition for accepting a pseudo-label was made more stringent: not only did the four LLMs have to be in unanimous agreement, but their collective prediction also had to match the prediction from the RoBERTa model that had been fine-tuned solely on the original 500-post training set. This verification was conducted on a per-fold basis. Specifically, for each of the five folds, predictions for the 1500 unlabelled posts were generated by its respective baseline RoBERTa model. The pseudo-labels that met this dual-agreement criterion (unanimous LLM consensus and alignment with the fold's baseline RoBERTa) were then integrated into that specific training fold, creating a new, more selectively enriched dataset, where the five RoBERTa models were trained once again on these newly refined data splits -this data/step will be referred as *Iterative 1*-, and the overall performance metrics were calculated as the average across all five folds. Given the notable performance improvements observed in *Iterative 1*, it was hypothesized that a second iteration might yield even greater gains. An

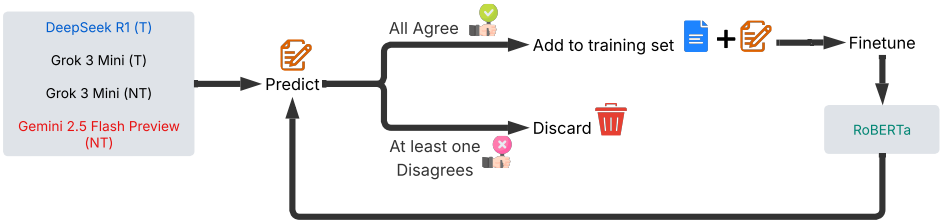


Figure 4.3: Schema of the iterative fine-tuning of RoBERTa using pseudo-labels generated from RoBERTa itself and LLMs. The process is identical to Figure 4.2, except that RoBERTa participates in the predictions for pseudo-labelling along the other 4 LLMs. The process is iterative because after RoBERTa is fine-tuned again on the enhanced dataset, the new fine-tuned version can be used for a new set of predictions.

additional iteration of this enrichment process was explored, using the models trained on the first enriched set as the new benchmark for agreement-*Iterative 2*-. However, this led to a decrease in performance, indicating that the point of maximum benefit from this semi-supervised approach had been already reached with *Iterative 1*. Consequently, the iterative process was not pursued further.

Following the identification of the first iteration as the most effective approach for the cross-validated models, the RoBERTa-large version that was fine-tuned on all 500 labelled posts (the one without 5-fold cross-validation, mentioned in Section 4.1.1) also underwent the exact same iterative pseudo-labelling process. This specific RoBERTa-large model, having been trained on all original labelled data and then enhanced through the first iteration of pseudo-labelling, was subsequently used for the final ensemble.

4.2.3 Ensemble

A final phase involved constructing an ensemble method to aggregate predictions from multiple models. The goal was to obtain a more robust and accurate final classification.

The ensemble combined the predictions of three distinct models. These included two LLMs that demonstrated the strongest performance: Gemini 2.5 Flash (non-thinking with the *calculators* prompt) and DeepSeek R1 (thinking with the *calculators* prompt enhanced once). The third component was the RoBERTa model obtained from the *Iterative 1* fine-tuning stage, chosen for its superior performance.

Predictions were aggregated using a weighted average of the one-hot encoded predictions. Although RoBERTa had logits available (probabilities for each class) that could have been used in the weighted average instead, testing showed no improve-

ments—likely because the model learned clear boundaries between classes but not accurate probability scores. The weight assigned to each model's prediction was its respective validation Weighted F1-score, as determined during its individual evaluation. This strategy ensured that models demonstrating higher accuracy on the validation data had a greater influence on the final predictions.

First, concerning the 5-fold cross-validation version of RoBERTa, the entire ensemble process was executed independently for each of the five cross-validation folds. The final reported performance metrics for this version represent the average results across all folds.

Finally, given the satisfactory results of the latter, the RoBERTa-large model that was initially fine-tuned on all 500 labelled data was then subsequently processed through the *Iterative 1* pseudo-labelling stage. This yielded the RoBERTa version that was included in the ensemble for the final evaluation.

4.3 Evaluation Criteria

Given the imbalanced nature of the dataset, with varying proportions across the four suicide risk levels (as shown in Table 3.1), evaluation metrics were selected to provide a balanced assessment of model performance. Following competition standards [30], the primary metric used is weighted F1-score, which accounts for class imbalance by Weighting the F1-score of each class by its support (number of true instances). This emphasizes performance on more prevalent classes while still considering all categories. Additional metrics include accuracy-providing a measure of general performance- and macro average F1-score-the unweighted mean of F1-scores across all classes, treating each class equally regardless of support, highlighting performance on minority classes like ”attempt”-.

To quantify uncertainty for RoBERTa fine-tuning, 5-fold stratified cross-validation was used to maintain class proportions, with metrics and errors obtained by averaging across folds. For LLMs and the ensemble final evaluation (in absence of a validation set), a bootstrap resampling approach was employed. Bootstrap is a statistical technique that enables the estimation of the sampling distribution of a metric by repeatedly resampling the observed data, thus allowing for the quantification of uncertainty in performance metrics, especially when only a limited amount of labelled data is available. 1000 iterations of resampling with replacement were performed, calculating metrics for each sample, and reporting the mean and standard deviation.

Chapter 5

Results

This chapter presents the empirical findings from the multi-stage experimental process detailed in the previous chapter. The results are organized to mirror the methodology: beginning with the foundational performance benchmarks on the labelled dataset, followed by the outcomes of the semi-supervised pseudo-labelling stages, finishing with the evaluation of the final ensemble model.

5.1 Stage 1

5.1.1 RoBERTa Initial Fine-tuning Performance

The performance results of RoBERTa-small and RoBERTa-large across various loss functions, averaged across the 5-folds, are presented in Tables 5.1 and 5.2. The reported metrics, mean and standard deviation, were computed directly from the five distinct performance scores obtained for each fold.

Table 5.1: RoBERTa-small inistial fine-tuning results for various combination of loss functions and learning rates. The metrics are averaged across the 5-folds and correspond to the epoch with the highest Weighed F1 score.

Loss	**lr**	**Best Epoch**	**Accuracy** (%)	**Weighted F1** (%)
Cross-Entropy	**5e-5**	**15**	$\mathbf{65.8 \pm 2.7}$	$\mathbf{64.5 \pm 2.4}$
Ordinal	5e-5	14	64.0 ± 3.9	62.4 ± 4.7
Soft F1	5e-5	63	64.0 ± 5.2	63.1 ± 5.7

For RoBERTa-small, the results in Table 5.1 indicate that the Cross-Entropy loss with a learning rate of 5e-5 yielded the best performance. This configuration achieved

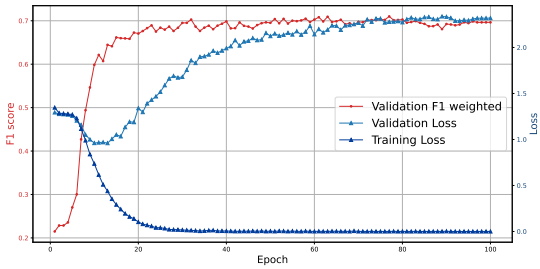
Table 5.2: RoBERTa-large initial fine-tuning results for various combination of loss functions and learning rates. The metrics are averaged across the 5-folds and correspond to the epoch with the highest Weighed F1 score.

Loss	**lr**	**Best Epoch**	**Accuracy** (%)	**Weighted F1** (%)
Cross-Entropy	1e-4	25	70.6 ± 2.9	70.1 ± 3.2
Cross-Entropy	**5e-5**	**77**	$\mathbf{71.8 \pm 4.9}$	$\mathbf{71.0 \pm 5.2}$
Ordinal	5e-5	29	68.4 ± 3.7	67.3 ± 4.2
Soft F1	5e-5	90	69.4 ± 4.2	68.7 ± 4.5

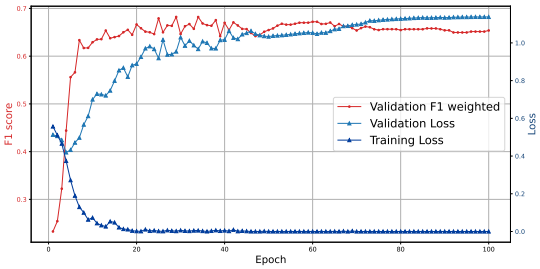
a weighted F1 score of 64.5%, peaking at an average of 15 epochs. The other loss functions, Ordinal and Soft F1, resulted in slightly lower performance metrics.

The experiments with RoBERTa-large, summarized in Table 5.2, show a similar trend but with notably higher performance overall. Once again, the Cross-Entropy loss proved to be the most effective, particularly with a learning rate of 5e-5. This combination achieved the highest weighted F1 score of 71.0% and an accuracy of 71.8%, with the best performance observed at 77 epochs. While the same loss function with a higher learning rate of 1e-4 also performed well, it did not surpass the 5e-5 configuration. Both the Ordinal and Soft F1 losses, though competitive, did not match the results of the standard Cross-Entropy loss.

