**Technical Report: AI Safety Models POC**

**1. Introduction**

**This Proof of Concept (POC) project was developed as part of the Solulab Machine Learning assignment to demonstrate the feasibility of applying AI Safety models for conversational AI platforms.**

**The goal was to design and integrate multiple safety-focused ML models to handle:**

* **Abuse Detection – flagging toxic, obscene, threatening, or insulting text.**
* **Escalation Recognition – detecting emotionally dangerous escalation in conversations.**
* **Crisis Detection – identifying indicators of severe emotional distress or self-harm (referred to as *SUI* for safety).**
* **Content Filtering – ensuring age-appropriate interactions (kid/teen/adult profiles).**

**The system integrates these models into a unified moderation pipeline.**

**2. High-Level Design Decisions**

* **Modular Pipelines: Each safety model was trained, validated, and saved separately (as .pkl pipelines), making the system extensible.**
* **Scikit-learn Pipelines: Used TF-IDF + Logistic Regression as baseline models for quick training, interpretability, and CPU efficiency.**
* **Real-Time Focus: Models were wrapped into inference functions that can be plugged into a chat pipeline, simulating real-time moderation.**
* **Ethical Considerations: Explicitly filtered crisis-related text under a neutral label (*SUI*) to avoid accidental triggering. Designed system for escalation to human moderators in crisis scenarios.**
* **Scalability: Chose vectorizer + linear classifiers for low-latency inference; can later be replaced with BERT/transformer-based models.**

**3. Data Sources & Preprocessing**

| **Model** | **Dataset Used** | **Preprocessing Steps** |
| --- | --- | --- |
| **Abuse Detection** | **Jigsaw Toxic Comments (Kaggle)** | **Removed HTML, lowercased text, handled missing labels, multi-label binarization** |
| **Escalation Recognition** | **DailyDialog / EmpatheticDialogues** | **Parsed dialog, converted emotion arrays, mapped to 7-class emotion labels** |
| **Crisis Detection** | **Suicide Detection Dataset (CSV provided)** | **Cleaned text, mapped labels → SUI / Non-SUI, split into train/test** |
| **Content Filtering** | **Jigsaw Toxic + Kaggle test labels (proxy)** | **Created Safe/Unsafe labels, stratified train/test split** |

**Vectorization: TF-IDF (max\_features=5000–10000, ngram\_range=(1,2)).  
Splitting: 80-20 train/test, stratified to preserve class balance.**

**4. Model Architectures & Training**

* **Abuse Detection:**
  + ***OneVsRest Logistic Regression***
  + **Multi-label classification: [toxic, severe\_toxic, obscene, threat, insult, identity\_hate]**
  + **Pipeline = TF-IDF → Logistic Regression**
* **Escalation Recognition (Emotion Classification):**
  + ***Logistic Regression (OvR)***
  + **Multi-class classification with 7 emotions**
* **Crisis Detection:**
  + ***Binary Logistic Regression***
  + **Labels: SUI / Non-SUI**
  + **Balanced class weights to handle imbalance**
* **Content Filtering:**
  + ***Binary Logistic Regression***
  + **Labels: Safe / Unsafe**
  + **Integrated with user profile (kid/teen/adult) to enforce stricter policies**

**5. Evaluation Results**

**📊 Abuse Detection**

* **Macro F1: ~0.48**
* **Best classes: Toxic, Obscene (~0.70+ F1)**
* **Challenging classes: Threat, Identity Hate (low recall due to imbalance)**

***(Insert Figure 1: Abuse Report Heatmap, Figure 2: Abuse Confusion Matrices)***

**📊 Escalation Recognition**

* **Accuracy: ~0.88**
* **Strong performance: Neutral, Joy**
* **Weak performance: Rare emotions (Fear, Disgust, Sadness had near-zero recall)**

***(Insert Figure 3: Escalation Report Heatmap, Figure 4: Escalation Confusion Matrix)***

**📊 Crisis Detection**

* **Accuracy: ~0.85**
* **Precision (SUI): ~0.78**
* **Recall (SUI): ~0.82 (prioritized recall for safety-critical tasks)**

***(Insert Figure 5: Crisis Report Heatmap, Figure 6: Crisis Confusion Matrix)***

**📊 Content Filtering**

* **Validation Accuracy: 0.93**
* **Safe: 0.96 F1**
* **Unsafe: 0.70 F1 (recall improved with class weighting)**

***(Insert Figure 7: Content Filtering Report Heatmap, Figure 8: Content Filtering Confusion Matrix)***

**6. High-Level Architecture**

**Pipeline Flow:**

**User Input → Preprocessing → [Abuse Model, Escalation Model, Crisis Model, Content Filter]**

**↓**

**Moderation Policy Layer → Final Action (Allow, Warn, Block, Escalate to Human)**

* **Modular models → independent training, reusable inference.**
* **Policy layer aggregates outputs and enforces rules depending on age group and severity.**

**7. Leadership & Iteration Strategy**

**If leading a team, I would structure iteration as follows:**

1. **Baseline First (Done): Start with TF-IDF + Logistic Regression for fast experimentation.**
2. **Error Analysis: Focus on underperforming classes (e.g., threats, rare emotions).**
3. **Data Augmentation: Use paraphrasing and back-translation to balance rare labels.**
4. **Advanced Models: Introduce fine-tuned BERT/DistilBERT for Abuse & Crisis tasks.**
5. **Human-in-the-Loop: Build dashboards where moderators review flagged cases to improve labeling.**
6. **Ethical Oversight: Ensure fairness testing (different demographics, slang, multilingual support).**

**8. Conclusion**

**This POC successfully demonstrates an end-to-end AI Safety pipeline for conversational systems.  
It integrates four distinct but complementary models into a unified moderation framework, enabling real-time detection of:**

* **Harmful or abusive language,**
* **Escalating emotional risks,**
* **Crisis situations requiring intervention,**
* **Age-appropriate content filtering.**

**The project lays the groundwork for scaling to production with advanced NLP models and moderator integration.**

**📌 Next Steps:**

* **Improve performance on minority classes (Fear, Threat, Disgust).**
* **Integrate BERT-based models for richer language understanding.**
* **Extend to multilingual moderation.**
* **Deploy as a real-time API for integration into chat platforms.**