VNU Journal of Science: Com. Science & Com. Eng., Vol. 42, No. 2 (2026) 42–63

VNU Journal of Science:
Computer Science and Communication Engineering

Journal homepage: http://www.jcsce.vnu.edu.vn/index.php/jcsce

Original Article

Enhancing Suicide Risk Classification: A Multi-Stage Framework with Sentence-Level Waterfall Architecture for Clinical Notes Analysis

Tu-Phuong Mai[1], Minh-Ha H. Le[1], Duc-Luong Tran[1], Duy-Cat Can[1,2,3], Hoang-Quynh Le[1]*

[1]*Faculty of Information Technology, VNU University of Engineering and Technology, Hanoi, 144 Xuan Thuy, Cau Giay, Hanoi, Vietnam*
[2]*Platform of Bioinformatics, Lausanne University Hospital (CHUV), Lausanne, Switzerland*
[3]*Faculty of Biology and Medicine, University of Lausanne (UNIL), Lausanne, Switzerland*

Received 19th December 2025
Revised 24th December 2025; Accepted 26th March 2026

Abstract: Suicide remains a leading cause of preventable death worldwide, yet current computational models for suicide risk assessment often struggle to filter clinically irrelevant information from electronic health records and to provide interpretable outputs that clinicians can act upon with confidence. We present a multi-stage framework with a novel sentence-level waterfall architecture that incrementally filters irrelevant and conflicting content while preserving key suicide-related indicators. This design enables direct linkage between predictions and specific textual evidence, offering transparent, sentence-level reasoning for each classification decision. At the hospital-stay level, the framework integrates sentence-level outputs through two complementary strategies, cascading inference and a generative language model, providing comprehensive assessments that reflect temporal changes and multiple clinical perspectives. Evaluation on the benchmark ScAN suicide attempt dataset demonstrates substantial gains over existing models, achieving a macro F1-score of 0.93 and particularly strong improvements in challenging categories, raising F1-scores for 'unsure' and 'negative' cases from 0.52 to 0.83. Detailed ablation and error analyses confirm that the sentence-level waterfall design is essential for both predictive accuracy and clinical interpretability, highlighting its robustness against noisy, heterogeneous EHR data.

The source code used in this study is available at https://github.com/candleMind/ESAP.

Keywords: Depression, Suicide Attempt, Mental Health, Electronic Health Records, Clinical Notes Analysis, Natural Language Processing.

* Corresponding author.
E-mail address: lhquynh@vnu.edu.vn
https://doi.org/10.25073/2588-1086/vnucsce.5615

1. Introduction

Suicide is a major global public health concern. More than 700,000 people die by suicide each year, making it the fourth leading cause of death among individuals aged 15-29 [1], with the true burden likely under-counted [2]. Crucially, many individuals who die by suicide have contact with healthcare services shortly beforehand: roughly half see a provider within one month and about 40% present to an emergency department within a year [3]. These figures underscore a pressing opportunity for timely, data-driven risk assessment within healthcare settings.

Electronic health records (EHRs) consolidate diverse, longitudinal clinical narratives authored by physicians, nurses, social workers, patients, and family members. While this breadth offers rich context for suicide risk assessment, it also introduces unique challenges: clinical notes are lengthy and heterogeneous, may contain conflicting statements (e.g., past vs. current intent), and are affected by redundancy that can degrade model performance [4]. Standard deep models struggle to attend to the entirety of long notes without sacrificing efficiency or fidelity, even with long-context architectures [5]. Equally important, clinical deployment requires transparent, evidence-linked reasoning to support clinician trust and oversight [6–8].

Prior work spans EHRs, free-text clinical notes, and social media corpora for suicide-related prediction [9–11]. Within EHRs, publicly available resources such as MIMIC-III [12] have enabled reproducible research on clinical language modeling (e.g., ClinicalBERT [13]). The ScAN dataset [14], a curated subset of MIMIC-III with expert-annotated *suicide attempt* (SA) and *suicide ideation* (SI) events, provides sentence-level supervision and stay-level labels suitable for benchmarking event detection and overall risk classification. Recent systems (e.g., ScANER [14]) leverage transformer-based retrieval and paragraph-level aggregation, yet they remain vulnerable to irrelevant or conflicting evidence and offer limited *sentence-level* interpretability. Moreover, ambiguous cases, which are common in practice, are often collapsed or under-modeled, reducing clinical usefulness. Figure 1 illustrates conflicting entries regarding suicidal behavior which obscure risk assessment, through three truncated EHR notes of a patient.

Problem formulation. We address the task of hospital-stay-level *suicide attempt risk* classification into four clinically meaningful categories: 'positive', 'unsure', 'negative', and 'neutral'. Our central thesis is that effective clinical NLP for suicide risk must (i) *filter* irrelevant or redundant text before stay-level inference, (ii) *resolve conflicts* between sources and timestamps (e.g., past vs. present attempt), and (iii) *expose* evidence at the *sentence level* to support clinical review.

Core idea. We propose a multi-stage framework centered on a **sentence-level waterfall architecture** that first filters irrelevant content, then refines suicide-related evidence into clinically meaningful categories. This design enables stay-level predictions that are both accurate and directly grounded in sentence-level evidence. By explicitly handling conflicting information and preserving uncertainty cases, the framework delivers interpretable, trustworthy assessments suitable for real-world clinical use.

Contributions. The main contributions of this study are summarized as follows:

- **Clinically grounded, interpretable architecture.** A multi-stage, sentence-level waterfall framework that links each stay-level decision to concrete sentence evidence, aligning with explainability needs in clinical deployment [6–8].
- **Conflict-aware reasoning over long, heterogeneous notes.** Post-processing and hierarchical aggregation explicitly

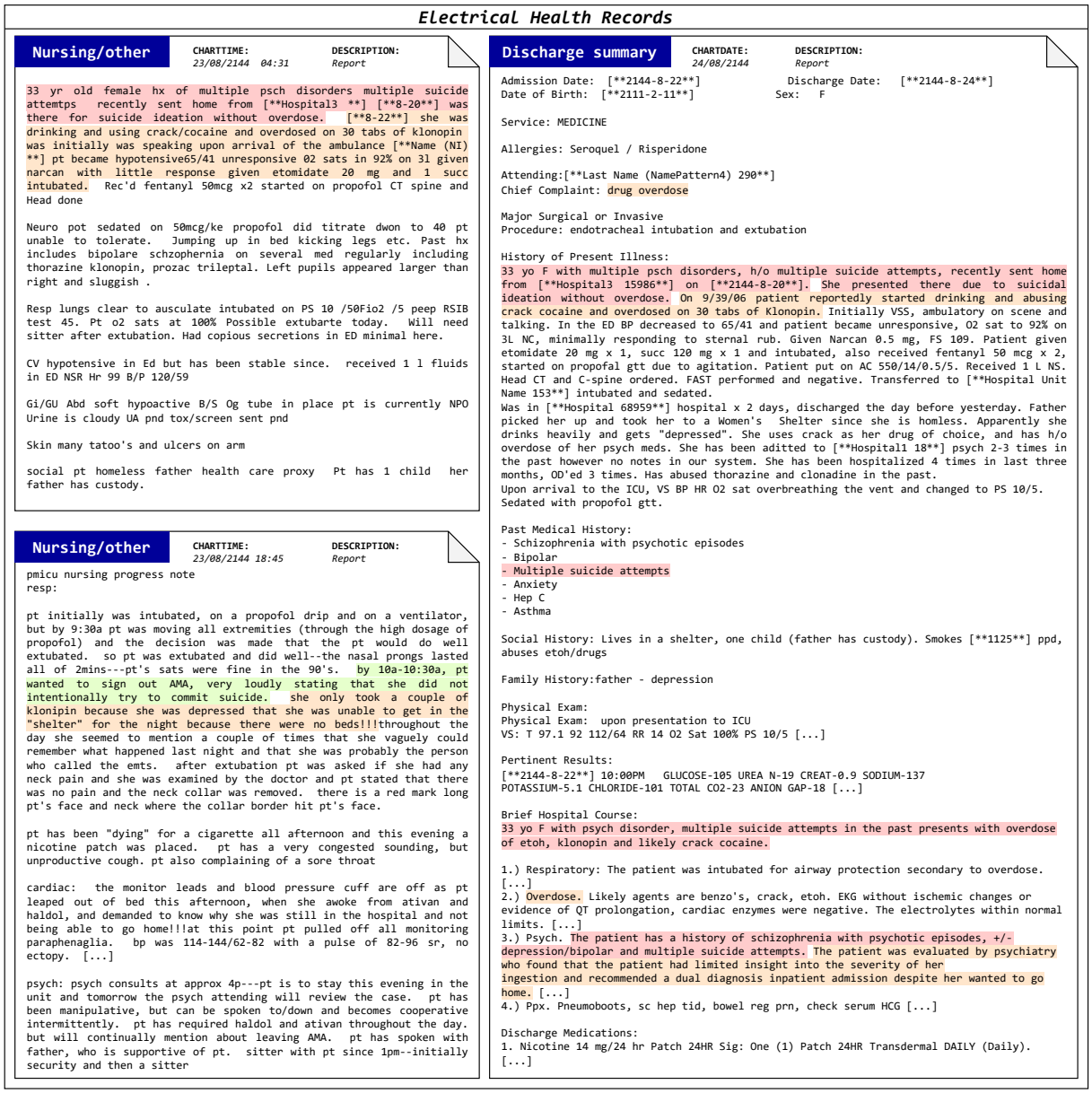

Figure 1. Example of conflicting information in clinical notes during a hospital stay of a patient who got admitted for overdose. The texts highlighted in red indicate information suggestive of suicide, orange highlights denote unclear whether the events are suicidal, while the sections in green indicate possible no suicidal intent. The MIMIC-III data is de-identified, with sensitive personal information masked and timestamps adjusted to safeguard patient privacy while preserving the relative chronological order of events. The clinical notes have been truncated for display purposes using [...] due to the very long text in the original EHRs.

address past-vs.-present statements and denials, mitigating common EHR inconsistencies [4].

