The Princeton Journal of Interdisciplinary Research
ISSN 3069-8200

Developing an Efficient Suicide-Detection System: How Small Models Compare to LLMs

Kaden Wu (author)[1]

[1]Canyon Crest Academy, San Diego, CA, United States of America

Abstract

As language evolves across generations and suicide rates continue to rise, automated detection of suicidal ideation in social media text has become increasingly important for early intervention. LLMs demonstrate strong accuracy, but their computational cost and large size limit real-world uses and implementation. This study evaluates whether smaller models can achieve comparable performance for suicide detection. A diverse set of models, ranging from thousands to trillions of parameters, was accessed across four datasets from Reddit and Twitter, two widely used social media platforms. All traditionally trained models were smaller than the LLMs. Results show that smaller models can achieve performance comparable to or exceeding LLMs while requiring substantially fewer computational resources. Model performance does not scale directly with parameter count, although large models tend to be more consistent across domains. These findings suggest that lightweight models can support deployable, real-time suicide detection systems, which are capable of flagging at-risk individuals and facilitating mental health intervention.

The Princeton Journal of Interdisciplinary Research
ISSN 3069-8200

Keywords: suicide ideation detection, large language models, social media analysis, lightweight models, out-of-distribution evaluation

Introduction

Suicide remains one of the leading causes of death worldwide. Each year, around 720,000 people die due to suicide and many more attempts (WHO, 2025). While suicide affects individuals across all age groups, it is particularly prevalent among teenagers. Because suicide is preventable, early identification and timely intervention are critical. However, suicide arises from a complex blend of emotional, social, psychological, and environmental factors, making detection and prevention challenging. A 2025 news report described a case in which a 16 year old disclosed suicidal thoughts to ChatGPT prior to his death by suicide (Hill, 2025)**.** The case raises broader questions about whether automated language systems can reliably detect and respond to high-risk expressions in real time, underscoring the urgent need for accurate and deployable suicide detection systems.

As communication increasingly shifts to digital platforms, social media and online forums have become primary venues through which individuals express emotional distress, including suicidal thoughts. This has motivated the development of automated systems capable of detecting and flagging suicidal ideation in posts. Such systems can analyze large volumes of text far faster than humans, enabling earlier identification and potentially fast intervention. However, there is a trade-off: humans are generally more accurate at detecting suicidal intent, whereas automated systems offer speed and scalability. Additionally, the task is complicated by linguistic nuances such as figurative language, sarcasm, and ambiguity. Words may carry multiple meanings, emotional expression does not always indicate suicidal

intent, and online "texting language," particularly among adolescents, evolves rapidly. These factors make accurate and reliable detection challenging.

Recent advances in machine learning have significantly improved performance on suicide detection tasks. However, there are still some crucial issues that exist with these models. Many existing solutions exhibit tradeoffs between efficiency and accuracy, and state-of-the-art models, including transformers and large language models (LLMs), often contain billions to trillions of parameters, requiring extensive computational resources for training, evaluation, and deployment. Even when highly accurate, the cost and size of such models can hinder real-world applicability, particularly when fast, real-time responses are required.

These considerations motivate the central question of this study: Can a simplified machine learning model reliably and accurately predict suicidal ideation in social media text? To address this, I examine the relationship between model size, architectural complexity, and generalization performance in suicide detection. By evaluating models ranging from Bag-of-Words classifiers to transformer-based architectures and state-of-the-art LLMs across both in-domain and out-of-distribution datasets, this work investigates whether performance scales with parameter count, how smaller fine-tuned models compare to general purpose LLMs, and how distributional shifts affect models of varying complexity.

Related Work

An expanding body of research has investigated the detection of suicidal ideation. Dzafic et al. (2025) constructed datasets and evaluated multiple models to assess their performance, suggesting that commonly used models may not always achieve the best results

