Proactive Safety Pipeline for Mental Health Crisis Detection and Intervention

Author: Kedar Vaidya, Tina Yuan, Brynn Zhang

Abstract

This project presents a two-stage transformer-based safety pipeline designed to detect and respond to mental health crises in conversational AI systems. The system combines risk classification with adaptive language model responses to provide empathetic, safety-focused interventions. Implementation includes binary and multi-class risk detection using fine-tuned DistilBERT classifiers, and parameter-efficient fine-tuning of Qwen 1.5-1.8B using LoRA for response generation. The binary classifier achieves 94% accuracy with 0.88 F1-score, while the multi-class classifier achieves 91% accuracy across four risk levels. The fine-tuned response generator produces contextually appropriate, empathetic responses with only 0.26% trainable parameters. Evaluation demonstrates the feasibility of deploying such systems in real-time mental health support contexts while highlighting important limitations and ethical considerations for responsible deployment.

---

1 Introduction

1.1 Motivation and Problem Statement

Large language models are increasingly deployed in conversational applications where users may express distress, suicidal ideation, or self-harm intent. Without proper safeguards, these systems risk either generating harmful content that reinforces negative thoughts or providing inadequate support during critical moments when users are most vulnerable. The challenge extends beyond simple content filtering to encompass nuanced understanding of mental health crisis severity and generation of appropriate therapeutic responses.

Traditional content moderation approaches typically employ binary classification (safe/unsafe) and block harmful outputs after generation. However, this reactive approach is insufficient for mental health contexts where proactive intervention and supportive engagement are paramount. Users expressing distress require not just detection of their crisis state, but thoughtful, empathetic responses calibrated to their level of need. Furthermore, the computational and data requirements for such systems must be practical enough for real-world deployment by organizations with limited resources.

1.2 Proposed Approach

This project develops a proactive safety pipeline consisting of three integrated components:

- **Risk Classification Module**: Fine-tuned DistilBERT transformer categorizes user inputs into discrete risk levels, enabling understanding of distress severity through both binary classification (safe/at-risk) and multi-class classification spanning four levels
- **Adaptive Response Generation**: Qwen 1.5-1.8B language model fine-tuned with Low-Rank Adaptation generates contextually appropriate, empathetic responses tailored to detected risk levels
- **Dynamic Routing Logic**: Directs inputs through different response strategies based on classification results, ensuring proportional and appropriate interventions

This architecture moves beyond detection alone to create an actionable safety framework where responses scale from validation and exploration for mild distress, to crisis-aware coping strategies for moderate risk, to immediate intervention with crisis resources for high-risk situations. The entire pipeline operates within practical computational constraints suitable for deployment on modest GPU infrastructure.

1.3 Novel Contributions

We aim to contribute to the intersection of natural language processing and mental health AI:

- Systematic comparison of lexicon-based and transformer-based approaches for mental health risk detection, demonstrating trade-offs between interpretability and performance
- Implementation of multi-level risk classification with four distinct categories rather than binary detection, enabling more nuanced response routing
- Application of parameter-efficient fine-tuning specifically for mental health response generation, achieving strong performance with minimal computational overhead
- Integration of detection and generation into a unified framework demonstrating how modern NLP techniques can be responsibly deployed in safety-critical contexts

While individual components build on established methods, their combination and application to proactive mental health intervention represents a novel system design addressing an urgent societal need.

---

2 Related Work and Background

Mental health crisis detection has been approached through various computational methods in recent years. Traditional approaches relied heavily on keyword matching and rule-based systems, which while interpretable, struggled with the nuanced and context-dependent nature of mental health language. Machine learning approaches have employed classical methods such as support vector machines and random forests trained on lexical, syntactic, and psychological features extracted from text. More recently, deep learning methods including recurrent neural networks,

convolutional neural networks, and transformer-based models have achieved state-of-the-art performance on suicide risk detection tasks using social media data.

Parameter-efficient fine-tuning methods have emerged as a practical solution for adapting large language models to specific domains without the computational cost of full fine-tuning. Low-Rank Adaptation (LoRA) has proven particularly effective, achieving performance comparable to full fine-tuning while training only a small fraction of parameters. This is especially relevant for mental health applications where domain adaptation is crucial but computational resources may be limited. Previous work has demonstrated LoRA's effectiveness across various NLP tasks, but its application to safety-critical mental health response generation remains underexplored.

This project builds upon these foundations by combining modern transformer-based classification with parameter-efficient response generation in an integrated pipeline. Unlike prior work focusing solely on detection, this approach addresses the complete interaction cycle from risk identification through appropriate response generation.

---

3 Data Collection and Preparation

3.1 Dataset Sources

The training data consists of two publicly available mental health conversation datasets merged for complementary coverage:

Dataset A: Human and LLM Mental Health Conversations

- Source: Kaggle (birdy654)
- Size: ~7,000 conversations
- Format: LLM prompts, context, therapeutic responses
- Characteristics: Structured instruction-following examples

Dataset B: MentalChat16K

- Source: Hugging Face (ShenLab)
- Size: ~16,000 conversations
- Format: Context-response pairs
- Characteristics: Diverse mental health topics including depression, anxiety, trauma, relationships

3.2 Preprocessing Pipeline

The preprocessing pipeline standardized heterogeneous dataset formats into a unified structure:

1. **Column Standardization**: Renamed columns to establish consistent naming

    - *instruction*: System-level guidance for the model
    - *input:* User messages representing concerns/questions
    - *output:* Expected counselor responses
2. **Instruction Assignment**: Applied generic instruction to Dataset A:

   "You are a helpful mental health counselling assistant. Please provide a safe, empathetic, and supportive answer to the user's input."
3. **Merging**: Concatenated both datasets
4. **Deduplication**: Removed exact matches on input-output pairs
5. **Shuffling**: Randomized dataset order (random_state=42)
6. **Export**: Saved as JSONL format for training compatibility

3.3 Dataset Statistics

- Total example (post-deduplication): ~23,000
- Average input length: 85 tokens (σ=47)
- Average output length: 120 tokens (σ=68)

Topic Distribution:

- Depression and low mood: 32%
- Anxiety and stress: 28%
- Relationship issues: 18%
- Self-esteem and identity: 12%
- Trauma and grief: 10%

---

4 Risk Detection System

4.1 Labeling Strategy Development

Creating training labels for mental health risk classification required developing two complementary approaches to generate risk labels from unlabeled data.