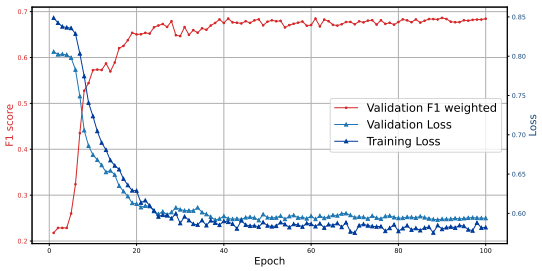
Figure 5.1 provides a visual representation of the training dynamics for RoBERTa-large with a learning rate of 5e-5 across the three different loss functions. The plots illustrate the validation weighted F1 score progression over the training epochs. The accuracy curve is not reported in the plots as its trajectory was observed to be very similar to the weighted F1 score in every case. For all three loss functions, the training loss consistently decreases before reaching a plateau. However, the validation loss behaves differently depending on the function. Interestingly, only the Macro Double Soft F1 validation loss mirrors the training loss, decreasing steadily without a subsequent rise. In contrast, the validation losses for Cross-Entropy and Ordinal loss decrease for only a few epochs before beginning to rise drastically. Despite this being a common overfitting sign, their corresponding validation Weighted F1 scores continue to increase, suggesting the model is still improving its classification on the metric of interest. This visual evidence further supports the quantitative results and the selection of Cross-Entropy as the most reliable and highest-performing loss for this task. Based on these findings, the configuration of Cross-Entropy loss with a learning rate of 5e-5 was chosen



(a) Cross-Entropy



(b) Ordinal



(c) Macro Double Soft F1

Figure 5.1: Training plots of RoBERTa-large with learning rates of 5e-5 for the three different losses.

for the subsequent stage of the research.

5.1.2 LLM Foundational Prompting Performance

Before conducting a full-scale comparison across all models, a preliminary experiment was performed to determine the most effective prompt structure. This initial evaluation carried on the Gemini 2.5 Flash, comparing the performance of the *Base* and *Calculators* prompts, is presented in Table 5.3. Both non-thinking (NT) and thinking (T) versions of each prompt were included to identify the optimal approach for the classification task. The data reveals a substantial performance advantage for the *Calculators* prompt across both thinking and non-thinking modalities. In the non-thinking configuration, the *Calculators* prompt achieved an accuracy of 75.4%, a striking improvement of nearly 25 percentage points over the *Base* prompt's 60.6%. A similar, although slightly less pronounced, superiority was observed in the thinking versions, where the *Calculators* prompt outperformed the *Base* prompt by over 6 percentage points in accuracy. Across all metrics, the *Calculators* prompt demonstrated a clear and consistent advantage.

Given this decisive outcome, the *Calculators* prompt family was selected for the evaluation of all LLMs in this study. The subsequent results, therefore, refer exclusively

Table 5.3: Comparison of Accuracy, Weighted and Macro Average F1 Scores for the *Base* and *Calculators* prompts on Gemini 2.5 Flash.

Type	Prompting	**Accuracy** (%)	**Weighted F1** (%)	**Macro Avg F1** (%)
NT	*Base*	60.6 ± 2.1	59.3 ± 2.3	62.4 ± 2.2
	Calculators	$\mathbf{75.4 \pm 1.9}$	$\mathbf{74.4 \pm 2.0}$	$\mathbf{74.3 \pm 2.1}$
T	*Base*	69.2 ± 2.0	68.1 ± 2.1	69.2 ± 2.1
	Calculators	$\mathbf{73.4 \pm 2.9}$	$\mathbf{72.7 \pm 2.1}$	$\mathbf{72.7 \pm 2.2}$

to this superior prompt design.

Table 5.4 and 5.5 present a comparison of the employed LLMs using the selected *Calculators* prompts. The first table uses the non-thinking version, while the second utilizes the thinking version.

Table 5.4: Comparison of Accuracy, Weighted and Macro Average F1 Scores for for the *calculators* non-thinking prompt.

Model	**Accuracy** (%)	**Weighted F1** (%)	**Macro Avg F1** (%)
Gemini 2.0 Flash lite	68.3 ± 2.0	67.6 ± 2.1	66.3 ± 2.5
Gemini 2.0 Flash	66.1 ± 2.1	64.3 ± 2.3	62.8 ± 2.6
Gemini 2.5 Flash	$\mathbf{75.4 \pm 1.9}$	$\mathbf{74.4 \pm 2.0}$	$\mathbf{74.3 \pm 2.1}$
Gemini 2.5 pro	74.5 ± 1.9	73.1 ± 2.1	73.0 ± 2.2
DeepSeek V3	66.9 ± 2.1	64.6 ± 2.3	62.3 ± 2.7
DeepSeek R1	$\mathbf{66.1 \pm 2.1}$	$\mathbf{66.1 \pm 2.1}$	$\mathbf{65.6 \pm 2.5}$
Grok 3 Mini	$\mathbf{75.6 \pm 1.9}$	$\mathbf{75.0 \pm 2.0}$	$\mathbf{74.8 \pm 2.1}$
Grok 3	72.5 ± 2.0	71.4 ± 2.1	69.7 ± 2.4

For the calculators non-thinking prompt, Grok 3 Mini and Gemini 2.5 Flash emerged as the top-performing models. Grok 3 Mini achieved the highest accuracy at 75.6%, closely followed by Gemini 2.5 Flash at 75.4%. Their weighted and macro average F1 scores were also comparable, indicating a robust and balanced performance across the different classes in the dataset. The other models, including the larger variants like Gemini 2.5 pro and Grok 3, exhibited lower performance across all metrics. All versions of DeepSeek and Gemini 2.0 lagged behind the Gemini 2.5 and Grok 3 families in this non-thinking prompt scenario.

The introduction of the calculators thinking prompt led to a general improvement in the performance of several models. Grok 3 Mini once again demonstrated the

Table 5.5: Comparison of Accuracy, Weighted and Macro Average F1 Scores for the *calculators* thinking prompt.

Model	**Accuracy** (%)	**Weighted F1** (%)	**Macro Avg F1**(%)
Gemini 2.5 Flash	$\mathbf{73.4 \pm 2.0}$	$\mathbf{72.7 \pm 2.1}$	$\mathbf{72.7 \pm 2.2}$
Gemini 2.5 pro	73.0 ± 2.0	71.4 ± 2.2	62.6 ± 7.2
DeepSeek R1	$\mathbf{73.8 \pm 1.9}$	$\mathbf{73.5 \pm 2.0}$	$\mathbf{73.7 \pm 2.2}$
Grok 3 Mini	$\mathbf{74.8 \pm 1.9}$	$\mathbf{74.5 \pm 2.0}$	$\mathbf{73.9 \pm 2.1}$
Grok 3	71.6 ± 2.0	71.1 ± 2.1	70.9 ± 2.3

highest accuracy, with a score of 74.8%. However, the most significant performance gain was observed in the DeepSeek R1 model, which saw its accuracy jump to 73.8% and its weighted F1 score increase to 73.5%. This suggests that the thinking prompt strategy is particularly effective for the DeepSeek architecture. While Gemini 2.5 Flash also performed well with the thinking prompt, its scores were slightly lower than its performance with the non-thinking prompt. Notably, the standard deviation of the macro average F1 score for Gemini 2.5 pro increased drastically, suggesting a less stable performance when using the thinking prompt.

5.1.3 Iterative Prompt Refinement via Self-Correction

Building on the foundational performance results, Table 5.6 shows the result of the iterative refinement process. As detailed in the Methodology chapter, this self-correction methodology utilizes the models' own outputs and performance insights to identify and rectify systematic errors. The prompt was enhanced with explicit instructions and examples aimed at preventing these specific errors. This first enhanced version is denoted with a + symbol. The process was then repeated: errors from the "+" prompt version were analysed, leading to a second, even more refined prompt, denoted as ++.

The process was initiated by selecting the three best-performing models (based on weighted F1) for each group from the previous stage: Grok 3 Mini, Gemini 2.5 Flash, and DeepSeek R1, all using the *Calculators* thinking prompt.

The impact of this self-correction technique was highly model-dependent. For DeepSeek R1, the refinement process yielded remarkable success. The first iteration, DeepSeek R1 +, achieved a Weighted F1 score of 77.4%, a significant improvement over its baseline performance. The accuracy score saw an even greater increase of more than 5 percentage points. This enhancement elevated DeepSeek R1 to become the single

Table 5.6: Comparison of Accuracy, Weighted and Macro Average F1 Scores for the enhanced *calculators* thinking prompt. Improvements and worsenings are respectively indicated with ↑ and ↓, showing the absolute percentage increase from the baseline.

Model	**Accuracy** (%)	**Weighted F1** (%)	**Macro Avg F1** (%)
Gemini 2.5 Flash	$\mathbf{73.4 \pm 2.0}$	$\mathbf{72.7 \pm 2.1}$	$\mathbf{72.7 \pm 2.2}$
Gemini 2.5 Flash +	$72.4 \pm 2.0\,(\downarrow 1.4)$	$72.1 \pm 2.0\,(\downarrow 0.8)$	$71.0 \pm 2.2\,(\downarrow 2.3)$
Gemini 2.5 Flash ++	$72.2 \pm 1.9\,(\downarrow 1.6)$	$71.7 \pm 2.0\,(\downarrow 1.4)$	$71.8 \pm 2.1\,(\downarrow 1.2)$
DeepSeek R1	73.8 ± 1.9	73.5 ± 2.0	73.7 ± 2.2
DeepSeek R1 +	$\mathbf{77.4 \pm 2.0\,(\uparrow 4.9)}$	$\mathbf{77.4 \pm 2.0\,(\uparrow 5.3)}$	$\mathbf{76.8 \pm 2.2\,(\uparrow 4.2)}$
DeepSeek R1 ++	$76.4 \pm 1.9\,(\uparrow 3.5)$	$76.5 \pm 1.9\,(\uparrow 4.5)$	$75.5 \pm 2.2\,(\uparrow 2.4)$
Grok 3 Mini	$\mathbf{74.8 \pm 1.9}$	$\mathbf{74.5 \pm 2.0}$	$\mathbf{73.9 \pm 2.1}$
Grok 3 Mini +	$74.6 \pm 2.0\,(\downarrow 0.3)$	$74.3 \pm 2.0\,(\downarrow 0.3)$	$74.1 \pm 2.1\,(\uparrow 0.3)$

best-performing model-prompt combination in the entire study. The second refinement (++), while still outperforming the baseline, showed a slight decrease in performance compared to the first, suggesting a point of diminishing returns.