- **Robust stay-level inference.** Two complementary predictors, a transparent cascading inference algorithm and a

generative language model with early-stage error handling, provide accuracy and resilience to upstream errors [15].

- **Comprehensive evaluation on a public benchmark.** On ScAN [14] (subset of MIMIC-III [12]), our approach achieves a macro F1 of 0.93 and markedly improves unsure/negative cases (F1 from 0.52 to 0.83), supported by ablations and error analyses that isolate the impact of each component.

In sum, we re-conceptualize suicide attempt risk assessment as *evidence-first*, conflict-aware inference over long clinical narratives, closing the gap between high predictive performance and the level of transparency required for real-world clinical decision support [16].

2. Related Work

Suicide risk modeling across data sources. Computational approaches have been integrated into clinical decision-making, leveraged structured EHR features (diagnoses, medications, utilization) and clinician assessments or patient self-reports to support suicide risk prediction while saving efforts [16]. Parallel lines of work used social media corpora to detect suicidal ideation from short posts [17, 18], benefiting from volume and immediacy but facing population shift and limited clinical interpretability. In contrast, clinical notes embed rich, longitudinal context recorded during care delivery. However, due to privacy and ethical considerations, many studies using clinical notes, especially research on mental health and suicide behaviors, are on private datasets from healthcare facilities [3, 19]. Public resources such as MIMIC-III [12], therefore, have catalyzed reproducible research in clinical NLP (e.g., ClinicalBERT [13]). The ScAN dataset [14] contributes expert-annotated *sentence-level* supervision for suicide attempt (SA) and suicide ideation (SI), along with stay-level labels. The problem can be formulated at various granularities, from sentence-level prediction [20] to the more applicable hospital stay-level [21], while others explored the influence of different clinical note types on suicide risk estimation [22]. Our work builds on this line by directly exploiting sentence-level supervision to make stay-level decisions that are evidence-linked and clinically reviewable.

Current methodological approaches in clinical notes. Research on suicide risk prediction from clinical notes employs a spectrum of methods, each suited to different problem formulations and granularities. Rule-based systems and traditional machine learning classifiers (e.g., logistic regression) offer high transparency by using explicit lexicons and context rules, but suffer from limited coverage and portability across diverse clinical documentation styles [11, 21]. In contrast, deep learning models, including CNNs, RNNs, and particularly Transformer-based architectures, have demonstrated high performance for detecting and predicting intentional self-harm, learning richer contextual representations from clinical narratives [10, 13, 20].

A significant methodological evolution is the move beyond simple binary classification towards modeling the clinical complexity of suicidality. Multi-label classification frameworks, powered by pre-trained language models like RoBERTa and domain-specific variants (e.g., MentalBERT [23]), are now used to identify co-occurring suicidality-related factors (SRFs), such as suicidal ideation (SI), suicide attempts (SA), and non-suicidal self-injury, within a clinical note. ScANER [14] retrieves evidence paragraphs with a transformer retriever (medRoBERTa [24]) and aggregates them via attention for multi-class severity stay-level prediction. Furthermore, temporal modeling techniques like Dynamic Topic Modeling are being applied to analyze sequences of notes, capturing fluctuations in suicide risk and

therapeutic focus over time [25]. Most recently, generative large language models (LLMs) have been explored, framing classification as a text-generation problem and showing competitive performance, though their clinical reliability and interpretability remain active areas of investigation [26].

Despite these advances, two critical problems persist. First, the sparse signal issue in long clinical notes: actionable mentions of suicide evidence are often rare and buried within lengthy, redundant documentation ("note bloat") [4]. While hierarchical models and long-context transformers manage computational load [5, 27], they still require sifting through vast amounts of non-informative text, diluting critical signals and complicating model training. Second, inadequate conflict and context handling: clinical narratives inherently contain contradictory cues, such as historical versus current ideation, patient-reported symptoms versus family history, or affirmed versus negated/denied events [28–31]. Most existing neural models, including recent transformer-based approaches, lack explicit mechanisms to resolve these conflicts, which can lead to unreliable predictions and undermine clinical trust [32]. Our approach narrows the unit of evidence to the *sentence*, adds conflict-aware post-processing, and provides deterministic aggregation rules that align with clinical review.

Interpretability and clinical trust. High-stakes deployment requires models whose decisions are understandable and auditable [6, 7, 33]. Popular post-hoc explainers (e.g., LIME, SHAP) [34, 35] provide local attributions but may be unstable and difficult to validate clinically. Attention heatmaps are accessible but can be misleading as explanations [36]. Consequently, there is a growing emphasis on intrinsically interpretable, *rationale-based* modeling, which ties predictions to human-readable evidence text spans [37, 38]. In practice, achieving trust involves several key requirements: providing transparent evidence (e.g., specific sentences that contributed to a risk score), ensuring clinical auditability through simple, understandable aggregation rules, and faithfully representing clinical uncertainty [39, 40]. Flagging 'unsure' cases, for example, is valuable for triage and clinician review, as it presents clinically relevant uncertainty [16]. Our framework operationalizes this rationale paradigm in the clinical setting by (i) formulate our task as four suicide-certainty level prediction, (ii) promoting sentences to explicit evidence, (iii) linking stay-level labels to these sentences, and (iv) enforcing simple, auditable rules (e.g., handling past-tense and denial cues) that clinicians can inspect.

Positioning and gaps addressed. In summary, the literature reveals four persistent gaps that our work addresses:

1. **Sparse-signal dilution in long notes.** Many systems ingest entire documents or coarse paragraphs, allowing irrelevant or redundant text to dominate representation learning [4, 5]. *Our remedy:* a sentence-level *waterfall* that first filters non-evidence sentences, then refines SA-related sentences before any stay-level reasoning.

2. **Limited, non-auditable explanations.** Attention or post-hoc attributions offer weak, non-faithful explanations [34, 36]. *Our remedy:* explicit, sentence-level evidences and deterministic aggregation rules that clinicians can audit.

3. **Poor handling of conflicts.** Prior models seldom enforce consistency regarding past vs. present or denial statements [14]. *Our remedy:* a conflict-aware post-processor that re-labels ambiguous SA sentences in context using domain cues [28].

4. **Under-modeling of clinical uncertainty.** Ambiguous cases are often collapsed into binary outcomes [9, 10]. *Our remedy:* a four-class formulation with targeted

Table 1. Distribution of unique suicide attempt and ideation *sentence-level annotations* in ScAN

Type	Positive	Negative	Unsure	Total
Suicide Attempt (SA)	11,080	164	2,235	13,479
Suicide Ideation (SI)	896	640	–	1,536

Table 2. Distribution of suicide attempt categories at the *hospital-stay level*

Subset	Positive	Negative	Unsure	Neutral
Training	263	29	80	945
Validation	36	4	10	126
Testing	46	6	12	199

improvements on 'unsure' and 'negative' cases, plus an error-handling training strategy that increases robustness to noisy upstream signals.

3. Materials and Methods

3.1. Materials

3.1.1. Cohort and Data Sources

Our experiments are conducted on the **Suicide Attempt and Ideation Events (ScAN)** dataset [14], which is a carefully curated subset of the MIMIC-III Clinical Database [12]. MIMIC-III contains de-identified health records of more than 40,000 patients admitted to the critical care units. The database includes demographic information, laboratory test results, medications, vital signs, and importantly for this study, free-text clinical notes from physicians, nurses, and other healthcare providers. The ScAN dataset builds upon MIMIC-III by providing expert annotations of *suicide attempt* and *suicide ideation* events at both the sentence and hospital-stay levels, enabling the development and evaluation of automated suicide risk classification methods from clinical notes.

3.1.2. Sentence-level Annotations

At the sentence level, ScAN provides granular annotations that identify whether a sentence refers to a suicide attempt or ideation, and if so, the certainty of that reference. For **suicide attempt (SA)** mentions, sentences are assigned one of three labels: 'positive SA', indicating a confirmed suicide attempt during hospitalization (often with ICD-coded method descriptors); 'unsure SA', used for ambiguous cases where intent is unclear (e.g., overdose vs. accident); and 'negative SA', marking explicit denial of a suicide attempt. For **suicide ideation (SI)** mentions, sentences are labeled as either 'positive SI', when there is an explicit mention of intent for suicide, self-harm, or hopelessness, or 'negative SI', when there is explicit evidence of the absence of such intent.

All other sentences are considered 'neutral'. The distribution of annotated sentences is reported in Table 1. Notably, the dataset is highly imbalanced, with positive SA sentences dominating and very few negative SA sentences, highlighting the difficulty of modeling rare but clinically important classes.

3.1.3. Stay-level Labels

At the hospital-stay level, ScAN assigns one of four categories: 'positive', indicating an explicit suicide attempt documented (e.g., a diagnosis of suicide attempt or intentional self-harm/overdose); 'unsure', when evidence is conflicting or intent remains indeterminate after review; 'negative', when there is clear evidence that no suicide attempt occurred; and 'neutral', when there is no mention of suicide, self-harm, or ideation anywhere in the stay.