4.1.1 Binary Lexicon-Based Labeling

The initial approach employed regular expression pattern matching against predefined categories:

Risk Categories and Patterns:

Category	Example Patterns
Suicide intent	“kill myself”, “end my life”, “suicide”, “want to die”
Self-harm	“cut myself”, “hurt myself”, “self harm”
Violence	“kill him”, “hurt someone”, “hurt her”
Crisis signals	“no point in living”, “nothing matters”, “give up”

Labeling Logic: Case-insensitive matching assigns label 1 if any pattern matches, label 0 otherwise

Advantages:

- Computationally efficient
- Highly interpretable
- Domain expert validation possible

Limitations:

- No gradation of distress (binary only)
- Misses implicit signals and metaphorical language
- Vulnerable to evasion through paraphrasing
- Highly imbalanced: ~95% safe, ~5% at-risk

4.1.2 Multi-Level Weighted Lexicon

Enhanced labeling scheme with four discrete risk levels:

Level	Label	Indicators	Weight
0	Safe	No risk indicators	0
1	Mild distress	“empty”, “worthless”, “hopeless”, “overwhelmed”, “numb”	1
2	Moderate risk	“wish I could disappear”, “better if I was gone”, “can’t go on”	2
3	High risk	“kill myself”, “end my life”, “suicide”, “take my life”	3

Scoring Function: Cumulative weighted score determines final risk level

- Score $\geq 3 \rightarrow$ Level 3
- Score $= 2 \rightarrow$ Level 2

- Score = 1 → Level 1
- Score = 0 → Level 0

Resulting Label Distribution:

- Level 0 (Safe): 78%
- Level 1 (Mild): 14%
- Level 2 (Moderate): 6%
- Level 3 (High): 2%

4.2 Binary Classification Model

4.2.1 Model Architecture and Training

Base Model: DistilBERT-base-uncased

- Architecture: Distilled BERT
- Transformer layers: 6
- Hidden dimensions: 768
- Attention heads: 12
- Parameters: ~66M
- Performance retention: 97% of BERT
- Speed improvement: 60% faster

Classification Head:

- Dropout layer (p=0.1)
- Linear transformation: 768 → 2 classes
- Loss function: Cross-entropy

Training Configuration:

- Optimizer: AdamW
- Learning rate: 2e-5
- Warmup steps: 100
- Epochs: 3
- Batch size: 16
- Max sequence length: 256 tokens
- Train/eval split: 90/10 (stratified)
- Training time: 45 minutes (T4 GPU)
- Compute cost: 0.75 GPU-hours

4.2.2 Evaluation Results

Overall Performance:

- Accuracy: 0.94
- Precision: 0.89

- Recall: 0.87
- F1-Score: 0.88

Confidence Distribution:

- Average confidence (safe inputs): 0.96
- Average confidence (risky inputs): 0.91
- Predictions in uncertain range (0.4-0.6): <5%

Error Analysis:

Error Type	Common Patterns	Example
False Negative	Implicit distress without explicit keywords	"I just want to sleep forever"
False Negative	Metaphorical expressions	"Everything feels gray and pointless"
False Positive	Third-person discussions	"My friend is talking about suicide"
False Positive	Academic/educational context	Discussing suicide prevention strategies

4.3 Multi-Class Classification Model

4.3.1 Model Architecture and Training

Architecture identical to binary classifier with modified output layer:

Key Modifications:

- Classification head: 768 → 4 classes (Levels 0-3)
- Loss function: Categorical cross-entropy
- Evaluation metrics: Weighted averaging across classes

Training Configuration: Same as binary classifier

- Training time: 50 minutes (T4 GPU)
- Compute cost: ~0.8 GPU-hours

4.3.2 Evaluation Results

Overall Performance:

- Accuracy: 0.91
- Weighted Precision: 0.88
- Weighted Recall: 0.85
- Weighted F1-Score: 0.86

Per-Class Performance:

Risk Level	Precision	Recall	F1-Score	Support
0 (Safe)	0.94	0.96	0.95	1,794
1 (Mild)	0.82	0.78	0.80	322
2 (Moderate)	0.75	0.70	0.72	138
3 (High)	0.88	0.83	0.85	46

Key Observations:

- For level 0 (safe) inputs, the model achieved exceptional performance with 94% precision, 96% recall, and 0.95 F1-score across 1,794 evaluation examples. This strong performance on the majority class confirms that the system rarely misclassifies safe inputs as risky, minimizing unnecessary user concern or intervention.
- For level 1 (mild distress), performance remained solid with 82% precision, 78% recall, and 0.80 F1-score across 322 examples. The slight performance decrease compared to level 0 shows the inherent ambiguity in distinguishing mild distress from safe expressions of normal negative emotions.
- Level 2 (moderate risk) proved more challenging, achieving 75% precision, 70% recall, and 0.72 F1-score across 138 examples. This performance decrease is because of reduced training data and the difficulty of the classification boundary between moderate and both mild and high risk. Moderate risk represents an inherently ambiguous middle ground where context and nuance become paramount.
- Level 3 (high risk) performed surprisingly well given its rarity, achieving 88% precision, 83% recall, and 0.85 F1-score across only 46 evaluation examples. This strong performance shows the relatively clear linguistic markers of severe crisis intent, which manifest through explicit crisis vocabulary that distinguishes level 3 inputs from other categories.
- Most errors occurred between adjacent risk levels, with level 1 inputs sometimes classified as level 0 or level 2, rather than jumping directly to level 3. This suggests the model has learned meaningful gradations of risk rather than arbitrary boundaries. The most common confusion involved level 1 and level 2 inputs, where the distinction between mild and moderate distress requires subtle contextual understanding that proves challenging even for human annotators.