In contrast, the performance of Gemini 2.5 Flash slightly degraded with each iteration. The weighted F1 dropped by 0.8 percentage points with the first enhancement and by a further 0.6 points with the second.

The results for Grok 3 Mini were largely neutral. The performance metrics for the enhanced prompts remained statistically indistinguishable from the baseline, with only marginal fluctuations well within the standard deviation. This indicates a high degree of stability in Grok 3 Mini's performance, which was not significantly influenced, either positively or negatively, by the prompt modifications.

In summary, the self-correction strategy proved to be exceptionally effective for DeepSeek R1. However, for models like Gemini 2.5 Flash and Grok 3 Mini, which already demonstrated strong baseline results, the iterative refinement did not confer any advantage.

5.2 Stage 2

5.2.1 Identifying the Best Pseudo-Labelling Approach

To determine the most reliable configuration for pseudo-labelling different combinations of LLM predictions were evaluated. As explained in Section 4.2.1, the combinations were assessed on the labelled training set of 500 posts using the first stage predictions, considering only instances where the models agreed on the label.

Three combinations were evaluated: (1) DeepSeek V3 (thinking) and Grok 3 Mini (non-thinking); (2) DeepSeek V3 (thinking), Grok 3 Mini (thinking), and Grok 3 Mini (non-thinking); and (3) all four models, adding Gemini 2.5 Flash (non-thinking) to the previous combination. Table 5.7 presents the agreement percentages, accuracies, and weighted F1 scores for these combinations.

Table 5.7: Metrics for three combination, based on agreement, of the best performing LLMs. The evaluations were taken on the 500 posts of the training set, using the same prediction as in the first stage, while keeping only the records where the models chosen agreed.

Models Combined	**Agreement** (%)	**Accuracy** (%)	**Weighted F1** (%)
DeepSeek V3 + (T) Grok 3 Mini (NT)	79.6	84.2 ± 1.8	84.1 ± 1.8
DeepSeek V3 + (T) Grok 3 Mini (T) Grok 3 Mini (NT)	74.6	86.6 ± 1.8	86.5 ± 1.8
DeepSeek V3 + (T) Grok 3 Mini (T) Grok 3 Mini (NT) Gemini 2.5 Flash (N)	68.4	88.3 ± 1.7	88.2 ± 1.7

The results reveal a clear trade-off between agreement rate and prediction quality. As more models are incorporated into the combination, the agreement percentage declines progressively from 79.6% for the two-model setup to 68.4% for the four-model ensemble, given the difficulty in achieving prediction agreement. However, this reduction is accompanied by substantial gains in performance metrics, with accuracy improving from 84.2% to 88.3% and weighted F1 score from 84.1% to 88.2%.

In addition to performance metrics, the class distributions resulting from each combination were examined to ensure they aligned with the proportions in the original labelled dataset. Table 5.8 shows the class proportions, which for each class vary with just one or two percentage points with respect to the original training set, indicating that the pseudo-labelling process preserved the original class balance effectively and did not introduce significant distributional shifts that could subsequently bias the model.

Table 5.8: Class proportions for the combined LLMs predictions of the training set. Different classes are indicated consistently with Table 3.1

Models Combined	**IN** (%)	**ID** (%)	**BR** (%)	**AT** (%)
DeepSeek V3 + (T) Grok 3 Mini (NT)	23.0	42.0	27.1	8.0
DeepSeek V3 + (T) Grok 3 Mini (T) Grok 3 Mini (NT)	23.0	42.6	26.5	8.0
DeepSeek V3 + (T) Grok 3 Mini (T) Grok 3 Mini (NT) Gemini 2.5 Flash (N)	21.1	44.0	26.9	8.5

Overall, the four-model combination was chosen due to its superior accuracy and weighted F1 scores. While the agreement rate was lower (at almost 70%), it still provided a substantial number of pseudo-labelled examples, while prioritizing high-confidence pseudo-labels.

5.2.2 RoBERTa Performance with Pseudo-Labels

The semi-supervised learning phase incorporated pseudo-labels generated from the selected LLM combinations into the training process for RoBERTa-large. The results, averaged across the 5-fold, are presented in Table 5.9. Performances were evaluated across four training data configurations: (1) the *Original labelled* dataset of 500 posts (first stage); (2) the *LLM-enhanced* dataset, which augmented the labelled data with pseudo-labels from the four-model LLM consensus; (3) *Iterative 1*, which further refined the pseudo-labels by requiring agreement between the LLM consensus and the first stage RoBERTa predictions; and (4) *Iterative 2*, which applied an additional

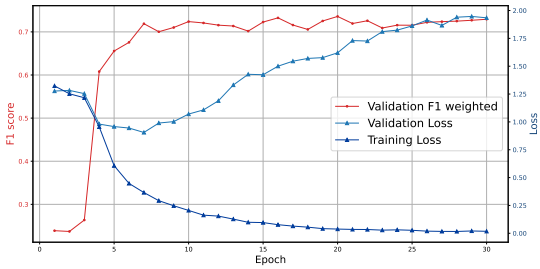
refinement using predictions from the RoBERTa model obtained from *Iterative 1*. The metrics reported correspond to the epoch achieving the highest Weighted F1 score in each fold. Improvements (absolute percentage) in Accuracy and Weighted F1 are shown relative to the *Original labelled* dataset baseline.

Table 5.9: Comparison of RoBERTa-large fine-tuning results across different training datasets augmented with pseudo-labels. In brackets, next to the fine-tune data, the number of additional pseudo-labels used is reported. Metrics are averaged across 5-fold cross-validation and correspond to the epoch with the highest Weighted F1 score. Improvements relative to the *Original labelled* dataset are indicated with ↑.

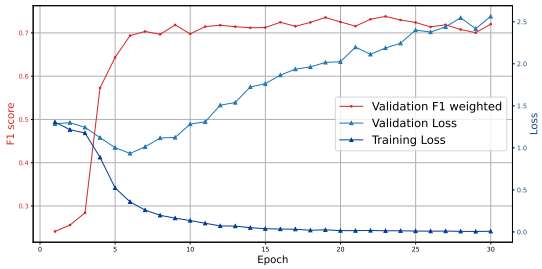
Finetune Data	**Best Epoch**	**Accuracy** (%)	**Weighted F1** (%)
Original labelled (+ 0)	77	71.8 ± 4.9	71.0 ± 5.2
LLM-enhanced (+ 1055)	20	$74.0 \pm 3.2\,(\uparrow 3.1)$	$73.6 \pm 3.3\,(\uparrow 3.7)$
Iterative 1 (+ 885)	**23**	$\mathbf{74.4 \pm 5.2}\,(\uparrow \mathbf{3.7})$	$\mathbf{73.8 \pm 5.6}\,(\uparrow \mathbf{3.9})$
Iterative 2 (+ 968)	22	$73.6 \pm 5.4\,(\uparrow 2.5)$	$72.8 \pm 5.8\,(\uparrow 3.7)$

The results clearly demonstrate the efficacy of pseudo-labelling in enhancing RoBERTa's performance. All three augmentation strategies yielded improvements over the baseline model trained solely on the original 500 labelled posts. The *LLM-enhanced* dataset provided a substantial boost, increasing the weighted F1 score by 3.7 percentage points to 73.6%. The *Iterative 1* approach, which filtered the LLM consensus labels against the baseline RoBERTa's predictions, achieved the peak performance. It delivered a final weighted F1 score of 73.8% and an accuracy of 74.4%, which is 3.7 and 3.9 percentage points respectively. Interestingly, this optimal configuration used fewer pseudo-labels (885) than the initial LLM-enhanced set (1055) while achieving the peak performance. The second refinement, *Iterative 2*, resulted in slightly lower performance than *Iterative 1*, even though almost 100 more pseudo-labels were provided to the model. The number of epochs required to reach maximum performance seemed to decrease proportionally to the number of fine-tuning examples.

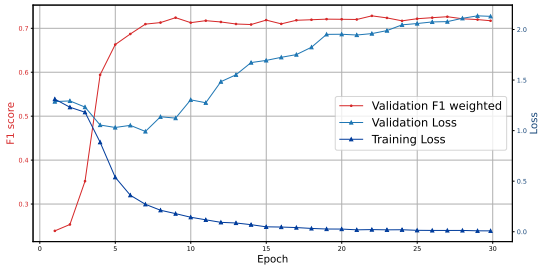
Figure 5.2 illustrates the training dynamics for RoBERTa-large across the three pseudo-labelling configurations using Cross-Entropy loss. Each subfigure displays the progression of training loss, validation loss, and validation weighted F1 score over epochs. In all cases, the training loss decreases steadily and plateaus towards the later epochs. The validation weighted F1 score rises sharply in the early epochs and reaches a plateau after fewer epochs compared to the original labelled dataset training (as



(a) LLM-enhanced



(b) Iterative Enhancing 1



(c) Iterative Enhancing 2

Figure 5.2: Training plots of RoBERTa-large fine-tuned with Cross-Entropy loss and learning rate of 5e-5 on the three pseudo-label augmented datasets.

observed in the Cross-Entropy plot in Figure 5.1). Again, the validation loss decreases initially but begins to increase after a few epochs in each configuration, but appears uncorrelated to other metrics such as weighted F1 score, which continues to improve or stabilize even as the loss increases. Accuracy curves, though not shown, exhibit trajectories almost identical to the weighted F1 scores across all plots.

5.2.3 Ensemble Model Performance

Validation performance of the ensemble model, integrating predictions from three selected high-performing components- Gemini 2.5 Flash using the non-thinking calculators prompt, DeepSeek R1 with the thinking calculators prompt enhanced once (”+”), and the RoBERTa-large model from the Iterative Enhancing 1 fine-tuning-are presented in Table 5.10, alongside its constituent models for comparison.