for this task. Yang et al. (2025) proposed a hybrid model to classify social media posts based on varying levels of suicide risk severity. Similarly, Shimgekar et al. (2025) focused specifically on improving detection of early indicators of suicidal ideation, demonstrating the potential applicability of automated systems in real-world intervention scenarios. Bhuiyan et al. (2025) introduced another hybrid model that achieved strong and reliable classification accuracy. Several studies have also examined the role of large language models in suicide detection. Nguyen and Pham (2024) investigated the effects of fine-tuning LLMs to improve labeling accuracy, while Kim et al. (2025) explored suicidal risk detection using LLMs on Chinese-language data, highlighting the potential for expanding suicide detection systems to multilingual contextsOther research has focused on model efficiency and architectural design. Brown et al. (2023) evaluated the effectiveness of knowledge distillation and model compression techniques for improving computational efficiency without sacrificing performance. Earlier work by Joulin et al. (2016) demonstrated that relatively simple models can perform competitively with more complex deep learning classifiers in certain text classification tasks. Sankpal (2025) analyzed emotional changes within individual data samples, providing evidence of correlations between emotional patterns commonly observed in mental health discussions. Additionally, Hsieh et al. (2023) explored step-by-step distillation approaches, showing that smaller models can outperform larger LLMs when trained with appropriate techniques.

Together, these studies highlight both the potential and the challenges of applying machine learning to suicide detection. While many approaches rely on large, complex models, growing evidence suggests that smaller and more efficient models may achieve comparable performance when properly designed and trained.

The Princeton Journal of Interdisciplinary Research
ISSN 3069-8200

Dataset / Models

Table 1

Dataset Information

Name	Size	Distribution (Pos/Neg)
SuicideWatch	~348K	50/50
Suicidal Ideation - Reddit	~12400	53/47
Suicidal Tweets	~1800	37/63
Suicidal Ideation - General	~232K	50/50

Note. "Pos" refer to instances in which suicidal ideation was present in text, and "neg" refers to instances in which suicidal ideation was not present.

Specialized online Kaggle datasets were used in this study. These datasets include posts from Reddit and Twitter, two widely used social media platforms. All datasets were previously annotated for the presence or absence of suicidal ideation. For each evaluation, 2000 randomly selected samples were used, with the exception being Suicidal Tweets, in which case the entire dataset was used. Because the trained models were developed using SuicideWatch, the remaining datasets served as out-of-distribution (OOD) evaluation sets.

Six models were trained and evaluated: a fine-tuned BERT model, a fine-tuned DistilBERT model, a Bag-of-Words (BoW) feedforward neural network, an embedding-based feedforward neural network, a standard text convolutional neural network (CNN), and a CNN with a multilayer perceptron (MLP) classification head. For the trained models, 30% of Dataset 1 was randomly sampled, with an 80/20 split used for training and evaluation, respectively. Because suicide detection is a binary classification task, model outputs were restricted to two classes: 0 (no suicidal ideation) and 1 (suicidal ideation).

BERT

For the BERT model, input text was tokenized using the standard BERT tokenizer. The core architecture was based on the BERT-base model obtained from the Hugging Face Transformers library. A linear classification layer was added on top of the BERT encoder to perform the final binary classification task. The model was trained using a learning rate of 0.00002 (2e-5). Training the model required approximately three hours to complete.

DistilBERT

The DistilBERT model was similar to the BERT model, the only difference being that instead of a BERT tokenizer and model, I used a DistilBERT tokenizer and model. DistilBERT is smaller than BERT (~ 66M parameters compared to ~110M parameters) and more robust. Training the model required approximately 1 hour and 30 minutes to complete.

BoW

The BoW model consisted of a simple linear layer with no activation functions. The vocabulary of the BoW model was built from the training dataset (Dataset 1). The model was

trained using a learning rate of 0.001 (1e-3). Training the model required approximately 17 minutes to complete. The architecture is as follows:

Vocabulary → Classification

Embedding

The embedding model consisted of two layers: an embedding layer and a linear layer for classification. The embedding dimension used was 128. Similar to BoW, the embedding model also built its vocabulary on the training dataset. The model was trained using a learning rate of 0.001 (1e-3). Training the model required approximately 18 minutes to complete. The architecture is:

Vocabulary → Embedding → Classification

CNN

The CNN (convolutional neural network) model consisted of three convolutions: a convolution with kernel size 3, a convolution with kernel size 4, and a convolution with kernel size 5 (Kim, 2014). It used pretrained embeddings from DistilBERT. The output after each output is fed into a ReLU activation function, and then one-dimensional max pooling is applied to each layer. The outputs of each layer are concatenated and fed into a dropout layer. The resulting dropout output is fed into the linear layer for the final classifier. Each convolution layer had a possible 128 filters, so the entire model had 384 possible filters. The model was trained using a learning rate of 0.0001 (1e-4). Training the model required approximately 25 minutes to complete. The architecture is as follows:

Vocabulary → Embedding → (Convolution + Pooling) x3 → Classification

CNN + MLP

This model was the same as the original CNN model, except instead of a linear layer for the classifier, the dropout output is sent to a more complex classifier with two hidden layers, ReLU after each layer, and dropout with probability 0.5 after each layer before sending the output to the linear layer. The model was trained using a learning rate of 0.0001 (1e-4), and training the model required approximately 25 minutes to complete. The architecture is as follows:

Vocabulary → Embedding → (Convolution + Pooling) x3 → MLP (Linear, ReLU, Dropout, Linear, ReLU, Dropout) → Classification

LLMs

Various top tier LLMs were tested, including LLAMA, Grok 4.1, Gemini 2.5, and GPT5. These were tested in a similar way to the other models, excepting the fact that there was no training for these LLMs, only evaluation. These were tested using the OpenRouter API.

Hyperparameters

All of the trained models have learning rate, number of training epochs, batch size, max number of tokens per input, hidden size of models, and number of samples as hyperparameters. Changing any of these would vary the output and accuracy. As for the

LLMs, hyperparameters include how many tokens they can output. DataLoaders had a batch size of 32.

Experiment

Data Preprocessing

The datasets were downloaded directly from Kaggle and imported into CSV format using the Pandas library. Because machine learning models operate more efficiently on numerical data than categorical labels, the annotations were converted into binary values (0 and 1). The text inputs and corresponding labels were then extracted from the datasets for processing. A custom dataset class was implemented to tokenize the input text and organize the data into a format suitable for model training and evaluation.

Training

The Adam optimizer was used. Because suicide detection is formulated as a binary classification task, cross-entropy loss was used. All models were trained on 20 epochs. The dataset was split into batches for batch processing, and in each epoch all batches are analyzed.

Evaluation

For evaluation, 2000 randomly selected samples were used to test the performance of the models and LLMs. The accuracy, false positive, and false negative counts of each model/LLM were recorded.

Resources

All models were trained on A100 and L4 GPUs in Google Colab. Libraries included are listed as follows: PyTorch, Pandas, Transformers, KaggleHub, Matplotlib, TQDM, OpenAI.

Prompting

For evaluation on the LLMs, the prompt was structured in the following way:

Output only 0 or 1.

0 = no suicidal indicators

1 = suicidal indicators

Text: {text}

Output:

If the output was either 0 or 1, it was recorded as a valid prediction. Outputs that did not match these values were classified as formatting errors. In practice, formatting errors were negligible and had minimal impact on the overall evaluation.

Few Shot

On Dataset 1, LLMs were evaluated using both few-shot and zero-shot prompting. In the few-shot setting, models are provided with several example inputs and outputs before completing the task, whereas in the zero-shot setting, models are asked to perform the task without any prior examples. Few shot prompting was implemented by providing the LLMs

with labeled examples of suicidal and non-suicidal text before evaluating them on the dataset.

The few-shot evaluation procedure was conducted as follows:

Output only 0 or 1.

Examples: {examples}

0 = no suicidal indicators

1 = suicidal indicators

Text: {text}

Output:

Results

The models analyzed span multiple orders of magnitude in parameter count, ranging from thousands to trillions of parameters.

Table 2

Model Sizes

Model	# of Parameters
BERT	~110M
DistilBERT	~66M
BoW	~404K

Embedding	~26M
CNN	~24M
CNN + MLP	~25M`
LLAMA	~70B
Grok	~1T
Gemini	~500B
GPT5	~2T

0 represents no suicidal ideation, while 1 represents suicidal ideation. A false positive is defined as when a model outputs a 1 instead of a 0, and a false negative is defined as when a model outputs a 0 instead of a 1.

Table 3

SuicideWatch (Dataset 1, Zero Shot)

Model	Accuracy	FP	FN
AVG	93.8	97	63
BERT	97.3	—	—
DistilBERT	96.9	—	—

BoW	92.0	—	—
Embedding	92.7	—	—
CNN	95.6	—	—
CNN + MLP	95.6	—	—
LLAMA	88.0 (ZS)	196	39
Grok	93.4 (ZS)	49	84
Gemini	92.6 (ZS)	79	73
GPT5	94.0 (ZS)	64	56

Note. AVG = Average. False Positive = FP, False Negative = FN, ZS = Zero Shot.