---

5 Response Generation System

5.1 Base Model Selection and Rationale

Selected Model: Qwen 1.5-1.8B

Characteristic	Specification	Rationale

Parameters	1.8 billion	Balance of capability and feasibility
Architecture	Transformer decoder	Standard causal LM architecture
Attention	Grouped-query attention	Reduced memory bandwidth
Activation	SwiGLU	Improved expressiveness
Position encoding	Rotary embeddings	Better positional information
Vocabulary	~151,000 tokens (BPE)	Good coverage of English + mental health terms
Inference latency	2-4 seconds (200 tokens, T4)	Practical for real-time conversation

5.2 Parameter-Efficient Fine-Tuning with LoRA

5.2.1 LoRA Configuration and Theory

Low-Rank Adaptation Principle: For pretrained weight matrix W (d × k), inject trainable decomposition $\Delta W = BA$ where B (d × r) and A (r × k) with rank $r << \min(d, k)$

Configuration Parameters:

Parameter	Value	Purpose
Rank (r)	16	Dimensionality of adaptation
Alpha	32	Scaling factor (α/r controls magnitude)
Dropout	0.05	Regularization
Target modules	q_proj, k_proj, v_proj, o_proj	Attention projections
Task type	CAUSAL_LM	Language generation

Parameter Efficiency:

- Total model parameters: 1.8B
- Trainable parameters (LoRA): 4.7M
- Trainable percentage: 0.26%

Advantages:

- Dramatically reduced memory requirements
- Faster training iterations
- Easy model versioning (store only adapters)
- Reduced risk of catastrophic forgetting

5.2.2 Training Data Formatting

Prompt Structure:

[instruction]

User: [input]

Example Formatted Prompt:

"You are a helpful mental health counselling assistant.

Please provide a safe, empathetic, and supportive answer to the user's input."

User: "I feel hopeless and nothing seems to matter anymore."

Assistant: "I hear that you're experiencing feelings of hopelessness right now. Those feelings are valid, and I want you to know that you're not alone in experiencing them. When everything feels overwhelming, it can be hard to see beyond the present moment. Would you like to talk more about what's been contributing to these feelings?"

Tokenization Details:

- Max sequence length: 1024 tokens
- Truncation: Enabled
- Padding: To max length
- Loss computation: Full sequence

5.2.3 Training Configuration and Process

Training Parameters:

Parameter	Value	Rationale
Per-device batch size	1	Memory constraints
Gradient accumulation steps	16	Effective batch size = 16
Learning rate	2e-4	Higher than full fine-tuning
Epochs	1	~1,400 gradient updates
Optimizer	AdamW (8-bit paged)	Memory efficiency
Precision	FP16 mixed precision	2× speed with minimal quality impact
Checkpoint frequency	Every 300 steps	3 checkpoints total

Quantization Strategy:

Component	Method	Impact
Base weights	4-bit NF4	75% memory reduction
Quantization constants	Double quantization	Further compression

Compute dtype	FP16	Balance of speed and stability

Training Resources:

- Hardware: Single NVIDIA T4 GPU (16GB)
- Training time: ~3 hours
- Compute cost: 3 GPU-hours (~$2-3 on cloud)
- Peak memory usage: ~12GB

5.3 Inference Pipeline

5.3.1 Model Loading and Configuration

Loading Process:

1. Load base Qwen 1.5-1.8B with 4-bit quantization
2. Load LoRA adapters via PEFT library
3. Merge adapters with base attention projections
4. Set to evaluation mode (dropout disabled)

Generation Parameters:

Parameter	Value	Purpose
Max new tokens	200	Substantial but not excessive length
Sampling method	Nucleus (top-p)	Controlled randomness
Top-p	0.9	Balance creativity and coherence
Temperature	0.7	Sharpness of probability distribution
Random seed	Fixed (for evaluation)	Deterministic outputs

5.3.2 Generation Quality and Characteristics

Observed Response Characteristics:

- Tone: Consistently empathetic and validating
- Opening patterns: Acknowledgment of user concerns
- Structure: Validation → exploration/support → open-ended questions
- Average length: 120 tokens
- Coherence: Multi-sentence responses with clear topic flow

Common Positive Patterns:

- “I hear that you’re going through a difficult time”
- “Your feelings are completely valid”
- Appropriate questions to continue dialogue

- Avoidance of toxic positivity or dismissiveness

Observed Limitations:

- Occasional phrase repetition across responses
- Frequent use of "I hear you" as opener
- Limited diversity in conversation starters
- Sometimes insufficient specificity for complex concerns

---

6 Integrated System Architecture and Evaluation

6.1 System Architecture and Routing Logic

Pipeline Flow:

User Input

↓

[Risk Classifier]

↓

├─ Level 0 (Safe) ————→ [Standard LLM Response]

├─ Level 1 (Mild) ————→ [Empathetic Support]

├─ Level 2 (Moderate) ——→ [Crisis-Aware Response]

└─ Level 3 (High) ————→ [Immediate Intervention + Resources]

↓

[Response Delivery]

Response Strategies by Risk Level:

Risk Level	Strategy	Key Elements
Level 0 (Safe)	Standard counseling	Active listening, validation, exploration
Level 1 (Mild)	Enhanced validation	Normalization of feelings, gentle exploration, rapport building
Level 2 (Moderate)	Crisis-aware support	Concrete coping strategies, psychoeducation, gentle help-seeking suggestions
Level 3 (High)	Immediate intervention	Strong validation, crisis resources (988, Crisis Text Line), safety emphasis

6.2 Example System Outputs

Example 1: Level 3 (High Risk):

- **Input**: "I feel like nothing matters and I want to disappear. I've been thinking about ending it all."
- **Classification**: Level 3, confidence 0.96
- **Response**: "I hear that you're in tremendous pain right now, and I'm deeply concerned about what you're sharing with me. Please know that even though it doesn't feel this way right now, your life has value and meaning. These feelings, while overwhelming, are not permanent. I need you to know that help is available right now, today: You can call or text 988 (the Suicide and Crisis Lifeline) to speak with trained counselors 24/7. You can also text HOME to 741741 (Crisis Text Line) for immediate text-based support. If you're in immediate danger, please call 911 or go to your nearest emergency room. Would you be willing to reach out to one of these services right now? You deserve support, and there are people ready to help you through this moment."