The ensemble outperformed each individual component across all metrics, achieving a weighted F1 score of 79.4%.

Table 5.10: Performance metrics for the ensemble model and its components.

Model	**Accuracy** (%)	**Weighted F1** (%)	**Macro Avg F1** (%)
Gemini 2.5 Flash	75.4 ± 1.9	74.4 ± 2.0	74.3 ± 2.1
DeepSeek R1	77.4 ± 2.0	77.4 ± 2.0	76.8 ± 2.2
RoBERTa-large	74.4 ± 5.2	73.8 ± 5.6	73.2 ± 5.4
Ensemble	$\mathbf{79.8 \pm 2.1}$	$\mathbf{79.4 \pm 2.0}$	$\mathbf{78.7 \pm 2.9}$

5.3 Final Evaluation on Held-Out Test Set

Final evaluations of the ensemble on the held-out test set of 100 posts yielded a Weighted F1 score of $76.9 \pm 7.7\%$, a Macro Average F1 of $80.1 \pm 6.9\%$, and an Accuracy of $77.0 \pm 7.6\%$. The errors associated with these metrics were evaluated using bootstrap resampling,. These results, while showing higher variance due to the smaller test set size, align closely with the cross-validation performance, confirming the ensemble's robustness and generalization capability.

Chapter 6

Discussion

6.1 Interpretation of Results and key Findings

This section interprets the empirical results presented in the previous chapter, highlighting key findings and their implications. Where relevant, the discussion addresses the effectiveness of different approaches and potential reasons for observed patterns.

6.1.1 RoBERTa Initial Fine-Tuning Performance

The fine-tuning experiments with RoBERTa models highlighted the advantages of larger architectures and standard loss functions. RoBERTa-large consistently outperformed RoBERTa-small, likely due to its greater capacity.

Cross-Entropy loss emerged as the most effective, surpassing Ordinal and Soft F1 losses. This suggests that treating the task as multi-class classification, rather than explicitly ordinal or F1-optimized, allows the model to learn balanced representations across imbalanced classes. The training plots revealed an unexpected outcome: validation loss often increased after early epochs, signalling potential overfitting, yet weighted F1 continued to improve alongside the other metrics such as accuracy. This suggests that validation loss is not indicative of true model performance improvements in this task. This decoupling implies that loss minimization may not align perfectly with F1 optimization in imbalanced settings such as this one.

6.1.2 Foundational Prompting with LLMs

The foundational prompting experiments revealed significant variations in LLM performance, highlighting the importance of prompt design. The consistent superiority

of the structured *Calculators* prompt suggests that structured prompts with additional guidance better align with nature of the task.

Newer, more efficient models like Grok 3 Mini and Gemini 2.5 Flash demonstrated good performance in this domain-specific classification. In contrast, the underperformance of larger models such as Grok 3 and Gemini 2.5 pro may indicate a tendency to overfit to general knowledge, making them less adaptable to the concise language of social media data.

The mixed results from implementing Chain-of-Thought (CoT) prompts demonstrate that explicit reasoning is not universally beneficial. For some models, like DeepSeek R1, CoT led to significant gains. For others that already perform well, like Gemini 2.5 Flash, it seems to introduce unnecessary complexity, undermining their inherent capabilities. This emphasizes that prompt selection must be tailored to the model. While a non-thinking *Calculators* prompt offers a robust baseline for agile LLMs, a thinking version may unlock superior performance in specific cases, with the added benefit of providing a verifiable justification for its classification.

A key implication is the potential for LLMs to serve as initial classifiers in real-world social media monitoring systems, without the need of task-specific training/fine-tuning. While the estimated uncertainties indicate stable performance, further validation on diverse datasets would be needed to confirm generalizability.

6.1.3 Iterative Prompt Refinement via Self-Correction

The model-dependent outcomes of the self-correction process suggest that a prompt's effectiveness is not universal but instead tightly related to the inherent reasoning architecture of the specific model.

The substantial performance boost in DeepSeek R1 indicates that its baseline performance was limited by identifiable and systematic errors. The initial prompt refinement successfully addressed these flaws, establishing it as the top performer. However, the diminished returns seen in the second refinement ("++") suggest that once these primary errors were fixed, further prompt additions introduced more noise than signal, leading to a performance bottleneck.

Conversely, the performance degradation of Gemini 2.5 Flash implies that its initial prompt was already near-optimal. The added instructions likely introduced counterproductive constraints, interfering with an already efficient reasoning process. Similarly, Grok 3 Mini's stability suggests a robust architecture that was largely insensitive to these

minor prompt modifications, performing consistently despite the additional details.

Overall, these findings suggest that refinement is most effective when it addresses specific, existing weaknesses. For models that are already well-aligned with a task, adding complexity to the prompt can lead to overthinking or conflicting instructions, ultimately degrading performance.

6.1.4 Semi-Supervised Learning with Pseudo-Labelling

The pseudo-labelling stage demonstrated the value of employing LLM predictions to augment limited labelled data. As hypothesised, integrating these high-quality pseudo-labels into RoBERTa-large fine-tuning yielded consistent improvements, with *Iterative 1* (requiring agreement between LLMs and initial RoBERTa) delivering the best results, suggesting that a highest confidence on the pseudo-labels is preferable to larger number of examples.

Training plots illustrated faster convergence using pseudo-labelling, attributable to the larger effective training set size introduced by the pseudo-labels, which provides more diverse examples and accelerates learning.

Overall, this findings highlight the advantages of RoBERTa-large. Despite being a model 50 to 1000 times smaller than the other general-purpose LLMs, RoBERTa still performed similarly to the best ones after introducing the semi-supervised learning approach. This is a remarkable result, especially considering how much cheaper it would be running it in a real-life setting.

6.1.5 Ensemble Model

The ensemble approach, which aggregated predictions from two high-performing LLMs (Gemini 2.5 Flash and DeepSeek R1) and the fine-tuned RoBERTa-large model from the *Iterative 1* pseudo-labelling stage, demonstrated clear advantages in enhancing classification robustness and accuracy. By employing a weighted average based on each model's validation Weighted F1-score, the ensemble obtained great performance by assigning greater influence to the more reliable predictors. The LLMs contributed broad reasoning capabilities from their general-purpose training, while the domain-adapted RoBERTa-large provided specialized insights refined through semi-supervised learning.

This consistent result on the final held-out set suggests strong generalization and reduced overfitting risk. Ultimately, this hybrid approach demonstrates that strategic

integration of compact, efficient models can rival resource-heavy alternatives, enabling cost-effective deployment for real-time social media suicide risk detection.

6.2 Comparison with Existing Literature

The results achieved in this study demonstrate competitive performance when compared to existing literature on suicide risk detection in social media, particularly within the context of the IEEE BigData 2024 Cup Challenge. The overview paper by Li et al. [30] provides comprehensive benchmarks, summarizing the strategies and outcomes of 13 finalist teams in the competition, with evaluation also focused on weighted F1-score due to class imbalance. In that challenge, top-performing teams achieved weighted F1-scores ranging from 54.9% to 76.1% on the final 100-post test set. The ensemble model of the present research, combining pseudo-labelled RoBERTa-large with Gemini 2.5 Flash and DeepSeek R1, attained a weighted F1-score of 79.4% on cross-validation and 76.9% on the held-out 30-post test set, positioning it solidly within the upper tier of competition results while using accessible, open-source components. It is important though to contextualize these results in light of the discrepancies between the original competition data and the one provided by the organizers for this research (as discussed in Paragraph 3.1). While these dataset differences might influence direct comparability to the precise leader-board rankings of the competition, the present metrics were calculated on an almost identical training set and a carefully curated test set, both maintaining the original distribution, solidifying the current findings.

A key alignment with the literature is the prevalent use of hybrid approaches integrating Bidirectional Language Models (BLMs) like BERT variants [9][27][26][45][14] with LLMs (used by 5 teams). The present methodology mirrors this trend, utilizing RoBERTa (a BLM) fine-tuned via pseudo-labelling alongside prompted LLMs, which Li et al. [30] note as a ”promising” strategy for addressing data scarcity and imbalance. For instance, competition teams commonly generated pseudo-labels for the 1500 unlabelled posts using high-confidence LLM predictions, followed by selection of quality samples via manual verification or confidence thresholds—similar to the LLM consensus filtering detailed in Section 4.2.1.

Comparisons with non-competition studies add further validity the current advancements. Early LLM applications, such as GPT-3.5 in zero/few-shot settings [46][47]), were limited by general-purpose training lacking domain specificity, as noted by Li et al. [30]. Fine-tuning domain-adapted LLMs like Mental-FLANT5 or MentalLLAMA

improved this to 75.5% weighted F1 [48], but our prompt-engineered LLMs alone (e.g., DeepSeek R1 at 77.4%) and ensemble exceed this, benefiting from recent models' enhanced reasoning (e.g., CoT prompting). Similarly, during the 2024 Workshop on Computational Linguistics and Clinical Psychology (CLPsych), participants achieved up to 90% accuracy via LLM-NLP hybrids on similar task [49], but Li et al. [30] caution against overfitting without cross-sample validation—the present bootstrapped uncertainties and held-out test alignment mitigate this risk.

Broader literature on Reddit suicidality datasets (e.g., Li et al. [31]; Shing et al. [50]; Gaur et al. [51]) reports weighted F1-scores of 65-75% for traditional ML or basic transformers, even in presence of uneven risk distributions (e.g., ¡10% for high-risk classes). The semi-supervised approach of the current research addresses this effectively, with macro F1 (80.1% on the test-set) indicating great minority class handling. However, challenges persist: like competition teams, diminishing returns were observed in iterative refinement.