The trained models performed better than the LLMs, which is expected because the trained models were trained on this dataset. With an average accuracy of 93.8%, the models overall did pretty well on this dataset. BERT performed the best, presumably due to its extensive training and complex transformer-based architecture. DistilBERT had a similar performance to BERT despite its almost 40% reduction in size (from ~110M to ~66M parameters). The BoW and embedding models performed worse than the two transformer based models. However, they still performed acceptably, despite being much smaller than other models presented. The CNN models performed in the middle of the other models, which is to be expected given that their sizes are in the middle of the trained models as well. LLMs were simply prompted without any tuning. GPT5 performed the best out of the LLMs,

with an accuracy of 94%. The worst performing LLM by far was LLAMA, revealing that a large number of parameters does not necessarily mean a better performance. Additionally, false positives and false negatives were also analyzed. Across all models, false positives occurred a lot more than false negatives. This could partially be due to the fact that LLMs are not used for specific suicidal ideation classification, but are instead used for general tasks. Nevertheless, generalization of emotional language could possibly lead to high false positives.

Table 4

SuicideWatch (Dataset 1, Few Shot)

Model	Accuracy	FP	FN
AVG	93.8	116.5	41.3
BERT	97.3	—	—
DistilBERT	96.9	—	—
BoW	92.0	—	—
Embedding	92.7	—	—
CNN	95.6	—	—
CNN + MLP	95.6	—	—

LLAMA	89.2 (FS)	202	14
Grok	93.6 (FS)	53	75
Gemini	92.6 (FS)	99	49
GPT5	93.1 (FS)	112	27

Note. AVG = Average. False Positive = FP, False Negative = FN, FS = Few Shot.

Overall, the averaged accuracy across models remained the same at 93.8%. Few-shot prompting only affected the LLMs, as they were the only models that received prompts. Interestingly, GPT-5 performed worse with few-shot prompting compared to zero-shot, in contrast to the other LLMs. While few-shot generally produced a small increase in accuracy, it also resulted in more false positives than false negatives. Although overall accuracy did not change substantially, the reduction in false negatives indicates that few-shot prompting can make LLMs less likely to miss suicidal posts. The effect of few-shot prompting was most pronounced for GPT-5, nearly doubling false positives while halving false negatives. This shift may have contributed to GPT-5's decreased performance, potentially due to overfitting and misidentification of patterns.

Table 5

Suicidal Ideation - Reddit (Dataset 2)

Model	Accuracy	FP	FN

AVG	86.0	187.8	91.8
BERT	84.4	140	173
DistilBERT	84.5	214	96
BoW	86.4	173	99
Embedding	85.0	198	102
CNN	84.4	228	84
CNN + MLP	85.7	196	90
LLAMA	86.5	238	32
Grok	86.7	173	94
Gemini	88.4	146	86
GPT5	88.3	172	62

Note. AVG = Average. False Positive = FP, False Negative = FN.

Dataset 2 is the first OOD (out-of-distribution) dataset that was evaluated. A substantial decline was observed in the overall accuracy of the models, suggesting that the transition from an in-distribution dataset to an out-of-distribution dataset is not smooth. Among the trained models, the best-performing model was the BoW model, achieving an accuracy of 86.4%. However, the overall performance of the trained models decreased

considerably, indicating that fine-tuning may improve accuracy on in-distribution data but can reduce generalization to broader datasets. Interestingly, DistilBERT outperformed BERT on this dataset, further suggesting that a greater number of parameters does not necessarily lead to better performance. Although BERT and DistilBERT share very similar architectures, DistilBERT appears to be more robust and adaptable to different datasets. Notably, DistilBERT is also more cautious than BERT, producing more false positives than false negatives. The two CNN models performed at a comparable level relative to other models in its domain. Despite their significantly smaller size compared with BERT and DistilBERT, the CNN models still achieved strong performance. Among the LLMs, LLAMA produced the fewest false negatives, indicating that it is the most cautious when detecting suicidal ideation. Despite this, it did not achieve the highest accuracy due to a large number of false positives, indicating lower precision. The traditionally trained models exhibited high numbers of false negatives, which substantially reduces overall accuracy. The BoW model performed relatively well, achieving results comparable to some of the largest LLMs. Finally, the generally poor performance across all models suggests that the semantic and syntactic linguistic patterns present in this dataset may differ from those that these models are typically trained to detect.