Example 2: Level 2 (Moderate Risk):

- **Input**: "I can't take this anymore. Everything feels hopeless and I don't see the point in trying."
- **Classification**: Level 2, confidence 0.78
- **Response**: "I can hear how overwhelmed and exhausted you're feeling right now. When in the middle of intense pain, it can feel like there's no way forward, and those feelings are real and valid. These feelings, while very difficult, can shift with time and support. Have you been able to talk with anyone about how you've been feeling? Sometimes having professional support can make a significant difference when struggling this deeply. In the meantime, are there any small things that have helped you feel even slightly better in the past, even temporarily? Sometimes taking things moment by moment, rather than trying to see the whole path forward, can be a little easier."

Example 3: Level 1 (Mild Distress):

- **Input**: "I've been feeling really overwhelmed with work lately and it's affecting my mood."
- **Classification**: Level 1, confidence 0.89
- **Response**: "It sounds like you're carrying a heavy load right now, and work stress can really take a toll on overall wellbeing and mood. It's actually very common for workplace pressure to affect how one feels day-to-day. Can you tell me more about what specifically feels most overwhelming? Is it the volume of work, particular pressures or deadlines, relationships at work, or something else? Understanding what's contributing most to the stress can sometimes help think about where small changes might make a difference. How have you been taking care of yourself outside of work during this stressful period?"

Example 4: Level 0 (Safe):

- **Input**: "I've been working on improving my self-confidence. What are some strategies that might help?"
- **Classification:** Level 0, confidence 0.94
- **Response**: "That's a wonderful goal to work toward, and it shows real self-awareness. Building self-confidence is definitely something that can develop over time with practice. Some strategies that people find helpful include: practicing self-compassion by talking to yourself the way you'd talk to a good friend, celebrating small wins and acknowledging efforts rather than just outcomes, identifying and challenging negative self-talk patterns when you notice them, and setting small, achievable goals that let you build momentum. It can also be valuable to reflect on strengths—what are some things you already feel confident about? Sometimes patterns from areas where confidence already exists can apply to areas for growth."

6.3 Quantitative Evaluation

6.3.1 Classification Performance Summary

Binary vs Multi-Class Comparison:

Classifier	Accuracy	Precision	Recall	F1-Score
Binary	0.94	0.89	0.87	0.88
Multi-Class	0.91	0.88 (weighted)	0.85 (weighted)	0.86 (weighted)

Multi-Class Per-Level Performance:

Risk Level	F1-Score	Primary Strength	Primary Challenge
0 (Safe)	0.95	Clear linguistic markers	Occasional mild distress missed
1 (Mild)	0.80	Reasonable boundary detection	Ambiguity with Level 0 and 2
2 (Moderate)	0.72	Acceptable given complexity	Limited training data, boundary ambiguity
3 (High)	0.85	Strong despite rarity	Very limited examples (n=46)

6.3.2 Generation Quality Assessment

Evaluation Methodology:

- Sample size: 50 inputs per risk level (200 total)
- Evaluators: 3 independent raters
- Inter-rater reliability: Cohen's $\kappa = 0.67$ (substantial agreement)
- Rating scale: 5-point Likert (1=poor, 5=excellent)

Aggregated Results Across All Risk Levels:

Dimension	Mean Score	Interpretation
Response Relevance	4.2/5.0	Generally addresses inputs appropriately
Empathy Tone	4.5/5.0	Successfully learned validating tone
Safety Compliance	4.8/5.0	Very few harmful or dismissive responses
Coherence	4.1/5.0	Logical structure, occasional repetition

Performance by Risk Level:

Risk Level	Relevance	Empathy	Safety	Coherence	Notable Pattern
Level 0	4.4	4.3	4.7	4.3	Highest coherence
Level 1	4.2	4.6	4.8	4.2	Balanced across dimensions
Level 2	4.1	4.5	4.9	4.0	Strong safety compliance
Level 3	4.0	4.6	5.0	3.8	Perfect safety, lower coherence

6.4 Qualitative Analysis

6.4.1 Strengths and Effective Patterns

Classification Strengths:

- Explicit crisis detection: Reliably detects “want to die”, “kill myself”, “hurt myself”
- Generalization capability: Recognizes paraphrases and varied expressions
- Semantic understanding: Goes beyond exact keyword matching
- Confidence calibration: High confidence scores align with actual accuracy

Generation Strengths:

- Consistent empathy: Validation language across all risk levels
- Logical structure: Validation → exploration → suggestions/questions
- Safety compliance: Zero harmful responses in 200-sample evaluation
- Crisis resource provision: 100% inclusion for Level 3, appropriate format
- Tone calibration: Successfully matches intervention intensity to risk

6.4.2 Limitations and Failure Modes

Classification Limitations:

Limitation	Description	Example
Implicit signals	Misses metaphorical distress	“I just want to sleep forever” → Level 0

Cultural variation	Western-centric patterns	Somatic or spiritual expressions missed
Third-person framing	Detects keywords, misses context	“My friend talks about suicide” → Level 3
Metaphorical language	Literal interpretation	“This homework is killing me” → Level 2

Generation Limitations:

Limitation Type	Description	Impact
Repetitive patterns	Overuse of opening phrases	“I hear you”, “That must be difficult”
Insufficient length	Brief responses for complex concerns	Inadequate exploration of multifaceted issues
Generic resources	US-focused crisis information	Non-US users receive less useful information
No personalization	Same resources for all contexts	Lacks demographic/geographic adaptation
Context limitation	No memory across turns	Cannot track escalation or prior discussion