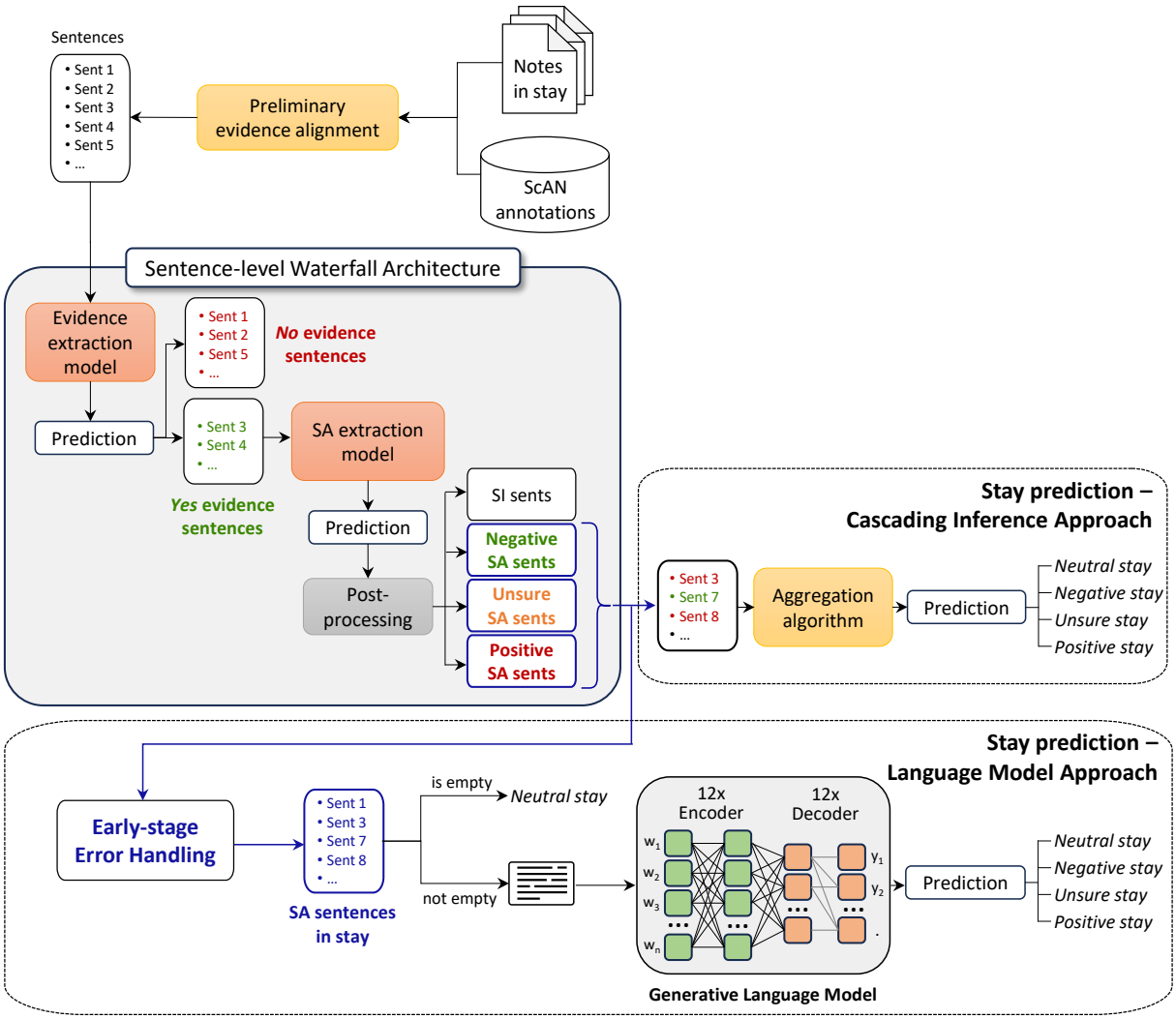

Figure 2. The comprehensive design of the multi-stage framework with sentence-level waterfall architecture for stay-level suicide attempt (SA) risk prediction.

Table 2 summarizes the final distribution of stay-level annotations across the training, validation, and test splits. Following the methodology of the original ScAN paper [14], we augmented the corpus by sampling an additional 1,195 'neutral' hospital stays from MIMIC-III to reflect the real-world imbalance in which most hospitalizations do not involve suicidality. Patient-level separation was strictly enforced across splits to prevent information leakage.

3.2. Overview of the Proposed Method

The goal of this study is to automatically classify each hospital stay into one of four suicide attempt (SA) risk categories: 'positive', 'unsure', 'negative', or 'neutral'. This problem is challenging because clinical notes are lengthy, heterogeneous, and often contain conflicting or irrelevant information from multiple sources such as physicians, nurses, and family members. To address these challenges, we introduce a **multi-stage framework** built around a **sentence-level waterfall architecture**, illustrated in Figure 2.

The design is motivated by the way clinicians gradually narrow down evidence: first identifying potentially relevant notes, then focusing on the details that matter for suicide risk assessment, and finally integrating evidence across time to form a judgment about the entire hospital stay. Our framework consists of three main components:

1. **Preliminary Evidence Alignment.** Raw hospital stay notes from MIMIC-III are segmented into sentences and aligned with ScAN annotations. This preprocessing step produces a unified dataset of sentences labeled as suicide-related or neutral, enabling fine-grained analysis at the sentence level.
2. **Sentence-level Waterfall Architecture.** The core of the framework is a two-stage sentence classifier: (i) an *evidence extraction model* that filters suicide-related sentences from irrelevant text, and (ii) an *SA extraction model* that further categorizes suicide-related sentences into 'positive SA', 'unsure SA', 'negative SA', or 'SI'. This step eliminates clinically irrelevant information, reduces noise, and provides interpretable sentence-level predictions linked directly to textual evidence.
3. **Stay-level Suicide Risk Prediction.** Sentence-level predictions are aggregated to classify the entire hospital stay. We explore two strategies: (i) a *cascading inference approach* that applies a hierarchical aggregation algorithm over sentence labels, and (ii) a *language model approach* that uses a fine-tuned T5 transformer with early-stage error handling to integrate multiple pieces of evidence.

This multi-stage design provides several advantages over prior work: (i) reducing "note bloat" and **irrelevant content**, a common issue in EHR-based NLP [4]; (ii) explicitly resolving **contradictions** across sentences and encounters, reflecting the realities of hospital documentation; (iii) improving **interpretability** by linking stay-level predictions to sentence-level evidence, aligning with recent calls for transparent AI in healthcare [6, 7]. In the following subsections, we describe each component of the framework in detail.

3.2.1. Preliminary Evidence Alignment

We transform the raw clinical notes from MIMIC-III [12] into a sentence-level dataset aligned with ScAN annotations [14]. The pipeline (Algorithm 1) concatenates all notes within a stay, segments text into sentences using a clinical-domain splitter (e.g., SciSpaCy) [41], maps ScAN character spans to sentence boundaries, normalizes surface forms while *preserving* negation and temporality cues [28, 30, 31], and removes within-stay duplicates to mitigate note redundancy [4]. De-identification artifacts are already handled by MIMIC-III's DUA-compliant process [12, 29].

Span-to-sentence mapping. Let a hospital stay d contain a concatenated note string T_d, which is segmented into sentences $S_d = \{s_i\}_{i=1}^{n}$ with character spans $[\hat{b}_i, \hat{e}_i]$. Let ScAN provide K_d gold spans $A_d = \{(b_k, e_k, y_k)\}_{k=1}^{K_d}$ with label $y_k \in \{\texttt{posSA}, \texttt{unsureSA}, \texttt{negSA}, \texttt{posSI}, \texttt{negSI}\}$. We assign each annotation to the sentence that has the maximal character-span overlap with it. For a given sentence i and annotation k, define:

$$\omega_{ik} = \frac{\left|[b_k, e_k] \cap [\hat{b}_i, \hat{e}_i]\right|}{\left|[b_k, e_k]\right|}, \quad \tilde{i}(k) = \arg\max_i \omega_{ik}.$$

If $\max_i \omega_{ik} \geq \tau$ (we use $\tau = 0.5$), assign y_k to $s_{\tilde{i}(k)}$; otherwise the sentence remains unlabeled. Sentences with no assigned ScAN label are set to 'neutral'.

Normalization and duplicate control. We apply light normalization (lowercasing; consistent whitespace; domain-specific acronym expansion) while *retaining* negation tokens (e.g., *no, denies, not*) and temporal markers (e.g., *history of, prior, status post*). To reduce "note bloat" [4], we remove exact duplicates within a

Algorithm 1 ScAN Matching and Evidence Alignment (concise)

Require: Concatenated text T_d for stay d; ScAN spans $A_d = \{(b_k, e_k, y_k)\}$; threshold τ
Ensure: Sentence list $\{(s_i, \ell_i)\}_{i=1}^{n}$ with labels $\ell_i \in$ {`posSA`, `unsureSA`, `negSA`, `posSI`, `negSI`, 'neutral'}

$\{(s_i, [\hat{b}_i, \hat{e}_i])\}_{i=1}^{n} \leftarrow$ SENTENCEIZE(T_d) ▷ clinical splitter
Initialize $\ell_i \leftarrow$ '*neutral*' for all $i = 1, \ldots, n$
for each $(b_k, e_k, y_k) \in A_d$ **do**
    $i^\star \leftarrow \arg\max_i \Omega((b_k, e_k), (\hat{b}_i, \hat{e}_i))$
    if $\Omega((b_k, e_k), (\hat{b}_{i^\star}, \hat{e}_{i^\star})) \geq \tau$ **then**
        $\ell_{i^\star} \leftarrow y_k$
    end if
end for
for $i = 1$ to n **do**
    $s_i \leftarrow$ NORMALIZE(s_i) ▷ lowercase, acronym expand, punct/space
    PRESERVECUES(s_i) ▷ negation/temporality tokens kept
end for
$\{(s_i, \ell_i)\} \leftarrow$ DEDUPLICATEWITHINSTAY($\{(s_i, \ell_i)\}$) ▷ exact/near-dup removal
$\{(s_i, \ell_i)\} \leftarrow$ SORTCHRONOLOGICALLY($\{(s_i, \ell_i)\}$)
return $\{(s_i, \ell_i)\}_{i=1}^{n}$

stay (by high Jaccard similarity), keeping the first chronological occurrence.

Outputs. The result is a compact, chronologically ordered list of unique sentences with high-precision labels, forming the input to the sentence-level waterfall models (Section 3.2.2).