In summary, this work advances prior efforts by demonstrating that cost-effective, open-source hybrids can achieve performance comparable to proprietary LLM ensembles (e.g., GPT-4 [9]), with implications for scalable deployment in resource-limited settings. Future studies could extend this to multimodal or cross-lingual data, as suggested by Li et al. [30].

Chapter 7

Conclusions

This research has explored innovative approaches to suicide risk detection in social media posts, utilizing a combination of fine-tuned transformer models and prompted LLMs to address the challenges of limited labelled data and class imbalance. By integrating semi-supervised learning techniques, such as pseudo-labelling, with ensemble methods, the study has demonstrated the potential for compact, efficient models to achieve high performance in this critical public health domain.

The key findings highlight the effectiveness of hybrid methodologies. RoBERTa-large, when fine-tuned with Cross-Entropy loss and augmented through iterative pseudo-labelling from high-confidence LLM predictions, achieved substantial improvements over baseline supervised training. Prompt engineering, particularly with structured prompts like the *Calculators* variant and Chain-of-Thought reasoning, unlocked strong capabilities in models such as DeepSeek R1 and Grok 3 Mini, often rivalling or surpassing fine-tuned alternatives. The self-correction refinement process further highlighted model-specific benefits, with notable gains for DeepSeek R1. Ultimately, the ensemble integrating RoBERTa-large with Gemini 2.5 Flash and DeepSeek R1 yielded a weighted F1-score of 79.4% on cross-validation and 76.9% on the held-out test set, positioning it competitively against benchmarks from the IEEE BigData 2024 Cup Challenge and broader literature. These results affirm that strategic use of open-source tools can enable scalable, cost-effective solutions for early intervention, particularly in resource-constrained low- and middle-income countries where suicide rates are disproportionately high.

Observations from the experiments reveal important patterns. The non-correlation of validation loss from metric improvements (e.g., weighted F1) during fine-tuning suggests that traditional overfitting indicators may not fully capture progress in imbalanced

classification tasks, emphasizing the need for metric-aligned evaluation. Additionally, the model-dependent efficacy of prompting strategies indicates that no universal approach exists; instead, specific design to architectural strengths—such as reasoning capabilities in DeepSeek R1—maximizes outcomes. The pseudo-labelling process, while effective, showed diminishing returns beyond the first iteration, likely due to increasing noise in subsequent refinements.

Despite these advancements, several unsolved problems persist in computational suicide risk detection. Data scarcity remains a fundamental barrier; even with pseudo-labelling, reliance on small annotated datasets (e.g., 500 posts) limits model robustness, particularly for rare high-risk classes like "Attempt" (comprising only 7-8% of samples). Class imbalance continues to limit generalization, as models may overemphasize prevalent categories, potentially missing subtle indicators in under-represented ones. Ethical and privacy concerns are unresolved, including the risk of re-identifying users from anonymized Reddit data or triggering false positives. Moreover, the black-box nature of RoBERTa an LLMs with no thinking processes employed complicates interpretability, making it difficult to verify reasoning in high-stakes decisions. Generalizability across platforms, languages, and cultural contexts is another gap, as the dataset is confined to English Reddit posts from the COVID-19 period, which may not capture evolving linguistic patterns or demographic variations. Finally, the absence of real-world deployment metrics leaves uncertainty about performance in dynamic, noisy social media environments.

Suggestions for further work include exploring multimodal integration, incorporating images, videos, or user metadata (e.g., posting frequency) alongside text to capture richer risk signals, potentially using models like Gemini's multimodal variants. Advanced semi-supervised techniques, such as contrastive learning or active learning for selective pseudo-label refinement, could further mitigate imbalance and noise. To enhance interpretability, future studies might incorporate some forms of attention visualization in RoBERTa and, again, structured CoT outputs from LLMs for clinical validation. Cross-lingual and cross-platform adaptation, via domain transfer learning, would broaden applicability to global contexts. Real-time deployment pilots, perhaps in collaboration with mental health organizations, could evaluate system efficacy, including intervention outcomes and user feedback. Finally, benchmarking against emerging LLMs (e.g., future Grok 4). These directions hold promise for evolving this research into practical tools that save lives.

Bibliography

[1] World Health Organization. *Tracking universal health coverage: 2023 global monitoring report*. World Health Organization, 2023.

[2] The Lancet Public Health. A public health approach to suicide prevention, 2024.

[3] Jennifer L Carey, Stephanie Carreiro, Brittany Chapman, Nathalie Nader, Peter R Chai, Sherry Pagoto, and Danielle E Jake-Schoffman. Some and self harm: The use of social media in depressed and suicidal youth. In *Proceedings of the... Annual Hawaii International Conference on System Sciences. Annual Hawaii International Conference on System Sciences*, volume 2018, page 3314, 2018.

[4] Glen Coppersmith, Ryan Leary, Patrick Crutchley, and Alex Fine. Natural language processing of social media as screening for suicide risk. *Biomedical informatics insights*, 10:1178222618792860, 2018.

[5] Shaoxiong Ji, Shirui Pan, Xue Li, Erik Cambria, Guodong Long, and Zi Huang. Suicidal ideation detection: A review of machine learning methods and applications. *IEEE Transactions on Computational Social Systems*, 8(1):214–226, 2020.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Bidirectional encoder representations from transformers. *arXiv preprint arXiv:1810.04805*, page 15, 2018.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[8] Inbar Levkovich and Mahmud Omar. Evaluating of bert-based and large language mod for suicide detection, prevention, and risk assessment: A systematic review. *Journal of Medical Systems*, 48(1):113, 2024.

[9] Jakub Pokrywka, Jeremi I Kaczmarek, and Edward J Gorzelańczyk. Evaluating transformer models for suicide risk detection on social media. In *2024 IEEE International Conference on Big Data (BigData)*, pages 8566–8573. IEEE, 2024.

[10] Evandro J. S. Diniz, José E. Fontenele, Adonias C. de Oliveira, Victor H. Bastos, Silmar Teixeira, Ricardo L. Rabêlo, Dario B. Calçada, Renato M. dos Santos, Ana K. de Oliveira, and Ariel S. Teles. Boamente: A natural language processing-based digital phenotyping tool for smart monitoring of suicidal ideation. *Healthcare*, 10(4), 2022. ISSN 2227-9032. doi: 10.3390/healthcare10040698.

[11] Panchanit Boonyarat, Di Jie Liew, and Yung-Chun Chang. Leveraging enhanced bert models for detecting suicidal ideation in thai social media content amidst covid-19. *Information Processing & Management*, 61(4):103706, 2024. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2024.103706.

[12] Xuhai Xu, Bingsheng Yao, Yuanzhe Dong, Saadia Gabriel, Hong Yu, James Hendler, Marzyeh Ghassemi, Anind K Dey, and Dakuo Wang. Mental-llm: Leveraging large language models for mental health prediction via online text data. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(1):1–32, 2024.

[13] Avinash Patil, Siru Tao, and Amardeep Gedhu. Evaluating reasoning llms for suicide screening with the columbia-suicide severity rating scale. *arXiv preprint arXiv:2505.13480*, 2025.

[14] Stefan Pasch and Jannic Cutura. Text classification with limited training data: Suicide risk detection on social media. In *2024 IEEE International Conference on Big Data (BigData)*, pages 8560–8565. IEEE, 2024.

[15] Thomas H McCoy and Roy H Perlis. Reasoning language models for more transparent prediction of suicide risk. *BMJ Mental Health*, 28(1), 2025.

[16] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837, 2022.

[17] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

[18] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

[19] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12):1–38, 2023.

[20] Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers. In *The eleventh international conference on learning representations*, 2022.

[21] Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *The Twelfth International Conference on Learning Representations*, 2023.

[22] Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. Promptbreeder: Self-referential self-improvement via prompt evolution. *arXiv preprint arXiv:2309.16797*, 2023.

[23] Archiki Prasad, Peter Hase, Xiang Zhou, and Mohit Bansal. Grips: Gradient-free, edit-based instruction search for prompting large language models. *arXiv preprint arXiv:2203.07281*, 2022.

[24] Masahiro Suzuki. Mental level prediction using long sentence discriminators and language generation models. In *2024 IEEE International Conference on Big Data (BigData)*, pages 8574–8580. IEEE, 2024.

[25] Jingyun Bi, Wenxi Zhu, Jingyun He, Xinshen Zhang, and Chong Xian. Large model fine-tuning for suicide risk detection using iterative dual-llm few-shot learning with adaptive prompt refinement for dataset expansion. In *2024 IEEE International Conference on Big Data (BigData)*, pages 8520–8526. IEEE, 2024.

[26] Max Lovitt, Haotian Ma, Song Wang, and Yifan Peng. Suicide risk assessment on social media with semi-supervised learning. In *2024 IEEE International Conference on Big Data (BigData)*, pages 8541–8549. IEEE, 2024.

[27] Vy Nguyen and Chau Pham. Leveraging large language models for suicide detection on social media with limited labels. In *2024 IEEE International Conference on Big Data (BigData)*, pages 8550–8559. IEEE, 2024.

[28] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896. Atlanta, 2013.

[29] Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *2020 International joint conference on neural networks (IJCNN)*, pages 1–8. IEEE, 2020.

[30] Jun Li, Yifei Yan, Ziyan Zhang, Xiangmeng Wang, Hong Va Leong, Nancy Xiaonan Yu, and Qing Li. Overview of ieee bigdata 2024 cup challenges: Suicide ideation detection on social media. In *2024 IEEE International Conference on Big Data (BigData)*, pages 8532–8540. IEEE, 2024.