6.4.3 Error Analysis

False Negative Patterns (Missed Risk):

Pattern Category	Example	Reason for Misclassification
Implicit ideation	“Sometimes I wonder if everyone would be better off if I wasn’t here”	Future tense, indirect expression
Resignation language	“I’m done trying”, “There’s no point anymore”	Lacks explicit crisis keywords
Metaphorical expression	“I want to disappear into nothingness”	Figurative rather than literal interpretation
Third-person testing	“What if someone wanted to end their life?”	Distancing mechanism not recognized

False Positive Patterns (Over-detected Risk):

Pattern Category	Example	Reason for Misclassification

Discussing others	“My friend told me she wants to hurt herself”	Third-person framing not recognized
Educational context	Discussing suicide prevention strategies	Academic discussion flagged as personal
Metaphorical frustration	“This situation is killing me”	Dramatic language taken literally
Historical reference	“I used to feel suicidal”	Past tense not adequately weighted

Generation Error Patterns:

Error Type	Frequency	Example Scenario
Insufficient specificity	~15%	Multi-faceted concern receives generic response
Template reliance	~12%	Exact phrasing repeated across similar inputs
Coherence issues	~8%	Slight redundancy or awkward transitions
Missed complexity	~10%	Addresses one aspect while ignoring others

---

7 Discussion

7.1 Comparison of Approaches

7.1.1 Binary versus Multi-Class Classification

The choice between binary and multi-class classification is a fundamental trade-off between simplicity and nuance. Binary classification achieves higher overall accuracy (94% vs 91%) and simpler interpretation. An input is either safe or requires intervention. This approach minimizes system complexity, reduces potential points of failure, and provides clear decision boundaries for routing logic. The higher recall (87%) proves valuable for safety-critical applications where missing genuine crises carries severe consequences.

However, multi-class classification enables more sophisticated and appropriate responses despite modest accuracy decrease. By distinguishing four risk levels, the system can calibrate intervention intensity to user need. Users expressing mild stress receive supportive validation without alarming crisis-level intervention. Users in moderate distress receive more active coping strategies and gentle help-seeking suggestions. Only users showing high crisis markers receive the most intensive intervention with immediate resources. This proportional response approach likely improves user experience and engagement, as over-intervention may cause users to disengage or distrust the system.

For deployment, the optimal choice depends on use case and risk tolerance. Systems prioritizing absolute safety and willing to accept higher false positive rates should employ binary classification. Systems prioritizing user experience and nuanced interaction where

over-intervention risks losing user trust should employ multi-class classification. A hybrid approach using binary classification as initial filter followed by multi-class refinement on flagged inputs could combine benefits of both approaches.7.1.2 Lexicon versus Transformer Classification

Comparative Analysis:

Dimension	Lexicon-Based	Transformer-Based
Interpretability	Complete transparency	Limited (attention visualization possible)
Development time	Rapid (expert-driven)	Longer (requires training data and compute)
Maintenance	Easy updates, no retraining	Requires retraining for updates
Implicit signals	Poor	Strong
Context understanding	Absent	Good
Metaphorical language	Fails	Handles reasonably
Novel expressions	Misses	Generalizes
Inference latency	<1ms	10-50ms
Training requirements	None	0.75 GPU-hours
Adversarial robustness	Vulnerable to evasion	More robust

Practical Deployment Strategy:

Layer	Method	Purpose
Layer 1: Pre-filter	Lexicon	Fast screening of obvious cases
Layer 2: Primary model	Transformer	Nuanced contextual understanding
Layer 3: Explanation	Lexicon	Provide interpretable justification for flags
Layer 4: Expert review	Human	High-uncertainty cases

7.1.3 Full Fine-Tuning versus LoRA

Resource Comparison:

Resource	Full Fine-Tuning	LoRA Fine-Tuning
Trainable parameters	1.8B (100%)	4.7M (0.26%)
GPU requirement	4-8× T4 or equivalent	Single T4

Training time	12-24 hours	3 hours
Memory requirement	>40GB (model parallelism needed)	12GB
Storage per model	~7GB	~20MB (adapters only)
Training cost (cloud)	$50-100	$2-3

Performance Trade-offs:

Aspect	Full Fine-Tuning	LoRA
Adaptation capacity	Maximum	95-100% of full fine-tuning
Catastrophic forgetting risk	Higher	Lower (base frozen)
Domain shift handling	Better for dramatic shifts	Best for moderate adaptation
General capabilities	May degrade	Preserved
Iteration speed	Slow	Fast

Recommendation for Approach:

Scenario	Recommended Approach
Limited compute resources	LoRA
Multiple model variants needed	LoRA
Preserving general capabilities critical	LoRA
Maximum performance at any cost	Full fine-tuning
Domain dramatically different from pretraining	Full fine-tuning
Production deployment (most cases)	LoRA

7.2 Computational Requirements and Scalability

7.2.1 Training Costs

Complete System Training:

Component	Time	Hardware	Compute	Cloud Cost
Risk classifier	45 min	Single T4	0.75 GPU-hr	$0.50
Response generator (LoRA)	3 hours	Single T4	3 GPU-hr	$2.00
Total	3.75 hours	Single T4	3.75 GPU-hr	$2.50

The complete system training process required modest computational resources accessible to most researchers and small organizations. Risk classifier training consumed approximately 0.75 GPU-hours on NVIDIA T4 (45 minutes wall clock time), with peak memory usage around 8GB.

Response generator training with LoRA required approximately 3 GPU-hours (3 hours wall clock), with peak memory usage around 12GB thanks to 4-bit quantization. Total training cost on cloud platforms like Google Colab Pro would amount to $2-3, making the approach highly accessible.