3.2.2. Sentence-level Waterfall Architecture

Given the extreme imbalance between suicide-related and unrelated sentences and the prevalence of conflicting accounts in EHR notes [4], we structure inference as a *waterfall*: first isolate potentially relevant evidence, then assign clinically meaningful subtypes. This architecture is designed to mimic the incremental approach humans take when analyzing complex text: first, extracting the most relevant information out of the vast information amount, then further considering it for our specific purposes. It yields sentence-aligned rationales that support interpretability and auditability [6, 7]. The waterfall structure consists of two consecutive stages:

- **Evidence Sentence Extraction Stage:** This stage addresses the significant imbalance between suicide-related and unrelated sentences in clinical notes. Using a binary classification model, we filter out irrelevant sentences, reducing the prediction space and computational load for the subsequent multi-class model, ensuring focus on key suicide-related clinical information.
- **SA Sentence Extraction Stage:** The sentences identified as potentially suicide-related are then subjected to a more thorough analysis. This stage employs a multi-class classification model to categorize each sentence into one of four classes: 'positive SA', 'unsure SA', 'negative SA', or 'SI' sentences. This classification step brings more focus on SA-related sentences and their suicide risks for the subsequent stay prediction stage.

Both stages utilize Bi-LSTM network structures, as illustrated in Figure 3, to capture the semantic nuances and contextual information within the sentences. Finally, the individual sentence predictions are reconsidered in the context of the entire stay by a post-processing step, ensuring the predictions maintain consistency across the patient's complete clinical narrative. With this waterfall manner, we not only improve the efficiency of our model but also enhance its ability to focus on the most pertinent information for suicide risk detection and facilitate the model's explainability.

Textual representation. A custom FastText model [42] is employed for text embedding,

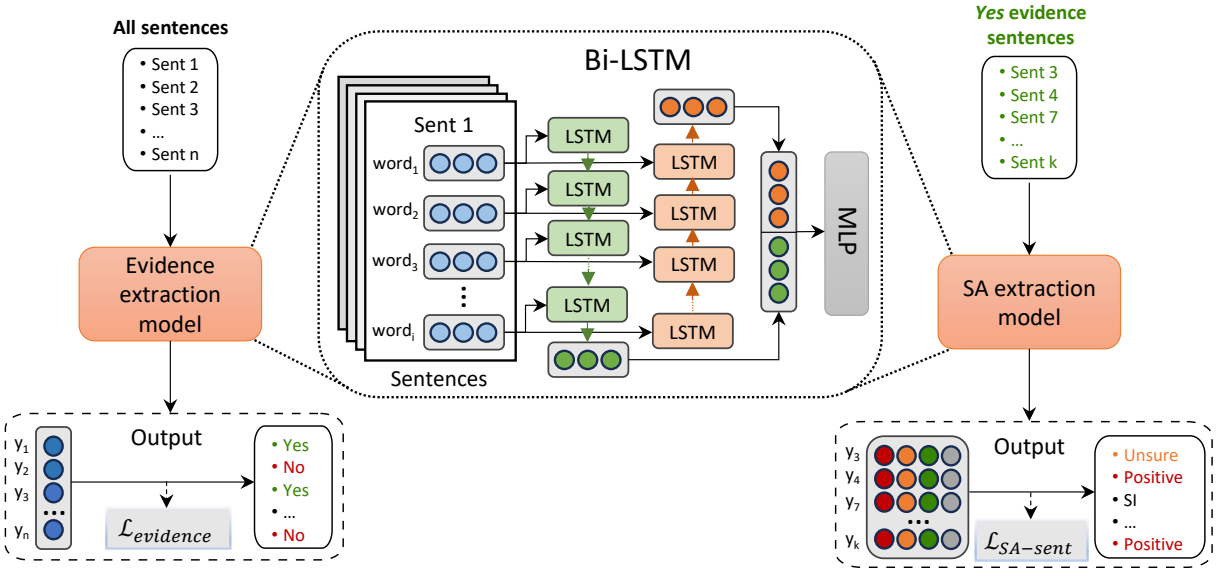

Figure 3. Sentence-level waterfall. Stage 1 filters suicide-*related* evidence from background text (binary classifier). Stage 2 refines evidence into positive SA/unsure SA/negative SA/SI (multi-class classifier). Both stages share FastText embeddings and a Bi-LSTM encoder; outputs feed stay-level inference.

which specializes in clinical text and facilitates fast inference in real-world applications. Despite using a non-transformer approach, the model showed superior performance in our setting, possibly due to its effectiveness in handling clinical terminology and frequent misspellings. Formally, let $S = (w_1, w_2, \ldots, w_n)$ be the input sentence, where w_i represents the i-th word in the sentence. Each word w_i is transformed into an embedding vector $\mathbf{x}_i \in \mathbb{R}^N$, where N is the dimensionality of the embedding space. The sequence of word embeddings forms a matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]$.

Waterfall extraction architecture. Taking the sequence matrix $\mathbf{X}$ as the input, both extraction models employ Bidirectional Long Short-Term Memory Network (Bi-LSTM) architecture to learn the sequential information from the words in both forward and backward directions of the text sequence. This shared architecture allows the model to capture the meaning of the whole sentence, focusing on suicide-related aspects, and results in rich representations of the input data.

The matrix $\mathbf{X}$ is processed through a Bi-LSTM network to capture bidirectional sequential information:

$$\overrightarrow{\mathbf{h}_n} = \overrightarrow{\text{LSTM}}(\mathbf{X}), \quad \overleftarrow{\mathbf{h}_n} = \overleftarrow{\text{LSTM}}(\mathbf{X}).$$

The concatenated forward and backward hidden states are passed through an MLP for label prediction:

$$\hat{y} = \sigma\left(\text{LeakyReLU}\left(\left[\overrightarrow{\mathbf{h}_n} \,\|\, \overleftarrow{\mathbf{h}_n}\right]\mathbf{W}_1 + \mathbf{B}_1\right)\mathbf{W}_2 + b_2\right),$$

where σ is sigmoid for evidence extraction or softmax for SA extraction, and $\mathbf{W}_1$, $\mathbf{B}_1$, $\mathbf{W}_2$, b_2 are learnable parameters.

Post-processing for sentence-level model. After the SA sentence extraction phase, evidence sentences are divided into four classes, with frequent confusion between 'positive SA' and 'unsure SA' classes requiring post-processing. This confusion arises from the ScAN dataset's stay-level annotation approach, where annotators first assess suicide attempt likelihood for the

entire stay, then label matching sentences, causing similar characteristics between 'positive SA' and 'unsure SA' sentences even when evidence clarity varies.

Post-processing reconsiders all predicted 'positive SA' and 'unsure SA' sentences per stay. Formally, let: $S = \{s_1, s_2, \ldots, s_n\}$ be the set of sentences initially classified as 'positive SA' or 'unsure SA'; $\mathcal{SP}$ be the set of suicide phrases; $\mathcal{PP}$ be the set of past-tense phrases; $\mathcal{DP}$ be the set of denial phrases.

Define the function $f(s)$ for each sentence $s \in S$ as:

$$f(s) = \begin{cases} 1, & \text{if } (s \cap \mathcal{SP} \neq \emptyset) \wedge (s \cap \mathcal{NP} = \emptyset), \\ 0, & \text{otherwise.} \end{cases}$$

where $\mathcal{NP} = \mathcal{PP} \cup \mathcal{DP}$ is the set of non-permissible phrases. Next, aggregating for the entire set S, and the relabeling function $\mathrm{R}(s)$ for each sentence $s \in S$ is then defined by:

$$\mathcal{R}(s) = \begin{cases} `positiveSA', & \text{if } \max_{s \in S} f(s) = 1, \\ `unsureSA', & \text{if } \max_{s \in S} f(s) = 0. \end{cases}$$

For that process, three sets of phases are predefined:

- The **suicide phrases set** $\mathcal{SP}$ comprises: "suicide attempt", "suicide note", "self-inflicted", "intentional overdose", and "commit suicide".
- The **past-tense phrases set** $\mathcal{PP}$ includes: "status post", "previous", "past", "prior", and "history".
- The **denial phrases set** $\mathcal{DP}$ contains: "not", "deny", "never", "unintentional".

Interpretability and clinical transparency. The waterfall provides sentence-level rationales by construction: stage 1 exposes *why* a sentence was considered; stage 2 specifies *what* risk subtype it expresses. Post-processing then documents deterministic consistency operations. Together, these steps yield transparent, auditable evidence chains while substantially reducing irrelevant text before stay-level inference.

Output to stay-level models. The module outputs (i) a filtered, chronologically ordered list of evidence sentences, (ii) subtype labels with calibrated confidence scores, and (iii) consistency-adjusted labels. These artifacts feed the cascading aggregation algorithm and the language-model classifier.

3.2.3. Suicide Attempt Stay Prediction

In this phase, the SA-related sentences from the previous sentence-level waterfall model are considered in the stay scope, to identify the suicide attempt risk of the stay. Two distinct approaches are proposed for this problem: the **Cascading Inference Approach** and the **Language Model Approach**. While the Cascading Inference Approach is quite straightforward when using an algorithm to aggregate the prediction SA classes of sentences in the stay clinical notes, the Language Model Approach with its text understanding ability, considers the stay suicide attempt risk.