[31] Jun Li, Xinhong Chen, Zehang Lin, Kaiqi Yang, Hong Va Leong, Nancy Xiaonan Yu, and Qing Li. Suicide risk level prediction and suicide trigger detection: A benchmark dataset. *HKIE Transactions Hong Kong Institution of Engineers*, 29 (4):268–282, 2022.

[32] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[33] Google. Gemini 2.5: Our most intelligent ai model. `https://blog.google/technology/google-deepmind/gemini-model-thinking-update` March 2025. Accessed: June 23, 2025.

[34] Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*, 2024.

[35] Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Huazuo Gao, Jiashi Li, Liyue Zhang, Panpan Huang, Shangyan Zhou, Shirong Ma, et al. Insights into

deepseek-v3: Scaling challenges and reflections on hardware for ai architectures. *arXiv preprint arXiv:2505.09343*, 2025.

[36] OpenCV. Grok-3 - most advanced ai model from xai. `https://opencv.org/blog/grok-3/`, February 2025. Accessed: June 23, 2025.

[37] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

[38] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[39] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[40] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521 (7553):436–444, 2015.

[41] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[42] Hanlin Zhang, Depen Morwani, Nikhil Vyas, Jingfeng Wu, Difan Zou, Udaya Ghai, Dean Foster, and Sham Kakade. How does critical batch size scale in pre-training? *arXiv preprint arXiv:2410.21676*, 2024.

[43] Xuying Li, Zhuo Li, Yuji Kosuga, and Victor Bian. Output length effect on deepseek-r1's safety in forced thinking. *arXiv preprint arXiv:2503.01923*, 2025.

[44] Andrew Zhao, Yiran Wu, Yang Yue, Tong Wu, Quentin Xu, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with zero data. *arXiv preprint arXiv:2505.03335*, 2025.

[45] Chayan Tank, Shaina Mehta, Sarthak Pol, Vinayak Katoch, Avinash Anand, Raj Jaiswal, and Rajiv Ratn Shah. Su-roberta: A semi-supervised approach to predicting suicide risk through social media using base language models. *arXiv preprint arXiv:2412.01353*, 2024.

[46] Milad Moradi, Kathrin Blagec, Florian Haberl, and Matthias Samwald. Gpt-3 models are poor few-shot learners in the biomedical domain. *arXiv preprint arXiv:2109.02555*, 2021.

[47] Bishal Lamichhane. Evaluation of chatgpt for nlp-based mental health applications. *arXiv preprint arXiv:2303.15727*, 2023.

[48] Abeed Sarker, Rui Zhang, Yanshan Wang, Yunyu Xiao, Sudeshna Das, Dalton Schutte, David Oniani, Qianqian Xie, and Hua Xu. Natural language processing for digital health in the era of large language models. *Yearbook of Medical Informatics*, 33(01):229–240, 2024.

[49] Jenny Chim, Adam Tsakalidis, Dimitris Gkoumas, Dana Atzil-Slonim, Yaakov Ophir, Ayah Zirikly, Philip Resnik, and Maria Liakata. Overview of the clpsych 2024 shared task: Leveraging large language models to identify evidence of suicidality risk in online posts. In *Proceedings of the 9th Workshop on Computational Linguistics and Clinical Psychology (CLPsych 2024)*, pages 177–190, 2024.

[50] Han-Chin Shing, Suraj Nair, Ayah Zirikly, Meir Friedenberg, Hal Daumé III, and Philip Resnik. Expert, crowdsourced, and machine assessment of suicide risk via online postings. In *Proceedings of the fifth workshop on computational linguistics and clinical psychology: from keyboard to clinic*, pages 25–36, 2018.

[51] Manas Gaur, Vamsi Aribandi, Amanuel Alambo, Ugur Kursuncu, Krishnaprasad Thirunarayan, Jonathan Beich, Jyotishman Pathak, and Amit Sheth. Characterization of time-variant and time-invariant assessment of suicidality on reddit using c-ssrs. *PloS one*, 16(5):e0250448, 2021.

Appendix A

Pseudo-code

A.1 Ordinal Loss

Here followed the pseudo-code for the ordinal loss employed for RoBERTa fine-tuning.

```
class OrdinalTrainer(Trainer):
    def compute_loss(self, model, inputs, return_outputs=False, **
        kwargs):
        labels = inputs.pop("labels").float()  # Pre-encoded
            cumulative vectors
        outputs = model(**inputs)
        logits = outputs.get('logits')
        loss_fct = nn.BCEWithLogitsLoss()
        loss = loss_fct(logits, labels)
        return (loss, outputs) if return_outputs else loss
```

Listing A.1: Pseudo code of ordinal cross-entropy loss in Python

Appendix B

LLM prompting

B.1 Base prompts

The two *Base* prompts versions (zero-Shot and CoT) are included below.

```
Based on the social media post provided, return the category that the post belongs to based on the following four categories and their definitions.
---
Description of risk categories:
- indicator: The post content has no explicit suicidal expression or has explicit expression of resolved suicidal issues.
- ideation: The post content has explicit suicidal expression but there is no plan or tendency to commit suicide.
- behavior: The post content has explicit suicidal expression and there is plan or tendency to act out self-harm or suicide, or mentioned historical experience of self-harm behaviour.
- attempt: The post content has explicit expression concerning recent suicide attempt, or mentioned historical experience of suicide attempt.
---
Note that the suicide risk only corresponds to the person writing the post not of other people potentially being mentioned.
Only answer with one word. It should be always one of the following 'indicator', 'ideation', 'behavior', 'attempt'
Never answer with something different than one of the four options. Never answer with NaN or empty answer
---
User post:
```

```
{{}}
```

Listing B.1: **Base** prompt, non-thinking version.

```
You are tasked with classifying social media posts into suicide risk categories. Think through this step-by-step before providing your final answer.

First, analyze the post using this thinking process:

<thinking>
1. CONTENT ANALYSIS: What are the key elements in this post?
   - Identify any mentions of death, dying, suicide, self-harm
   - Note emotional indicators (hopelessness, despair, etc.)
   - Look for specific plans, methods, or preparations
   - Check for references to past attempts

2. EXPLICIT SUICIDE MENTIONS:
   - Is suicide explicitly mentioned? (Yes/No)
   - Are there direct expressions of wanting to die or not be alive? (Yes/No)

3. PLAN OR METHOD IDENTIFICATION:
   - Are specific methods mentioned? (tools, locations, substances)
   - Are there signs of preparation? (notes, acquiring means)

4. TEMPORAL CONTEXT:
   - Does this refer to past events or current/future intentions?
   - Are past attempts specifically described?

5. CATEGORY REASONING: Based on the above analysis, which category fits best?
</thinking>

Now classify the post into one of these categories:

---
Description of risk categories:
- indicator: The post content has no explicit suicidal expression or has explicit expression of resolved suicidal issues.
- ideation: The post content has explicit suicidal expression but there is no plan or tendency to commit suicide.
- behavior: The post content has explicit suicidal expression and
```

```
   there is plan or tendency to act out self-harm or suicide, or
   mentioned historical experience of self-harm behaviour.
- attempt: The post content has explicit expression concerning
   recent suicide attempt, or mentioned historical experience of
   suicide attempt.

---
IMPORTANT RULES:
- The suicide risk only corresponds to the person writing the post,
   not other people potentially being mentioned
- Only answer with one word: 'indicator', 'ideation', 'behavior', or
    'attempt'
- Never answer with anything different than one of the four options
- Never answer with NaN or empty answer
- Show your thinking process, then provide your final classification

---
User Post: {{}}

Your response format:
<thinking>
[Your step-by-step analysis here]
</thinking>

Final Classification: [your answer]
```

Listing B.2: *Base* prompt, thinking version.

B.2 Calculators prompts

Here are the two *Calculators* prompts: one without thinking and one using CoT.

```
Based on the social media post provided, return the category that
   the post belongs to based on the following four categories and
   their definitions.
---
 Indicator:
Definition: The post or content lacks any explicit expression
   concerning suicide. There may be general expressions of distress
   or sadness, but no direct mention or indication of suicidal
   thoughts or feelings. May include also a vague feeling about
   dying or death but no explicit mentioning of suicide. Note that
```

```
    the mentioning of suicide need to be very explicit in order to
    not be 'indicator'. Very large general dissatisfaction about life
     and hopefullness is still 'indicator' when suicide/killing
    oneself is not mentioned. When in doubt whether the suicide was
    explicitly mentioned or not choose 'indicator' over 'ideation'
Ideation:
Definition: The content includes explicit expressions of suicidal
    thoughts or feelings, i.e. a desire to not be alive any longer,
    but without any specific plan to commit suicide. This can range
    from vague thoughts about not wanting to live to stronger, more
    explicit desires to end ones life (albeit without a specific plan
     how to do so). If no desire to die or to commit suicide is
    expressed, consider it 'indicator'. If the post contains specific
     ideas of how to commit suicide (ways/locations/means/methods)
    consider it 'behavior'.
Behavior:
Definition: The post includes explicit expressions of suicidal
    thoughts and also mentions (even if only briefly) a plan to
    commit suicide or exhibits self-harming behavior. It must include
     some form of explicit planning like a specific method or
    preperations taken (e.g. suicide note, lethal medication/drugs,
    tools/weapons (e.g. knifes/guns/ropes) suitable to end ones life,
     suitable locations (e.g. bridges/cliffs/buildings to jump off
    from, train lines to get run over by). If you are unsure whether
    there is a explicit plan or not choose 'ideation' over 'behavior'
Attempt:
Definition: The content describes past attempts at suicide. This
    category is focused on historic actions rather than current plans
    . A concrete action need to have happened in the past (e.g.
    overdose). When someone merely thought of an attempt in the past
    this classifies not as an 'attempt'
Note that when a post refers to past attempts but also mentions
    current plans then label it as 'attempt'
---
Note that the suicide risk only corresponds to the person writing
    the post not of other people potentially being mentioned.
Only answer with one word. It should be always one of the following
     'indicator', 'ideation', 'behavior', 'attempt'
Never answer with something different than one of the four options.
    Never answer with NaN or empty answer
---
User Post:
```

```
{{}}
```