Scaling with Dataset Size:

Dataset Size	Estimated Training Time	Memory Required
23K (current)	3.75 hours	12GB
50K examples	8 hours	12GB
100K examples	15 hours	14GB
250K examples	36 hours	16GB (may need multiple GPUs)

The costs scale linearly with dataset size up to memory limits. Training on 100K examples instead of 23K would require roughly 4x training time but remain feasible on single GPU. Multiple model variations for hyperparameter search or ablation studies can be trained in parallel across multiple GPUs, with full experimentation completing in 1-2 days on modest hardware.

Data Annotation Costs:

Task	Rate	Speed	Cost per 10K Examples
Mental health conversation labeling	$25/hour	~100/hour	$2,500-3,000
Quality control (20% sample)	$30/hour	~150/hour	$400-500
Total	-	-	~$3,000

Data collection and annotation represent greater costs than model training. While we leveraged existing labeled datasets, organizations applying this approach to proprietary data would need to invest in annotation. Assuming $25/hour for trained annotators labeling at ~100 examples/hour with quality control, annotating 10,000 examples would cost approximately $2,500-3,000. However, active learning and semi-supervised approaches could substantially reduce annotation requirements.

7.2.2 Inference Costs and Latency

Per-Request Performance:

Operation	Latency (CPU)	Latency (GPU)	Throughput
Classification	50ms	10ms	20/sec (CPU), 100/sec (GPU)
Generation (200 tokens)	N/A (impractical)	2-4 seconds	10-15 concurrent (single T4)
Total pipeline	N/A	2-4.5 seconds	10-15 users/T4

Classification inference proves lightweight, requiring 50ms on CPU or 10ms on GPU for a single input. This supports both real-time interactive applications and batch processing of large

volumes. A single CPU core can handle ~20 classifications per second, while a single GPU can process 100+ per second, making classification highly scalable.

Optimization Potential:

Optimization Technique	Expected Improvement	Implementation Effort
INT8/INT4 quantization	30-50% latency reduction	Medium
TensorRT compilation	30-40% latency reduction	Medium
ONNX Runtime	20-30% latency reduction	Low
Speculative decoding	40-60% perceived latency reduction	High
Batched inference	2-3× throughput	Low-Medium

More aggressive quantization (INT8 or even INT4 for inference) could reduce latency by 30-50% with minimal quality impact. Compilation optimizations through TensorRT or ONNX Runtime could achieve similar speedups. Speculative decoding techniques that draft multiple tokens in parallel before verification could substantially reduce perceived latency. Batched inference serving multiple requests simultaneously could improve GPU utilization, particularly during off-peak periods.

Production Cost Estimation (10,000 conversations/day):

Resource	Quantity	Unit Cost	Daily Cost
Classification (CPU)	500 CPU-seconds	$0.0002/sec	$0.10
Generation (GPU)	10-15 T4-hours	$0.50/hour	$5.00-7.50
Total	-	-	$5.10-7.60

Cost per conversation: $0.0005-0.0008

Monthly cost (300K conversations): $150-230

Serving 10,000 conversations per day (each requiring one classification and one generation) would require approximately: 500 CPU-seconds for classification ($0.10 on cloud CPUs) and 10-15 GPU-hours for generation ($5-8 on T4 pricing). Total daily cost of ~$8 translates to $0.0008 per conversation, making the approach economically viable.

7.3 Interpretability and Trust

7.3.1 Model Transparency

Explainability Mechanisms:

Mechanism	Availability	Utility	Implementation Status
Risk level labels	Yes	High - clear categorization	Implemented

Confidence scores	Yes	High - uncertainty quantification	Implemented
Matched keywords	Partial	High - interpretable justification	Lexicon only
Attention visualization	Possible	Medium - shows focus areas	Not implemented
LIME/SHAP explanations	Possible	Medium - feature importance	Not implemented
Generation rationale	No	High - why this response	Not feasible with current LLMs

Transparency Levels by Component:

Component	Transparency Level	Key Challenge
Lexicon labeling	Complete	None - fully interpretable
Transformer classification	Moderate	Black-box predictions
LoRA generation	Low	Opaque decision-making
Routing logic	Complete	Rule-based, fully transparent

7.3.2 Error Handling and Confidence Calibration

Confidence-Based Routing Strategy:

Confidence Range	Action	Rationale
< 0.5	Flag for human review	High uncertainty
0.5 - 0.7	Use classification with caution flag	Moderate uncertainty
0.7 - 0.9	Standard automated response	Normal confidence
> 0.9	High-confidence automated response	Very confident prediction

Our classifiers provide confidence scores that could be used for uncertainty-based routing. For instance, inputs with confidence below 0.7 might trigger human review rather than fully automated response. Confidence calibration, ensuring predicted probabilities match actual accuracy, would improve utility of these scores for decision-making.

Threshold Tuning Options:

Application Type	Recommended Threshold	Priority

Maximum safety (crisis hotline)	0.3	Maximize recall (minimize false negatives)
Balanced (general chatbot)	0.5	Balance precision and recall
Reduced intervention (low-risk context)	0.7	Maximize precision (minimize false positives)

Currently, we use fixed thresholds (0.5 for binary classification) to make decisions. Adaptive thresholding based on deployment context could improve performance. Conservative applications might use 0.3 threshold to maximize sensitivity, flagging more inputs as potentially risky. Less conservative applications might use 0.7 threshold to reduce false positives. Evaluation across threshold values and comparison of resulting precision-recall curves would inform optimal threshold selection.

Post-Generation Safety Screening (proposed):

Check	Purpose	Implementation
Harmful content detection	Catch reinforcement of negative thoughts	Train separate safety classifier
Resource inclusion verification	Ensure crisis resources present when needed	Rule-based checker
Tone appropriateness	Verify empathetic, non-dismissive language	Separate classifier or rubric

The generation component lacks explicit confidence scores, instead relying on implicit quality reflected in output. Post-generation safety classifiers could screen responses for harmful content before delivery, providing an additional safety layer. Such classifiers could flag responses that minimize user concerns, provide specific harmful information, or fail to include appropriate crisis resources when needed.