Cascading inference approach. Predicted SA-related sentences and their risk levels are passed through an aggregation algorithm to classify the stay into four categories, as illustrated in Algorithm 2. The details of how the aggregation algorithm works can be stated as follows:

1. **Filtering SA-unrelated stays:** The first step is determining whether the stay contains any SA sentences or not, if no SA sentence is detected, the stay is assigned as neutral.
2. **Detecting past SA sentences:** Since the decision of suicide attempt stay should only depend on the statement in the current period, the past SA sentences are detected and removed from the considered texts. Simply, the past SA sentence is detected by having one word in the aforementioned suicide phases set and one word in the past phases set at the same time.
3. **Hierarchically classifying stays by aggregating sentence-level predictions:** Analysis of sentence-level predictions

Algorithm 2 Cascading Aggregation for Stay Classification

Require: P: set of past-tense phrases; SP: set of suicide phrases
function ClassifyStay(S) ▷ S: list of (text, label) sentences
  if $S = \emptyset$ **then**
    return 'neutral'
  end if
  $S_{\text{present}} \leftarrow$ FilterPastSentences(S)
  for all $s \in S_{\text{present}}$ **do**
    if s.label = '*positive*' **then**
      return 'positive'
    else if s.label = '*negative*' **then**
      return 'negative'
    else if s.label = `UNSURE` **then**
      return `UNSURE`
    end if
  end for
  return 'neutral'
end function
function FilterPastSentences(S)
  $S_{\text{present}} \leftarrow \emptyset$
  for all $s \in S$ **do**
    $t \leftarrow s$.text
    if $(\exists w \in t : w \in SP) \wedge (\exists u \in t : u \in P)$ **then**
      continue ▷ drop past SA mentions
    else
      $S_{\text{present}} \leftarrow S_{\text{present}} \cup \{s\}$
    end if
  end for
  return S_{present}
end function

reveals that 'positive SA' class achieves the highest certainty among three classes: 'positive SA', 'negative SA', and 'unsure' 'SA'. Furthermore, within these three suicide risk levels, the 'positive SA' and 'negative SA' classes are more distinctly defined compared to the 'unsure SA' class.

The hierarchical aggregation algorithm is reasonable since the sentence-level prediction has been post-processed and reached a certain consensus degree among present evidence sentences. This cascading inference approach indicates that the stay result can be derived in a fast and efficient manner with a simple aggregation algorithm.

Language model approach. The cascading inference strategy effectively classifies SA stays using sentence-level classifications but depends largely on sentence-level model precision. When these models fall short, a more sophisticated language model approach becomes necessary.

The T5 (Text-to-Text Transfer Transformer) architecture [15] is employed as the core language model for this approach, utilizing an encoder-decoder Transformer structure with 12 blocks each, to effectively capture complex relationships in SA-related sentences and grasp the full patient context. In more detail, this approach collects all SA-related sentences from stay clinical notes, orders them chronologically, and marks stays without SA-related sentences as 'neutral' stay. Otherwise, the language model processes the concatenated SA-related sentences x as input through encoder $\mathcal{E}$ and decoder $\mathcal{D}$ to produce output text $y \in \{\text{'}positive\text{'}, \text{'}unsure\text{'}, \text{'}negative\text{'}, \text{'}neutral\text{'}\}$:

$$y = \mathcal{D}(\mathcal{E}(x))$$

3.3. Model Training Procedure

3.3.1. Sentence-level Waterfall Architecture Training Objectives

The sentence-level waterfall architecture comprises two sequential stages, each trained independently to optimize specific objectives.

Evidence sentence extraction loss. The first stage involves identifying sentences containing evidence of suicide-related content. We employ a weighted binary cross-entropy loss function to address the class imbalance problem:

$$\mathcal{L}_{\text{evidence}} = -\frac{1}{N_d}\sum_{i=1}^{N_d}(\alpha \cdot y_i \log \hat{y}_i + (1-y_i)\log(1-\hat{y}_i)),$$

where N_d is the number of sentences per stay d, $y_i \in \{0, 1\}$ is ground-truth label, $\hat{y}_i$ is predicted

probability, and α is negative-to-positive ratio to address class imbalance.

SA sentence extraction loss The second stage classifies the evidence sentences into four suicide risk classes, utilizing a weighted cross-entropy loss to manage the multi-class imbalance:

$$\mathcal{L}_{\text{SA-sent}} = -\frac{1}{N_d}\sum_{i=1}^{N_d}\sum_{c=1}^{C} w_c y_{i,c}\log(\hat{y}_{i,c}).$$

Here, total C classes, with w_c as class weights, while $y_{i,c}$ and $p_{i,c}$ represent ground-truth and predicted probabilities for sentence i of class c respectively. w_c balances learning across classes.

3.3.2. Language Model Training Procedures for Stay Classification

The stay classification stage employs a fine-tuned T5 model, which also includes an early-stage error handling training process. It integrates information from previous stages, handles potential upstream errors, and optimizes performance for the final stay-level risk detection.

Early-stage error handling training process. Traditional methods use only ground-truth SA sentences for training with three classes ('positive', 'unsure', 'negative'), overlooking potential errors in the sentence extraction stages. This leads to misclassification of neutral stays in the later stay prediction stage. We proposed an error handling training approach using both golden and predicted SA sentences as input, with four classes including 'neutral'. This enables error correction from the SA sentence extraction stage and improves model performance by handling imperfect inputs more effectively.

Generative language model loss. The model is trained using teacher forcing, a technique where the ground truth from a prior time step is used as input for the current time step during training. The loss function for the T5 model is based on the standard cross-entropy loss, which is commonly used in sequence-to-sequence tasks. Given an input sequence x and target sequence y, the loss is computed as:

$$\mathcal{L}_{\text{stay}} = -\sum_{t=1}^{T}\log \mathrm{P}(y_t|y_{<t}, x),$$

where T is the length of the target sequence, y_t is the target token at position t, $y_{<t}$ represents all preceding tokens, and $\mathrm{P}(y_t|y_{<t}, x)$ is the model's predicted probability of the correct token at each position.

4. Results and Discussion

In this section, we evaluate the performance of the multi-stage model and compare it with comparative models. Ablation tests are also conducted to evaluate the contributions of the model's components, as well as the strategies and resources utilized. The metrics of macro-averaged precision (`P`), recall (`R`), and F1-score (`F1`) are employed for evaluation.

4.1. Model Performance and Comparison

We use ScANER [14] as the comparative model, which applies a multi-head attention layer on the vector representation of evidence-extracted paragraphs to predict class labels. Additionally, we compare the suicide attempt stay prediction results of our two approaches:

- **Language Model Approach** is based on T5, a well-known Text-To-Text Transfer Transformer developed by Google AI.
- **Cascading Inference Approach** classifies stay by applying an aggregation algorithm using the predicted SA-related sentences.

The results are presented in Table 3. Both our Large Language Model and Cascading Inference approaches significantly outperform ScANER across all metrics and labels. In the dataset with added neutral stays, our models achieve macro F1-scores of 0.90 and 0.93, respectively, compared to ScANER's

0.78. Notably, the Cascading Inference model demonstrates substantial improvements in predicting unsure and negative cases, with an F1-score increase from 0.52 to 0.83.

The Cascading Inference model demonstrates the best overall performance, with the highest or tied-highest scores in most categories. It achieves perfect recall for positive stays, balanced performance (0.83 macro F1-score) for negative and unsure stays, and near-perfect results (0.99 macro F1-score) for neutral stays. The Large Language Model approach also shows strong performance, with 0.90 macro F1-score. Our models also maintain their strong performance on the original ScAN dataset, particularly excelling in positive, negative, and unsure stay classifications. We also investigates the influence of different clinical note types on the Cascading Inference model, as shown in Table 4. The results confirm that all note types are contributed to the model performance, but at various degrees, with the most informative note types are discharge summary, nursing, and radiology.

These results clearly demonstrate the effectiveness of our sentence-level waterfall approach in accurately classifying lengthy clinical notes, offering substantial improvements over existing methods.

4.2. Performance of Sentence-level Waterfall Architecture

Our model's performance stems from the effectiveness of the sentence-level waterfall architecture. This section examines the performance of two key components: evidence sentence extraction and SA sentence extraction models.

For evidence extraction models, which constitute the first stage of the waterfall architecture, two model architectures are put into comparison: T5 and Bi-LSTM architectures. The second stage of SA sentence extraction takes the evidence sentences as input and classifies them into four risk categories, being tested the following configurations:

- **T5-T5**: The output of T5 evidence extraction model is passed into SA extraction model with T5 architecture.
- **LSTM-LSTM**: The output of the Bi-LSTM evidence extraction model is provided as input to the SA extraction model with a Bi-LSTM architectural configuration.
- **LSTM-T5**: The results produced by the Bi-LSTM evidence extraction model are fed into the SA extraction model employing the T5 architecture.

We compare our models with ScANER's evidence retriever, which uses multi-task objective training with medRoBERTa [24] to predict suicide evidence in text paragraphs. Since our models operate at the sentence level, we aggregate sentence labels into group labels for 20 consecutive sentences, similar to ScANER's evidence retriever module for comparison.

The comparative results of the evidence extraction models are shown in Table 5, and the SA extraction models include cascaded errors from the previous evidence extraction stage are presented in Table 6 (the `Neutral SA` class includes no evidence sentences and SI sentences). Our evidence model's performance surpassed the ScANER with the increase of 6% macro F1-score, and our SA model further outperformed the ScANER with 0.85 macro F1-score compared to 0.63 macro F1-score, proving the efficiency of the sentence-level waterfall architecture.