Listing B.3: *Calculators* prompt, non-thinking version.

```
You are tasked with classifying social media posts into suicide risk categories. Think through this step-by-step before providing your final answer.

First, analyze the post using this thinking process:

<thinking>
1. CONTENT ANALYSIS: What are the key elements in this post?
   - Identify any mentions of death, dying, suicide, self-harm
   - Note emotional indicators (hopelessness, despair, etc.)
   - Look for specific plans, methods, or preparations
   - Check for references to past attempts
2. EXPLICIT SUICIDE MENTIONS:
   - Is suicide explicitly mentioned? (Yes/No)
   - Are there direct expressions of wanting to die or not be alive? (Yes/No)
   - If unclear, lean toward 'indicator' over 'ideation'
3. PLAN OR METHOD IDENTIFICATION:
   - Are specific methods mentioned? (tools, locations, substances)
   - Are there signs of preparation? (notes, acquiring means)
   - If uncertain about explicit planning, choose 'ideation' over 'behavior'
4. TEMPORAL CONTEXT:
   - Does this refer to past events or current/future intentions?
   - Are past attempts specifically described?
5. CATEGORY REASONING: Based on the above analysis, which category fits best?

</thinking>
Now classify the post into one of these categories:
**Indicator:**
Definition: The post or content lacks any explicit expression concerning suicide. There may be general expressions of distress or sadness, but no direct mention or indication of suicidal thoughts or feelings. May include also a vague feeling about dying or death but no explicit mentioning of suicide. Note that the mentioning of suicide need to be very explicit in order to not be 'indicator'. Very large general dissatisfaction about life and hopelessness is still 'indicator' when suicide/killing
```

```
    oneself is not mentioned. When in doubt whether the suicide was
    explicitly mentioned or not choose 'indicator' over 'ideation'.
28
29 **Ideation:**
30 Definition: The content includes explicit expressions of suicidal
    thoughts or feelings, i.e. a desire to not be alive any longer,
    but without any specific plan to commit suicide. This can range
    from vague thoughts about not wanting to live to stronger, more
    explicit desires to end ones life (albeit without a specific plan
     how to do so). If no desire to die or to commit suicide is
    expressed, consider it 'indicator'. If the post contains specific
     ideas of how to commit suicide (ways/locations/means/methods)
    consider it 'behavior'.
31
32 **Behavior:**
33 Definition: The post includes explicit expressions of suicidal
    thoughts and also mentions (even if only briefly) a plan to
    commit suicide or exhibits self-harming behavior. It must include
     some form of explicit planning like a specific method or
    preparations taken (e.g. suicide note, lethal medication/drugs,
    tools/weapons (e.g. knives/guns/ropes) suitable to end ones life,
     suitable locations (e.g. bridges/cliffs/buildings to jump off
    from, train lines to get run over by). If you are unsure whether
    there is an explicit plan or not choose 'ideation' over 'behavior
    '.
34
35 **Attempt:**
36 Definition: The content describes past attempts at suicide. This
    category is focused on historic actions rather than current plans
    . A concrete action need to have happened in the past (e.g.
    overdose). When someone merely thought of an attempt in the past
    this classifies not as an 'attempt'. Note that when a post refers
     to past attempts but also mentions current plans then label it
    as 'attempt'.
37
38 ---
39 IMPORTANT RULES:
40 - The suicide risk only corresponds to the person writing the post,
    not other people potentially being mentioned
41 - Only answer with one word: 'indicator', 'ideation', 'behavior', or
     'attempt'
42 - Never answer with anything different than one of the four options
```

```
- Never answer with NaN or empty answer

---
User Post: {{}}

Your response format:
<thinking>
[Your step-by-step analysis here]
</thinking>

Final Classification: [your answer]
```

Listing B.4: *Calculators* prompt, thinking version.

B.3 Enhanced prompts

Below are reported the prompts obtained from the first iteration of the self enhancement process performed on Gemini 2.5 Flash, DeepSeek R1 and Grok 3 Mini.

```
You are tasked with classifying social media posts into suicide risk
    categories. Think through this step-by-step before providing
   your final answer.

First, analyze the post using this thinking process:

<thinking>
1. CONTENT ANALYSIS: What are the key elements in this post?
    - Identify any mentions of death, dying, suicide, self-harm, or
       specific methods/means.
    - Note emotional indicators (hopelessness, despair, anhedonia,
       being a burden, etc.).
    - Look for specific plans, methods (tools, locations, substances
       ), or preparations (notes, acquiring means).
    - Check for references to past actions, including those
       initiated but not completed.
    - Note the imminence of any stated intentions (e.g., "tonight,"
       "soon").
    - Assess the writer's stated desire: Is there a genuine wish to
       die, or are they expressing distress about *unwanted*
       suicidal thoughts?

2. EXPLICIT SUICIDE/DEATH INTENT:
```

```
    - Does the post explicitly mention "suicide," "kill myself," or
      similar direct phrases? (Yes/No)
    - Does the post express a clear desire to die or not be alive (e
      .g., "I want to be gone," "I want it to be over," "I'm going
      to die soon" in a context of despair)? (Yes/No)
    - If suicidal thoughts are mentioned, does the writer explicitly
       state they *do not* want these thoughts or *do not* want to
      act on them (e.g., "my brain says X, but I don't want to")?
      This might lean towards 'Indicator' if the primary sentiment
      is distress about the thoughts themselves, rather than a
      desire to die.
    - If suicide/death is not explicitly mentioned or the desire is
      clearly negated or unwanted, lean towards 'Indicator'.

3. PLAN, METHOD, OR PREPARATION IDENTIFICATION:
    - Is a specific method, tool, substance, or location mentioned
      or strongly implied in relation to a suicidal act? (e.g.,
      pills, gun, bridge, "overdose," "jump").
    - Are there signs of preparation (e.g., "wrote a note," "got the
       pills," "saying goodbye")?
    - If there's a desire to die but no clear indication of plan/
      method/preparation, it's likely 'Ideation'. If there is *any*
       mention of a method or preparation, it leans towards '
      Behavior'.

4. TEMPORAL CONTEXT & ACTION ASSESSMENT:
    - Does the post describe a past event, a current/future
      intention, or both?
    - Crucially, does it describe a *specific, potentially lethal
      action* that was initiated in the past or very recently, even
       if it was stopped, interrupted, or failed? (e.g., "I took
      pills but woke up," "I tried to [method] but stopped myself/
      was found"). This is key for 'Attempt'.
    - If a past action is described, was it a concrete suicidal act
      or just thinking about one?

5. CATEGORY REASONING: Based on the above analysis, which category
   fits best?
    - Distinguish 'Indicator' (general distress, or unwanted
      thoughts about suicide without desire) from 'Ideation' (
      desire to die, explicit suicidal thoughts).
```

```
    - Distinguish 'Ideation' (desire/thoughts, no plan) from '
      Behavior' (desire/thoughts *with* a plan, method, or
      preparation).
    - Distinguish 'Behavior' (current/future plan) from 'Attempt' (a
       past or very recent *initiated action* that was potentially
      lethal).
    - If a past attempt is mentioned alongside current ideation/
      behavior, 'Attempt' takes precedence if the past action was
      concrete and potentially lethal.

</thinking>
Now classify the post into one of these categories:

**Indicator:**
Definition: The post expresses general distress, sadness,
   hopelessness, or discusses suicide abstractly, but **lacks a
   clear, current, personal expression of desire to die or suicidal
   intent from the author.** It may include mentions of suicidal
   thoughts if these are explicitly stated as unwanted and not
   representing a current personal desire to die (e.g., "my brain
   tells me to kill myself, but I don't actually want to die").
   Vague feelings about dying or death, or very general
   dissatisfaction with life, fall here if suicidal intent isn't
   explicit. **If in doubt between Indicator and Ideation due to
   ambiguity about genuine suicidal desire, choose Indicator.**

**Ideation:**
Definition: The content includes **explicit personal expressions of
   suicidal thoughts or feelings, or a desire to not be alive any
   longer,** but without any specific plan, method, or preparations
   mentioned for carrying it out. This can range from phrases like "
   I want to die," "I wish I wasn't here," "I'm going to end it all
   ," or "I'm going to die soon" (when stated in a context of
   despair), to more direct thoughts of suicide. **If no desire to
   die or commit suicide is personally expressed by the author for
   themselves, consider 'Indicator'. If a plan/method is present,
   consider 'Behavior'.**

**Behavior:**
Definition: The post includes explicit personal expressions of
   suicidal thoughts/desire to die **AND also mentions (even if only
    briefly) a plan to commit suicide, a specific method, or
```

```
   preparations being taken.** This must include some form of
   explicit planning element like a specific method (e.g., "thinking
    of overdosing on X pills," "want to jump from Y"), tools/weapons
    (e.g., "have a gun," "getting a rope"), suitable locations (e.g
   ., "going to the bridge"), or preparations (e.g., "writing my
   suicide note," "giving things away"). **If you are unsure whether
    there is an explicit plan/method, choose 'Ideation' over '
   Behavior'.**

**Attempt:**
Definition: The content describes **past or very recent, specific,
   self-initiated actions by the author that were potentially lethal
   , even if interrupted by self or others, or ultimately failed to
   cause death** (e.g., "I took all my pills last night but my
   friend found me," "I tried to hang myself today but I stopped
   right before passing out," "when I overdosed two years ago"). A
   concrete, potentially life-threatening action by the author must
   have been *initiated*. This category is focused on these historic
    or very recent actions. If a post refers to such past attempts
   and also mentions current plans/ideation, it should still be
   labeled 'Attempt' if the past action is clearly described. Merely
    *thinking* of an attempt in the past is 'Ideation' or 'Behavior'
    (if plan was involved then).

---
IMPORTANT RULES:
- The suicide risk only corresponds to the person writing the post,
   not other people potentially being mentioned.
- Only answer with one word: 'indicator', 'ideation', 'behavior', or
    'attempt'.
- Never answer with anything different than one of the four options.
- Never answer with NaN or empty answer.

---
User Post: {{}}

Your response format:
<thinking>
[Your step-by-step analysis here]
</thinking>