7.4 Ethical Considerations and Responsible Deployment

7.4.1 Safety and Liability

Deploying AI systems for mental health support raises a lot of ethical questions about safety and responsibility. False negatives, where the system fails to identify genuine crisis situations, could result in severe harm if users receive inadequate support during critical moments. False positives, while less directly harmful, could erode user trust and cause distress by treating normal challenges as crises. System failures, whether from technical errors or adversarial inputs, could produce harmful outputs despite our best efforts to prevent them.

These risks require careful consideration of liability and accountability. Organizations deploying such systems must clearly communicate limitations to users, making explicit that the system provides supplementary support rather than professional mental health care. Disclaimers should emphasize that the system cannot guarantee accuracy and users experiencing serious distress should seek professional help. Terms of service must address potential system failures and limit

liability appropriately. Disclaimers alone is insufficient. Organizations have affirmative obligations to build systems that minimize foreseeable harms. This requires ongoing monitoring of system performance, rapid response to identified failures, and continuous improvement based on real-world outcomes. Incident response protocols should address how the organization will handle cases where system failures contribute to harmful outcomes.

There is not one single group bearing responsibility for system failures. Developers who create the technology, organizations that deploy it, and individuals who use it all share responsibility in different ways. Clear contracts and communication between these parties help ensure appropriate accountability. Ultimately, regulatory frameworks may be needed to establish standards of care for mental health AI systems.

7.4.2 Privacy and Data Handling

Mental health conversations involve deeply personal and sensitive information requiring robust privacy protections. Users sharing distress, trauma, or crisis situations have reasonable expectations that their disclosures remain confidential. Our system design must incorporate privacy-preserving practices at every stage.

During training, we used only publicly available datasets where contributors provided informed consent for research use. However, organizations deploying similar systems with proprietary user data must obtain clear, specific consent for using conversations to improve models. Users should understand how their data will be used, stored, and protected. Options to opt out of data collection while still accessing services should be provided when feasible.

During deployment, conversations should be encrypted in transit and at rest, with access restricted to authorized personnel only. Retention policies should minimize how long data is stored, balancing improvement of services with privacy protection. Anonymization techniques should remove or obfuscate personally identifiable information before data is used for analysis or model improvement.

Special considerations apply to high-risk conversations. When users express suicidal ideation, the system must balance privacy with safety. In some jurisdictions, mandated reporting requirements may require disclosure to authorities when imminent danger is identified. Users should be informed of these limitations on confidentiality before engaging with the system. However, mandatory disclosure should be implemented thoughtfully to avoid deterring users from seeking help when they need it most.

7.4.3 Equity and Accessibility

AI systems can perpetuate or exacerbate existing inequalities in mental health care access and quality. Our training data comes primarily from English-language sources and may not adequately represent diverse cultural expressions of distress. Users from non-Western cultural backgrounds, non-native English speakers, or populations underrepresented in training data may receive lower quality service.

Addressing these disparities requires proactive efforts. Training data should be expanded to include diverse populations, languages, and cultural contexts. Partnerships with mental health organizations serving diverse communities can help source appropriate data with community

input on what constitutes culturally competent support. Evaluation should also specifically measure performance across demographic groups to identify and address disparate impact. Third, crisis resources provided by the system should reflect user context, including location, language, and cultural background.

Accessibility for users with disabilities requires attention to multiple dimensions. Visual impairments necessitate screen reader compatibility and alternative text descriptions. Cognitive disabilities may require simplified language or alternative communication modalities. Hearing impairments are less directly relevant for text-based systems but become important if voice interfaces are added. Universal design principles should guide interface development to maximize accessibility.

Economic accessibility is also a consideration. While our system design minimizes computational costs, deployment still requires resources that may limit access to well-funded organizations. Open-sourcing models, code, and approaches can democratize access to these technologies, enabling resource-constrained organizations including nonprofits and community health centers to deploy similar systems. We will need to release our code and model weights publicly to support this goal.

7.5 Limitations and Possible Future Improvements

7.5.1 Current System Limitations

Critical Limitations:

Limitation	Impact	Severity
No conversation memory	Cannot track escalation or reference prior discussion	High
No clinical validation	Unknown performance with genuine crisis language	Critical
No crisis system integration	Cannot hand off to human counselors	High
Single language (English)	Excludes majority of global population	High
Training data mismatch	Synthetic/collected vs. authentic crisis language	Moderate
Text-only modality	Misses vocal and visual distress cues	Moderate
Generic resource provision	Not personalized to user context	Moderate
No outcome tracking	Cannot measure actual impact on user wellbeing	High

7.5.2 Possible Improvements and Extensions

Several extensions could address the current limitations and enhance system capabilities. Implementing Retrieval-Augmented Generation for resource provision would enable dynamic fetching of crisis resources appropriate to user context. Building a knowledge base of global crisis hotlines, mental health services, and evidence-based self-help resources organized by geography, language, and concern type would enable tailored recommendations. The generation model would retrieve relevant resources based on classified risk level and user context.

Adding an output safety classifier to screen generated responses would provide an additional safety layer. This classifier would detect if generated text minimizes user concerns, provides potentially harmful information, or fails to include appropriate crisis resources when needed. Flagged responses could be regenerated with modified prompts or routed to human review.

Expanding multi-lingual support through translation of lexicons and fine-tuning on Spanish, Mandarin, Hindi, and other high-need language datasets would improve accessibility. Cross-lingual transfer learning approaches could enable support for lower-resource languages with limited training data. Implementing conversation memory through storage and retrieval of recent interaction history would enable contextual responses referencing prior discussion using vector databases to semantically search conversation history for relevant context.

Clinical validation studies in partnership with mental health organizations would evaluate system performance with genuine users experiencing distress, requiring Institutional Review Board approval, informed consent procedures, and careful study design to ensure participant safety. Implementation of longitudinal tracking to detect escalation patterns and trajectory changes would enable earlier intervention for users showing increasing distress over time, though such predictive capabilities raise additional ethical concerns about privacy and the potential for over-prediction leading to unnecessary intervention.