Within the sentence-level models, the Bi-LSTM architecture handles each sentence more effectively than the T5 architecture, showing that the focus on more detailed and word-based features of the Bi-LSTM structure strongly benefits the model's performance. The results imply the strengths of sentence-level waterfall architecture, not only reducing a tremendous

Table 3. Performance of stay suicide risk models across different classes and macro-averaged scores

Model	Positive			Negative_Unsure			Neutral			Macro		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Adding neutral stays to ScAN dataset												
ScANER [14]	0.81	0.93	0.87	0.48	0.58	0.52	0.98	0.93	0.96	0.76	0.81	0.78
Large Language Model‡	0.90 (±0.03)	**1.00** (±0.00)	0.95 (±0.02)	**0.86** (±0.05)	0.67 (±0.07)	0.75 (±0.05)	0.99 (±0.01)	**0.99** (±0.02)	**0.99** (±0.01)	**0.92** (±0.02)	0.89 (±0.03)	0.90 (±0.02)
Cascading Inference‡	**0.94** (±0.02)	**1.00** (±0.00)	**0.97** (±0.01)	0.83 (±0.05)	**0.83** (±0.05)	**0.83** (±0.02)	**1.00** (±0.00)	0.98 (±0.02)	**0.99** (±0.01)	**0.92** (±0.01)	**0.94** (±0.03)	**0.93** (±0.02)
Original ScAN dataset												
ScANER* [14]	-	-	-	-	-	-	-	-	-	-	-	-
Large Language Model‡	0.90 (±0.03)	**1.00** (±0.00)	0.95 (±0.02)	**0.86** (±0.05)	0.67 (±0.07)	0.75 (±0.05)	0.67 (±0.17)	**0.50** (±0.25)	**0.57** (±0.13)	0.81 (±0.06)	**0.72** (±0.08)	**0.76** (±0.05)
Cascading Inference‡	**0.94** (±0.02)	**1.00** (±0.00)	**0.97** (±0.01)	0.83 (±0.05)	**0.83** (±0.05)	**0.83** (±0.02)	**1.00** (±0.00)	0.25 (±0.25)	0.40 (±0.27)	**0.92** (±0.01)	0.69 (±0.07)	0.73 (±0.06)

**The original paper did not provide these results* *P: Precision, R: Recall, and F: F1-score.*

‡: Our results are mean ± standard deviation over 10 runs. The highest results in a column is highlighted in bold.

Table 4. Performance of Cascading Inference model with inputs of different clinical note types

Note type	Positive			Negative_Unsure			Neutral			Macro		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
All	0.94	**1.00**	**0.97**	0.83	**0.83**	**0.83**	**1.00**	0.98	**0.99**	0.92	**0.94**	**0.93**
Discharge summary	0.94	0.98	0.96	0.88	0.78	0.82	0.99	0.98	**0.99**	0.94	0.91	0.92
Nursing	0.94	0.67	0.78	0.80	0.44	0.57	0.92	0.98	0.95	0.89	0.70	0.77
Radiology	**1.00**	0.78	0.88	**1.00**	0.22	0.36	0.94	**0.99**	0.96	**0.98**	0.66	0.73
Physician	0.78	0.15	0.25	0.75	0.17	0.27	0.78	1.00	0.88	0.77	0.44	0.47
Social work/Consult/Nutrition	**1.00**	0.09	0.16	**1.00**	0.06	0.11	0.72	1.00	0.84	0.91	0.38	0.37
Respiratory/Rehab services	**1.00**	0.04	0.08	**1.00**	0.06	0.11	0.70	1.00	0.82	0.90	0.37	0.34
General/Case management	0.60	0.07	0.12	**1.00**	0.06	0.11	0.71	1.00	0.83	0.77	0.38	0.35
Echo/ECG	**1.00**	0.02	0.04	0.00	0.00	0.00	0.70	1.00	0.82	0.57	0.34	0.29

P: Precision, R: Recall, and F: F1-score. The highest results in a column is highlighted in bold.

Table 5. Performance of evidence sentence extraction models across different classes and macro scores

Model	Yes			No			Macro		
	P	**R**	**F1**	**P**	**R**	**F**	**P**	**R**	**F**
ScANER [14]	0.79	0.87	0.83	0.95	0.91	0.93	0.87	0.89	0.88
T5	0.79	0.90	0.84	0.81	**0.93**	0.87	0.80	0.92	0.86
Bi-LSTM	**0.88**	**0.99**	**0.93**	**0.99**	0.92	**0.96**	**0.94**	**0.95**	**0.94**

amount of text and noise from the SA-unrelated sentences, but also improving the model's interpretability and proving the reliability of the model.

Table 6. Performance of suicide attempt (SA) sentence extraction models across different classes and macro scores

Model	Positive SA			Neg_Unsure SA			Neutral SA			Macro		
	P	R	F	P	R	F	P	R	F1	P	R	F
ScANER [14]	0.71	0.74	0.73	0.19	0.26	0.22	0.95	0.92	0.93	0.62	0.64	0.63
T5-T5	0.88	**0.94**	0.91	0.51	**0.81**	0.63	**0.97**	0.89	0.93	0.79	**0.88**	0.82
LSTM-T5	**0.92**	0.93	0.92	0.60	0.72	0.65	0.96	**0.94**	**0.95**	**0.83**	0.86	0.84
LSTM-LSTM	0.91	**0.94**	**0.93**	**0.61**	0.77	**0.68**	0.96	0.93	0.94	**0.83**	**0.88**	**0.85**

P: Precision, R: Recall, and F: F1-score. Neg_Unsure: Negative_Unsure.
The highest result in a column is highlighted in bold.

4.3. Discussion

We assess the impact of each model component through an ablation study, systematically removing it to measure performance degradation. The results, summarized in Figure 4, highlight the importance of each component in our framework and validate our framework design. The **waterfall sentence architecture** is the most critical, its removal leads to substantial performance degradation across all categories and the macro-averaged F1, underscoring its essential role. Eliminating the **sentence-level post-processing** module also causes a significant reduction, particularly for the 'unsure' category. The **evidence sentence extraction** step within the waterfall architecture also proves to be notably crucial, without filtering the noisy sentences showing the model's robustness in noise filtering, especially for 'unsure' and 'neutral' stays. Likewise, removing the **early-stage error handling** mechanism leads to substantial performance drops in the language model approach, especially for 'negative' stays, highlighting its importance in managing upstream errors. **Using cascading inference with T5 sentence-level model** (instead of the bi-LSTM model) results in lower performance, particularly for the 'unsure' category. Interestingly, incorporating **Suicidal Ideation (SI) sentences** into stay prediction inputs negatively affects performance, likely due to increased uncertainty in identifying suicide events. Replacing our **Cascading Inference** with a **Large Language Model (LLM)** also decreases performance, though to a lesser degree. The substitution of **BERT embeddings for FastText embeddings** demonstrates diminished performance, verifying the better embedding choice for our settings.

4.4. Error analysis

The error analysis in Table 7 highlights specific misclassification tendencies at the hospital stay-level. The model sometimes over-interprets evidence of overdose, leading to 'positive' predictions for 'unsure' cases with conflicting patient statements, where the patient denied suicidality despite there were "numerous empty and partially empty pill bottles". Furthermore, extremely dangerous overdose and substance abuse 'neutral' cases might be predicted as 'unsure' or even 'positive', interpreting inherent risk as potential self-harm even without explicit suicidal declarations for the current admission. Finally, the final model uses only suicide attempt-related sentences extracted through the sentence-level waterfall architecture, which helps the overall model focus more on suicide attempt-related cases, but also creates a blind spot. For instance, in the last example, the sentence "Patient not suicidal" is classified as suicide ideation and subsequently excluded from

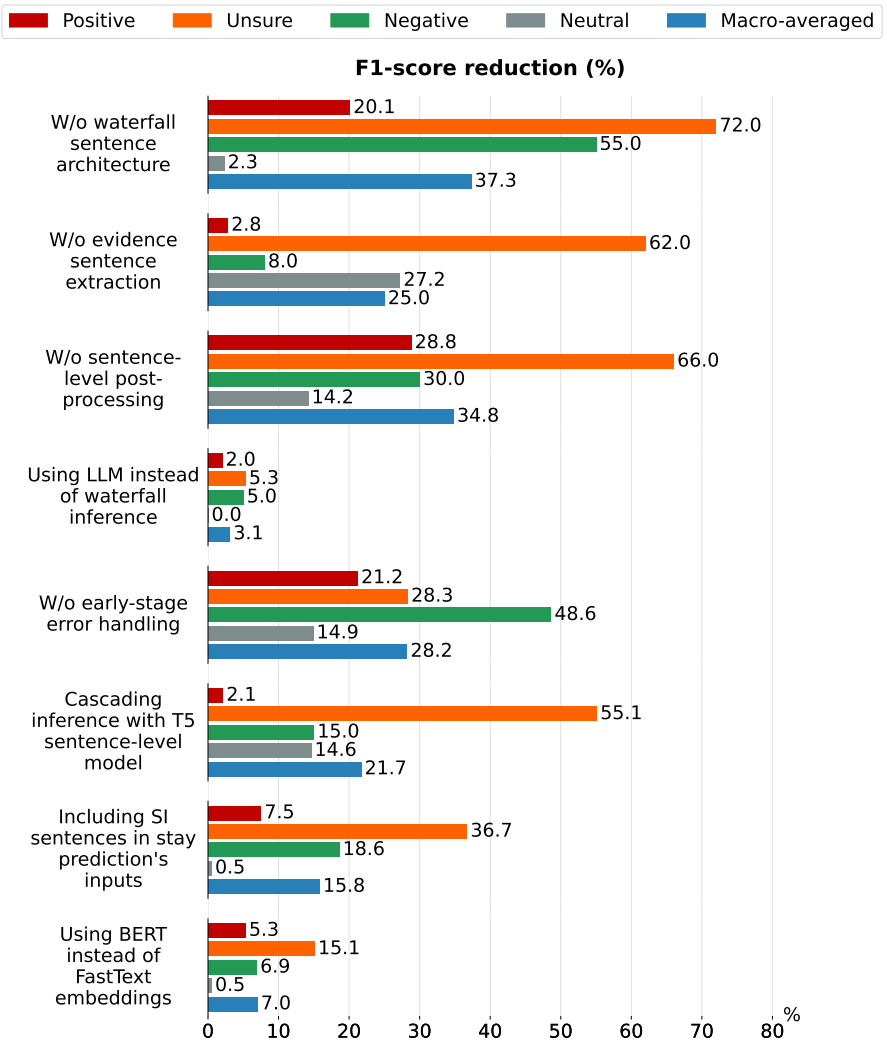

Figure 4. Model components analysis showing the performance drop (%) compared to the full model when a given component is removed.

the final prediction, leading to a misclassification to 'unsure' when the patient is, in fact, negative for suicide risk.