```

```
Final Classification: [your answer]
```

Listing B.5: **Calculators** prompt, thinking-version, self corrected by Gemini 2.5 Flash.

```
You are tasked with classifying social media posts into suicide risk
    categories. Analyze posts systematically using this framework:

<thinking>
1. CONTENT DECONSTRUCTION:
   - Identify explicit suicide references: "suicide", "kill myself",
       "end my life", "not want to live"
   - Note indirect death references: "not wake up", "be gone", "
      painless way to die"
   - Flag emotional markers: hopelessness, worthlessness, unbearable
       pain
   - Detect specific methods: pills, hanging, jumping, firearms
   - Identify preparations: acquiring means, writing notes, scouting
       locations
   - Recognize past actions: "tried to", "attempted", "overdosed"

2. INTENT ANALYSIS (CRITICAL):
   - Active endorsement: Does writer claim ownership of thoughts? ("
      I want to die")
   - Passive experience: Are thoughts presented as intrusive? ("my
      brain tells me")
   - Rejection noted? Explicit disavowal? ("but I won't do it")

3. TEMPORAL FRAMEWORK:
   - Past focus: Historical descriptions (e.g., "last week I tried")
   - Present intent: Current plans (e.g., "doing it tonight")
   - Future ideation: Non-specific timing (e.g., "someday")

4. CATEGORY DECISION TREE:
   Start here
   a) Any past suicide attempt described?     ATTEMPT
   b) Explicit suicide intent + (specific method/preparation OR
      immediate timeline)     BEHAVIOR
   c) Explicit suicide desire without plan/immediacy     IDEATION
   d) ONLY distress/metaphors/rejected thoughts     INDICATOR
</thinking>

Category Definitions with Key Clarifications:

```

```
**Indicator** (Distress without actionable suicide intent)
- General unhappiness, hopelessness, or despair
- Metaphorical death references ("want to disappear")
- Rejected/intrusive thoughts ("brain tells me to kill myself but I don't want to")
- NO expressions of active suicidal desire
- *Clarification*: "I'm going to die soon" without explicit suicide context indicator

**Ideation** (Active suicidal desire without actionable plan)
- Explicit wish to die: "want to die", "don't deserve to live"
- Non-actionable methods: "painless way" (without specification)
- No preparation or immediate timeline
- *Key boundary*: Becomes BEHAVIOR if paired with "tonight" or "now"

**Behavior** (Intent + Capability)
- Specific method stated: pills, cutting, jumping location
- Preparation evidence: bought pills, wrote note
- Imminent timeline: "tonight", "right now", "this weekend"
- *Critical expansion*: Expressions of immediate intent ("end it tonight") qualify even without method details

**Attempt** (Past concrete action)
- Description of actual self-harm with suicidal intent
- Includes interrupted/aborted attempts ("put belt around neck but stopped")
- Overrules concurrent ideation/behavior when mentioned
- *Clarification*: "tried to" or "attempted" always attempt

---
IRREDUCIBLE RULES:
1. Writer-focused: Ignore mentions of others' actions
2. Temporal hierarchy: Attempt > Behavior > Ideation
3. Immediacy trumps: "tonight" beats non-specific ideation
4. Rejection downgrade: "I don't actually want to" indicator
5. Output exactly one: 'indicator', 'ideation', 'behavior', or 'attempt'

---
User Post: {{}}

Response Format:
```

```
<thinking>
[Apply framework steps 1-4]
</thinking>
Final Classification: [single category]
```

Listing B.6: **Calculators** prompt, thinking-version, self corrected by DeepSeek R1.

```
You are tasked with classifying social media posts into suicide risk
    categories: Indicator, Ideation, Behavior, or Attempt. Analyze
   the post step-by-step using the thinking process below before
   providing your final answer.

#### Thinking Process
<thinking>
1. *Content Analysis*: Identify the key elements in the post:
   - Look for mentions of death, dying, suicide, self-harm, or
     specific methods/means (e.g., pills, guns, jumping).
   - Note emotional indicators (e.g., hopelessness, despair,
     loneliness, worthlessness).
   - Identify any specific plans (e.g., timing like "tonight"),
     methods (e.g., tools, substances, locations), or preparations
     (e.g., writing a note, acquiring means).
   - Check for references to past actions (e.g., "I tried," "I took
     pills"), including those initiated but not completed.
   - Assess the imminence of intentions (e.g., "tonight," "soon").
   - Evaluate the writers stated desire: Is it a genuine wish to
     die, or distress about unwanted suicidal thoughts?
   - *Key Check*: Look for any indication of past attempts or
     ongoing actions (e.g., " I ve  taken something").

2. *Explicit Suicide/Death Intent*:
   - Does the post explicitly mention "suicide," "kill myself," or
     similar phrases? (Yes/No)
   - Does it express a clear, personal desire to die or not be alive
      (e.g., "I want to die," "I want to be gone," " I m  going to
     die soon" in a despairing context)? (Yes/No)
   - *Clarification*: Even if "suicide" isnt mentioned, phrases
     like "I want to die" or "I dont want to be alive" count as
     explicit intent unless clearly negated (e.g., "I dont mean
     it").
   - If suicidal thoughts are present but the writer explicitly
     states they do not want to act on them (e.g., "my brain says
     kill myself, but I dont want to"), lean toward *Indicator*.
```

```
   - If intent is vague or absent, lean toward *Indicator*.

3. *Plan, Method, or Preparation Identification*:
   - Is a specific method (e.g., "overdose," "jump off a bridge"), tool, substance, or location mentioned or implied?
   - Are there signs of preparation (e.g., "I got the pills," "wrote a note")?
   - Is there a clear intention to act imminently (e.g., "tonight," "soon"), even without a method?
   - *Rule: If theres a desire to die but no plan, method, or imminent intent, its **Ideation. If any of these elements are present, its **Behavior*.

4. *Temporal Context & Action Assessment*:
   - Does the post describe a past event, current/future intent, or both?
   - Does it detail a specific, potentially lethal action initiated by the author in the past or very recently (e.g., "I took pills last night," "I tried to hang myself")?
   - *Key Check: If it implies an action is *currently happening or just happened (e.g., "I havent taken enough"), classify as *Attempt*.
   - If past actions are mentioned, were they concrete (e.g., "I overdosed") or just thoughts/plans?

5. *Category Reasoning*:
   - *Indicator* vs. *Ideation: Use **Indicator* for general distress or unwanted thoughts without clear intent; use *Ideation* for explicit desire to die.
   - *Ideation* vs. *Behavior: Use **Ideation* if no plan/imminence; use *Behavior* if plan, method, or imminent intent is present.
   - *Behavior* vs. *Attempt"": Use **Behavior* for plans without action; use *Attempt* for past or ongoing lethal actions.
   - If a past attempt is mentioned with current thoughts, prioritize *Attempt* if the past action was concrete.
</thinking>

#### Categories
*Indicator*:
- *Definition: The post expresses general distress, sadness, hopelessness, or mentions suicide abstractly, but **lacks a clear
```

```
   , current, personal desire to die or suicidal intent from the
   author*. Includes suicidal thoughts if explicitly unwanted (e.g.,
    "my brain says kill myself, but I  d o n t  want to"). Vague
   statements (e.g., "I hate living") stay here unless they clearly
   express a desire to die.
- *Key: If the post says "I want to die" or similar,  i t s  **
   Ideation* unless negated.

*Ideation*:
- *Definition: The post contains **explicit personal expressions of
   suicidal thoughts or a desire to not be alive* (e.g., "I want to
   die," "I wish I was gone," " I m  going to die soon" in despair),
    but *lacks specific plans, methods, preparations, or imminent
   intent*. Includes vague intent to die without "suicide"
   explicitly stated, unless negated.
- *Key: Moves to **Behavior* if a plan or imminent intent appears.

*Behavior*:
- *Definition: The post includes **explicit suicidal thoughts/desire
    to die AND mentions a plan, method, preparation, or imminent
   intent* (e.g., " I ll  jump off a bridge," " I m  doing it
   tonight"). Includes specific methods (e.g., "overdose on pills"),
    tools (e.g., "got a rope"), or imminent timing (e.g., "tonight")
    even without a method if context implies action soon.
- *Key: Moves to **Attempt* if an action has been initiated.

*Attempt*:
- *Definition: The post describes **past or very recent, specific,
   self-initiated actions by the author that were potentially lethal
   * (e.g., "I took pills last night," "I tried to overdose").
   Includes ongoing actions (e.g., "I  h a v e n t  taken enough"). Past
    attempts take precedence if concrete, even with current thoughts
   .
- *Key: Mere thoughts of past attempts are **Ideation* or *Behavior
   *.

#### Important Rules
- Classify based only on the  w r i t e r s  risk, not others mentioned.
- Answer with one word: *indicator, **ideation, **behavior, or **
   attempt*.
- Do not use anything other than these four options (no NaN, empty
   responses).
```

```
61
62 #### Response Format
63 <thinking>
64 [Step-by-step analysis]
65 </thinking>
66 Final Classification: [answer]
```

Listing B.7: **Calculators** prompt, thinking-version, self corrected by Grok Mini.