Developing personalization mechanisms that adapt communication style and strategy based on individual user preferences and response patterns could improve engagement through reinforcement learning from user feedback. Comparison of different model architectures and training approaches through carefully designed ablation studies would identify optimal configurations, including comparing LoRA with other parameter-efficient methods (prefix tuning, adapter layers), comparing different base model sizes, or comparing fine-tuning with few-shot prompting of larger models.

Integration with crisis intervention services enabling seamless handoff to human counselors for high-risk cases would create a warm transfer rather than simply providing resource information, requiring partnerships with crisis hotlines, APIs for real-time availability checking, and protocols for managing the transition from AI to human support. Implementation of real-time alerts to designated support contacts with user consent could provide safety nets for vulnerable users, though such features require extremely careful design to respect user autonomy, prevent misuse, and ensure alerts genuinely improve safety rather than creating additional risks.

Development of multimodal capabilities incorporating voice interaction, facial expression analysis, and behavioral signals would enable richer assessment and more natural interaction, though multimodal processing dramatically increases system complexity and introduces additional privacy concerns. Finally, contributing to the development of regulatory frameworks and best practices for mental health AI would help establish industry standards ensuring responsible development and deployment through participation in standard-setting organizations, publication of ethical guidelines, or engagement with policymakers developing relevant regulations.

---

8 Conclusion

This project demonstrates the technical feasibility and practical challenges of building a proactive safety pipeline for mental health crisis detection and intervention in conversational AI systems. The two-stage approach combining transformer-based risk classification with parameter-efficiently fine-tuned response generation achieves strong performance while remaining computationally practical for real-world deployment.

Performance:

Classification Performance:

- Multi-class risk classifier: 91% accuracy across four risk levels
- Binary classifier: 94% accuracy for simpler decision boundaries
- Enables nuanced routing of user inputs based on severity

Generation Quality:

- LoRA-based fine-tuning of Qwen 1.5-1.8B: 0.26% trainable parameters
- Contextually appropriate, empathetic responses across all risk levels
- Strong safety compliance: zero harmful outputs in 200-sample evaluation

System Performance:

- End-to-end latency: 2-4 seconds per request
- Deployable on single T4 GPU
- Cost: <$0.001 per conversation at scale
- Training cost: $2-3 total on cloud infrastructure

The evaluation reveals consistent empathetic tone, appropriate risk-calibrated responses, and strong safety compliance. The system successfully distinguishes between levels of distress and generates proportional interventions, from supportive validation for mild concerns to immediate crisis resource provision for high-risk situations. The parameter-efficient approach enables resource-constrained organizations to build similar systems without access to massive computational infrastructure.

Challenges include difficulties with implicit distress signals that avoid explicit crisis keywords, cultural and linguistic variation in expressions of distress, occasional repetitive patterns in generated responses, and generic rather than personalized resource provision. The system processes each input independently without conversation memory, preventing tracking of escalation patterns or personalization based on prior discussion. Most critically, the system lacks clinical validation with genuine users experiencing mental health crises.

This project systematically compares binary versus multi-class classification and lexicon versus transformer approaches for risk detection, providing practical guidance for system designers. It demonstrates the effectiveness of parameter-efficient fine-tuning specifically for mental health response generation and presents an integrated architecture showing how detection and

generation components can be combined into actionable safety frameworks. Most importantly, it highlights both the promise and limitations of AI systems for mental health support.

Potential applications include mental health chatbots, crisis hotline augmentation, social media content moderation, and educational tools for mental health awareness. The modular architecture allows organizations to adapt components to their specific needs, risk tolerance, and resources. However, successful deployment requires clear communication of system limitations, integration with existing crisis infrastructure, continuous monitoring and improvement, and partnership with mental health experts for validation and oversight.

Next Steps

By thoughtfully combining advances in natural language processing with careful attention to safety, ethics, and responsible deployment practices, systems can be built that not only avoid harm but actively contribute to user wellbeing during vulnerable moments. The path forward requires continued technical innovation, rigorous evaluation including clinical trials, meaningful engagement with mental health professionals and communities, and commitment to the principle that these powerful technologies must ultimately serve human flourishing.

This system is designed to augment, not replace, human mental health professionals. Deployment in real support contexts requires extensive additional validation, professional oversight, and commitment to continuous improvement based on real-world outcomes. The technical feasibility demonstrated here represents a necessary but insufficient step toward responsible deployment of AI systems in mental health care.

References

[1] J. Bai *et al.*, “Qwen Technical Report,” Sep. 2023, doi: https://doi.org/10.48550/arXiv.2309.16609.

[2] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, “QLoRA: Efficient Finetuning of Quantized LLMs,” May 2023, doi: https://doi.org/10.48550/arxiv.2305.14314.

[3] E. S. Hu *et al.*, “LoRA: Low-Rank Adaptation of Large Language Models.,” Oct. 2021.

[4] S. Ji, C. P. Yu, S. Fung, S. Pan, and G. Long, “Supervised Learning for Suicidal Ideation Detection in Online User Content,” *Complexity*, vol. 2018, no. 1, pp. 1–10, Sep. 2018, doi: https://doi.org/10.1155/2018/6157249.

[5] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter,” Mar. 2020, doi: https://doi.org/10.48550/arXiv.1910.01108.

[6] Shen, “MentalChat16K,” *Huggingface.co*, 2025. https://huggingface.co/datasets/ShenLab/MentalChat16K

[7] J. J. Bird, “Human and LLM Mental Health Conversations,” *Kaggle.com*, 2024. https://www.kaggle.com/datasets/birdy654/human-and-llm-mental-health-conversations

Acknowledgments

We thanks the creators of the MentalChat16K and Human and LLM Mental Health Conversations datasets for making their data publicly available for research purposes. Google Colab provided computational resources used in this project. The open-source community behind HuggingFace Transformers, PEFT, and related libraries made this work possible.

Important Note: This system is designed for research and demonstration purposes only. It has not been clinically validated and should not be deployed for real mental health support without extensive additional testing, professional oversight, and appropriate safety measures.