4.5. Ethical Considerations

This work addresses suicide risk assessment, a sensitive domain where algorithmic errors can have severe consequences. While our model demonstrates promising performance on the ScAN dataset, several ethical concerns must be carefully considered before any clinical deployment. First, the model is trained on MIMIC-III data from a single healthcare system, which may not generalize to diverse populations with different demographic characteristics, healthcare access patterns, or cultural backgrounds. Second, while our approach provides sentence-level evidence extraction to support interpretability, this system has not been validated by clinicians in real-world settings.

We emphasize that this work represents

Table 7. Representative mis-classification patterns and underlying causes of the classification model at **hospital stay-level**

Label	**Prediction**	**Cause**	**Representative examples**
Unsure	Positive	Model over-interprets physical evidence of overdose.	***Numerous empty and partially empty pill bottles*** *were found around her including ativan, vicodin, fioricet, and tylenol. The patient was unable to recall the events of the last two to three days and claims that she was only taking five tylenol pills.* ***She denied suicidality but she had one prior suicide attempt by overdose****.*
Neutral	Unsure	Model detects potential self-harm from dangerous substance use patterns.	*Mr. [Known Lastname 805] is a 63 year old man with history of polysubstance abuse [...]* ***He is aware that these ingestions could kill him.*** *He does state that he regrets the isopropyl alcohol ingestion and recent cocaine use and that he would like to turn his life around [...]*
Neutral	Positive	Model misclassifies accidental drug overdose as suicide-related when no self-harm context exists.	*36 year old f [...] unresponsiveness secondary to* ***cocaine plus/minus opioid overdose*** *[...] intubated for airway protection*
Negative	Unsure	The last sentence is classified as suicide ideation and is excluded for stay prediction.	***Depression (possible suicide attempt 4 - 30)****. 51 year old female with past medical history significant for etoh abuse, ? etoh withdrawal seizures, and ? suicide attempt who was found down at home by her daughter. [...] She had a history of depression and etoh use;* ***per patient she was not trying to commit suicide [...] Patient not suicidal.***

a research prototype intended to assist, not replace, clinical judgment. The model should be viewed as a preliminary screening tool that may help clinicians efficiently identify relevant evidence in lengthy EHR notes, but final risk assessments must always be made by qualified healthcare professionals through comprehensive evaluation. Before any clinical adoption, rigorous prospective validation is required, including clinician-in-the-loop studies, fairness audits across demographic subgroups, and evaluation of how the system integrates into existing clinical workflows without introducing new harms or exacerbating existing disparities in mental healthcare access and quality.

5. Conclusion

This research addresses fundamental challenges in clinical suicide risk assessment by introducing a multi-stage framework using a sentence-level waterfall architecture to classify suicide risk from clinical notes. The framework's transparent, sentence-level evidence provision transforms clinical practice by enabling practitioners to understand and validate each

prediction. This interpretability fosters trust and supports evidence-based decision-making in critical scenarios, while comprehensive stay-level risk assessments account for temporal changes across hospital stays with clinical sophistication. The framework outperformed existing baselines on a suicide attempt dataset, achieving a 60% improvement in classifying uncertain and negative cases, directly addressing the most problematic areas. These achievements establish a new standard for clinical decision support systems, demonstrating that advanced predictive accuracy and clinical transparency can coexist effectively.

However, this work also has limitations, including inability to effectively handle blurrier statements such as suicide ideation, limited labeled clinical data for suicide risk detection, and uncertain generalizability to other healthcare systems with different patient demographics or EHR structures. Future work may explore applications to social media data for population-level monitoring, a larger dataset for suicide detection at clinical hospital stay level, extension to other mental health conditions such as depression and anxiety disorders, and real-world clinical deployment validation.

Acknowledgement

This work has been supported by VNU University of Engineering and Technology under project number CN24.15.

References

[1] K. L. Lovero, P. F. Dos Santos, A. X. Come, M. L. Wainberg, M. A. Oquendo, Suicide in Global Mental Health, Current Psychiatry Reports, Vol. 25, No. 6, 2023, pp. 255–262, https://doi.org/10.1007/s11920-023-01423-x.

[2] J. Snowdon, N. G. Choi, Undercounting of Suicides: Where Suicide Data Lie Hidden, Global Public Health, Vol. 15, No. 12, 2020, pp. 1894–1901, https://doi.org/10.1080/17441692.2020.1801789.

[3] M. K. Nock, A. J. Millner, E. L. Ross, C. J. Kennedy, M. Al-Suwaidi, Y. Barak-Corren, V. M. Castro, F. Castro-Ramirez, T. Lauricella, N. Murman, et al., Prediction of Suicide Attempts Using Clinician Assessment, Patient Self-report, and Electronic Health Records, JAMA Network Open, Vol. 5, No. 1, 2022, pp. e2144373–e2144373, https://doi.org/10.1001/jamanetworkopen.2021.44373.

[4] J. Liu, D. Capurro, A. Nguyen, K. Verspoor, "Note Bloat" Impacts Deep Learning-based NLP Models for Clinical Prediction Tasks, Journal of Biomedical Informatics, Vol. 133, 2022, pp. 104149, https://doi.org/10.1016/j.jbi.2022.104149.

[5] I. Beltagy, M. E. Peters, A. Cohan, Longformer: The Long-Document Transformer, CoRR, Vol. abs/2004.05150, 2020, https://doi.org/10.48550/arXiv.2004.05150.

[6] F. Doshi-Velez, B. Kim, Towards A Rigorous Science of Interpretable Machine Learning, 2017, https://doi.org/10.48550/arXiv.1702.08608.

[7] C. Rudin, Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead, Nature Machine Intelligence, Vol. 1, No. 5, 2019, pp. 206–215, https://doi.org/10.1038/s42256-019-0048-x.

[8] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, N. Elhadad, Intelligible Models for Healthcare: Predicting Pneumonia Risk and Hospital 30-day Readmission, Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2015, pp. 1721–1730, https://doi.org/10.1145/2783258.2788613.

[9] F. R. Tsui, L. Shi, V. Ruiz, N. D. Ryan, C. Biernesser, S. Iyengar, C. G. Walsh, D. A. Brent, Natural Language Processing and Machine Learning of Electronic Health Records for Prediction of First-time Suicide Attempts, JAMIA Open, Vol. 4, No. 1, 2021, pp. ooab011, https://doi.org/10.1093/jamiaopen/ooab011.

[10] R. C. Kessler, M. S. Bauer, T. M. Bishop, R. M. Bossarte, V. M. Castro, O. V. Demler, S. M. Gildea, J. L. Goulet, A. J. King, C. J. Kennedy, et al., Evaluation of A Model To Target High-Risk Psychiatric Inpatients for An Intensive Postdischarge Suicide Prevention Intervention, JAMA Psychiatry, Vol. 80, No. 3, 2023, pp. 230–240, https://doi.org/10.1001/jamapsychiatry.2022.4634.

[11] F. Xie, D. S. L. Grant, J. Chang, B. I. Amundsen, R. C. Hechter, Identifying Suicidal Ideation and Attempt From Clinical Notes Within a Large Integrated Health Care System, The Permanente Journal, Vol. 26, No. 1, 2022, pp. 85–93, https://doi.org/10.7812/TPP/21.102.

[12] A. E. Johnson, T. J. Pollard, L. Shen, L.-w. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, R. G. Mark, MIMIC-III, A Freely Accessible Critical Care Database,

Scientific Data, Vol. 3, No. 1, 2016, pp. 1–9, https://doi.org/10.1038/sdata.2016.35.

[13] K. Huang, J. Altosaar, R. Ranganath, Clinicalbert: Modeling Clinical Notes and Predicting Hospital Readmission2019, https://doi.org/10.48550/arXiv.1904.05342.

[14] B. P. S. Rawat, S. Kovaly, W. R. Pigeon, H. Yu, SCAN: Suicide Attempt and Ideation Events Dataset, Proceedings of the conference. Association for Computational Linguistics. North American Chapter. Meeting, Vol. 2022, NIH Public Access, 2022, p. 1029, https://doi.org/10.18653/v1/2022.naacl-main.75.

[15] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring The Limits of Transfer Learning with A Unified Text-To-Text Transformer, Journal of Machine Learning Research, Vol. 21, No. 140, 2020, pp. 1–67, https://doi.org/10.48550/arXiv.1910.10683.

[16] A. Pigoni, G. Delvecchio, N. Turtulici, D. Madonna, P. Pietrini, L. Cecchetti, P. Brambilla, Machine Learning and The Prediction of Suicide in Psychiatric Populations: A Systematic Review, Translational Psychiatry, Vol. 14, No. 1, 2024, pp. 140, https://doi.org/10.1038/s41398-024-02852-9.

[17] A. M. Schoene, A. Turner, G. R. De Mel, N. Dethlefs, Hierarchical Multiscale Recurrent Neural Networks for Detecting Suicide Notes, IEEE Transactions on Affective Computing, 2021, https://doi.org/10.1109/TAFFC.2021.3057105.

[18] N. Wang, F. Luo, Y. Shivtare, V. D. Badal, K. Subbalakshmi, E. Lee, Learning Models for Suicide Prediction from Social Media Posts, 2021, https://doi.org/10.48550/arXiv.2105.03315.

[19] V. Rozova, K. Witt, J. Robinson, Y. Li, K. Verspoor, Detection of Self-harm and Suicidal Ideation in Emergency Department Triage Notes, Journal of the American Medical Informatics Association, Vol. 29, No. 3, 2022, pp. 472–480, https://doi.org/10.1093/jamia/ocab261.

[20] B. E. Bunnell, A. Tsalatsanis, C. Chaphalkar, S. Robinson, S. Klein, S. Cool, E. Szwast, P. M. Heider, B. J. Wolf, J. S. Obeid, Automated Detection and Prediction of Suicidal Behavior from Clinical Notes using Deep Learning, PLoS One, Vol. 20, No. 9, 2025, pp. e0331459, https://doi.org/10.1371/journal.pone.0331459.

[21] N. Biscoe, D. Leightley, D. Murphy, Developing a Tool for Identifying Clinical Risk From Free-Text Clinical Records: Natural Language Processing Study, JMIR AI, Vol. 4, No. 1, 2025, pp. e64898, https://doi.org/10.2196/64898.

[22] T. M. Li, J. Chen, F. O. Law, C.-T. Li, N. Y. Chan, J. W. Chan, S. W. Chau, Y. Liu, S. X. Li, J. Zhang, et al., Detection of Suicidal Ideation in Clinical Interviews for Depression using Natural Language Processing and Machine Learning: Cross-Sectional Study, JMIR medical informatics, Vol. 11, No. 1, 2023, pp. e50221, https://doi.org/10.2196/50221.

[23] S. Ji, T. Zhang, L. Ansari, J. Fu, P. Tiwari, E. Cambria, MentalBERT: Publicly Available Pre-trained Language Models for Mental Healthcare, Proceedings of the 13th Language Resources and Evaluation Conference, 2022, pp. 7184–7190, https://aclanthology.org/2022.lrec-1.778/ Accessed on June 01, 2025.

[24] S. Verkijk, P. Vossen, MedRoBERTa.nl: A Language Model for Dutch Electronic Health Records, Computational Linguistics in the Netherlands Journal, Vol. 11, 2021, pp. 141–159, https://clinjournal.org/clinj/article/view/132/ Accessed on April 09, 2025.

[25] M. Levis, J. Levy, M. Dimambro, V. Dufort, D. J. Ludmer, M. Goldberg, B. Shiner, Using Natural Language Processing to Evaluate Temporal Patterns in Suicide Risk Variation Among High-Risk Veterans, Psychiatry Research, Vol. 339, 2024, pp. 116097, https://doi.org/10.1016/j.psychres.2024.116097.

[26] M. Huang, Z. Li, Y. Hu, W. Wang, A. Wen, S. Lane, S. Selek, L. Shahani, R. Machado-Vieira, J. Soares, et al., Multi-Label Classification with Generative AI Models in Healthcare: A Case Study of Suicidality and Risk Factors, 2025, https://doi.org/10.48550/arXiv.2507.17009.

[27] M. Zaheer, G. Guruganesh, A. Dubey, J. Ainslie, C. Alberti, S. Ontanon, P. Pham, A. Ravula, Q. Wang, L. Yang, A. Ahmed, Big Bird: Transformers for Longer Sequences, Advances in Neural Information Processing Systems (NeurIPS), Vol. 33, 2020, pp. 17283–17297, https://doi.org/10.1016/j.knosys.2024.111410.

[28] W. W. Chapman, W. Bridewell, P. Hanbury, G. F. Cooper, B. G. Buchanan, A Simple Algorithm for Identifying Negated Findings and Diseases in Discharge Summaries, Journal of Biomedical Informatics, Vol. 34, No. 5, 2001, pp. 301–310, https://doi.org/10.1006/jbin.2001.1029.

[29] Ö. Uzuner, Y. Luo, P. Szolovits, Evaluating the State-Of-The-Art in Automatic De-Identification, Journal of the American Medical Informatics Association, Vol. 14, No. 5, 2007, pp. 550–563, https://doi.org/10.1197/jamia.M2444.

[30] W. Sun, A. Rumshisky, O. Uzuner, Evaluating Temporal Relations in Clinical Text: 2012 i2b2 Challenge, Journal of the American Medical Informatics Association, Vol. 20, No. 5, 2013, pp. 806–813, https://doi.org/10.1136/amiajnl-2013-001628.

[31] Y. B. Gumiel, L. E. Silva e Oliveira, V. Claveau,

N. Grabar, E. C. Paraiso, C. Moro, D. R. Carvalho, Temporal Relation Extraction in Clinical Texts: A Systematic Review, ACM Computing Surveys (CSUR), Vol. 54, No. 7, 2021, pp. 1–36, https://doi.org/10.1145/3462475.

[32] Z. Li, Y. Hu, S. Lane, S. Selek, L. Shahani, R. Machado-Vieira, J. Soares, H. Xu, H. Liu, M. Huang, Suicide Phenotyping from Clinical Notes in Safety-Net Psychiatric Hospital Using Multi-Label Classification with Pre-Trained Language Models, AMIA Summits on Translational Science Proceedings, Vol. 2025, 2025, pp. 260, https://doi.org/10.48550/arXiv.2409.18878.

[33] G. Holmes, B. Tang, S. Gupta, S. Venkatesh, H. Christensen, A. Whitton, Applications of Large Language Models in the Field of Suicide Prevention: Scoping Review, Journal of Medical Internet Research, Vol. 27, 2025, pp. e63126, https://doi.org/10.2196/63126.

[34] M. T. Ribeiro, S. Singh, C. Guestrin, "Why Should I Trust You?": Explaining the Predictions of Any Classifier, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2016, pp. 1135–1144, https://doi.org/10.18653/v1/N16-3020.

[35] S. M. Lundberg, S.-I. Lee, A Unified Approach to Interpreting Model Predictions, Advances in Neural Information Processing Systems (NeurIPS), Vol. 30, 2017, pp. 4765–4774, https://doi.org/10.48550/arXiv.1705.07874.

[36] S. Jain, B. C. Wallace, Attention is not Explanation, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, 2019, pp. 3543–3556, https://doi.org/10.18653/v1/N19-1357.

[37] L. A. Lepow, P. Adekkanattu, M. Cusick, H. Coon, B. Fennessy, S. O'Connell, C. Pierce, J. Rabbany, M. Sharma, M. Olfson, et al., A Natural Language Processing Pipeline based on the Columbia Suicide Severity Rating Scale, MedRxiv, 2024, pp. 2024–12, https://doi.org/10.1101/2024.12.19.24319352.

[38] Y. Song, P. Zhou, C. Escobar-Viera, C. Biernesser, W. Huang, J. Hu, Two-Stage Voting for Robust and Efficient Suicide Risk Detection on Social Media, 2025, https://doi.org/10.48550/arXiv.2510.08365.

[39] A. Seyedsalehi, J. Bailey, M. G. Ogonah, T. R. Fanshawe, S. Fazel, Prediction Models for Self-harm and Suicide: A Systematic Review and Critical Appraisal, BMC Medicine, Vol. 23, No. 1, 2025, pp. 549, https://doi.org/10.1186/s12916-025-04367-6.

[40] Y.-h. Sheu, J. Simm, B. Wang, H. Lee, J. W. Smoller, Continuous Time and Dynamic Suicide Attempt Risk Prediction with Neural Ordinary Differential Equations, npj Digital Medicine, Vol. 8, No. 1, 2025, pp. 161, https://doi.org/10.1038/s41746-025-01552-y.

[41] M. Neumann, D. King, I. Beltagy, W. Ammar, ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing, Proceedings of the 18th BioNLP Workshop and Shared Task, Association for Computational Linguistics, 2019, pp. 319–327, https://doi.org/10.18653/v1/W19-5034.

[42] P. Bojanowski, É. Grave, A. Joulin, T. Mikolov, Enriching Word Vectors with Subword Information, Transactions of the Association for Computational Linguistics, Vol. 5, 2017, pp. 135–146, https://doi.org/10.1162/tacl_a_00051.

A. Training Environment and Hyperparameters Configurations

This appendix details the computational environments, software stacks, and hyperparameter configurations used for training and evaluating all baseline models.

A.1. Hardware and Software Environment

A Linux server was used for the experiments and was configured with the following specifications:

- CPU: Intel(R) Xeon(R) CPU @ 2.00GHz
- GPU: 2× NVIDIA Tesla T4 GPUs (16GB VRAM each)
- RAM: 32GB Memory
- Storage: 128GB

The software configurations are:

- Operating System: Ubuntu 22.04 LTS
- CUDA Version: 12.6
- Python: 3.11.11
- PyTorch: 2.6.0+cu124
- HuggingFace Transformers: 4.51.3
- SentenceTransformers: 3.4.1
- PEFT: 0.14.0
- spaCy: 3.8.5, with `en_core_web_sm` model for sentence segmentation
- Additional Python packages: HuggingFace datasets 3.6.0, scikit-learn 1.2.2, Pandas 2.2.3, SciPy 1.15.2, NumPy 1.26.4, Matplotlib 3.7.2, plotly 5.24.1, pytrec_eval 0.5.

A.2. Model-Specific Hyper-parameters

A.2.1. Sentence-level waterfall architecture

This module includes two consecutive stages: Evidence sentence extraction stage and SA sentence extraction stage. Both stages utilize two Bi-LSTM networks and two separate heads, which share configurations. LSTM configurations are:

- Aggregation Model: **LSTM** (Long Short-Term Memory)
- Input Size: 300 (FastText embedding)
- Hidden Size: 64
- Number of Layers: 1
- Dropout: 0.3

The optimization parameters are listed below:

- Optimizer: AdamW
- Learning rate: $5e-3$
- Weight decay: $1e-5$
- Loss function: `BCEWithLogitsLoss`
- Batch size: 128
- Epoch: 50

Output Classification Head and Training Setup:

- Head Structure: Multi-Layer Perceptron (MLP)
- Hidden Layer Size: 128
- Activation Function: `LeakyReLU`
- MLP Dropout: 0.3
- Evidence extraction head's output: 1
- SA extraction head's output: 4

A.2.2. Large language model

The following describes the finetuning configuration for the large language model:

- Base model: `t5-large`
- PEFT r: 8
- PEFT α: 32
- PEFT Dropout: 0.1

The optimization parameters are listed below:

- Optimizer: AdamW
- Learning rate: $1e-4$
- Weight decay: $1e-4$
- Loss function: `BCEWithLogitsLoss`
- Batch size: 16
- Epoch: 50