Table 6

Suicidal Tweets (Dataset 3)

Model	Accuracy	FP	FN

AVG	77.3	52.9	351.5
BERT	71.0	83	435
DistilBERT	68.9	115	441
BoW	77.4	20	383
Embedding	76.8	35	379
CNN	72.7	104	383
CNN + MLP	73.5	69	404
LLAMA	86.3	59	185
Grok	79.9	14	345
Gemini	83.1	18	283
GPT5	83.8	12	277

Note. AVG = Average. False Positive = FP, False Negative = FN.

Dataset 3 is the second OOD dataset evaluated. Compared to the other two datasets, this one exhibited the lowest overall model performance, making it the most challenging. The traditionally trained models experienced a substantial drop in accuracy, with the BoW model emerging as the top performer. Despite having the fewest parameters, it outperformed models with larger parameter counts. In general, models with more parameters tended to be more

cautious than those with fewer parameters.The increased difficulty of Dataset 3 can be attributed to the larger number of false negatives, likely due to the presence of more subtle indicators in the text. Interestingly, LLaMA performed the best on this dataset, while the other models struggled, suggesting that LLaMA may have greater familiarity with Twitter data than with Reddit data. Among the four LLMs, LLaMA also has the fewest parameters, further supporting the notion that accuracy does not necessarily scale with model size. Most models tended to miss false negatives, highlighting that encouraging models to be more cautious when classifying text could improve their accuracy. The superior performance of LLMs relative to traditionally trained models on this dataset also suggests that Twitter posts exhibit semantics that differ from those on Reddit.

Table 7

Suicidal Ideation – General Dataset 4

Model	Accuracy	FP	FN
AVG	89.4	139.1	73.3
BERT	91.6	23	145
DistilBERT	91.6	93	75
BoW	88.1	38	201
Embedding	86.1	23	256

CNN	92.1	97	62
CNN + MLP	91.4	61	120
LLAMA	75.8	476	8
Grok	92.8	67	78
Gemini	88.5	194	36
GPT5	91.4	144	29

Note. AVG = Average. False Positive = FP, False Negative = FN.

Dataset 4 is the final out-of-distribution (OOD) dataset evaluated. This dataset draws from a wide variety of social media platforms, making it the most generalizable. Overall, model accuracies were higher, indicating that most models generalize reasonably well across a broad domain. The notable exception was LLaMA, which performed significantly worse than expected. When generalized, BERT and DistilBERT achieved identical overall accuracy, with the primary difference lying in the distribution of false positives and false negatives. BERT exhibited considerably more false negatives than DistilBERT, suggesting that it was less cautious in detecting suicidal indicators. Consequently, DistilBERT performed better in this scenario, as it was able to identify more indicators than BERT. Across all models, the best-performing model was Grok, with the CNN model close behind. Notably, the CNN model captured substantially more suicidal indicators than its counterpart using an MLP classifier. Between the BoW and embedding models, false positives were

nearly identical, but the BoW model detected significantly more suicidal indicators than the embedding model, despite the latter having a higher parameter count.

Analysis

It should be noted that minor errors occurred during dataset evaluation. While these errors were not substantial, they demonstrate that in certain cases, LLMs may refuse to follow instructions even when explicitly prompted, highlighting a potential safety concern. All models were evaluated on the same amount of testing data. Overall, the traditionally trained models achieved an accuracy of 85.88%, while the LLMs slightly outperformed them with 88.40% accuracy. The following table summarizes these averages:

Table 8

Average Accuracy For Each Model

Model	Accuracy
BERT	86.0
DistilBERT	85.5
BoW	86.0
Embedding	85.1
CNN	86.2

CNN + MLP	86.5
LLAMA	85.2
Grok	86.8
Gemini	88.2
GPT5	89.4

These averages are only rough estimates and may not serve as ideal indicators, as the traditionally trained models were trained on Dataset 1, whereas the LLMs were trained on a variety of datasets. Comparing BERT and DistilBERT, BERT performed slightly better overall, although their accuracy was largely comparable across all datasets. However, DistilBERT produced more false positives and fewer false negatives than BERT, indicating that it detected more suicidal indicators while adopting a more cautious approach. This characteristic may make DistilBERT more suitable for deployment in real-world settings.

Between the BoW and embedding models, BoW exhibited a simpler architecture but achieved higher overall performance. The embedding model displayed a higher false-negative rate, suggesting reduced sensitivity to suicidal indicators. Both CNN models performed well, with the CNN + MLP model achieving the highest accuracy among all trained models. Among the LLMs, LLaMA appeared overly conservative in its classification behavior, generating more false positives than false negatives across most datasets. This cautious approach explains its strong performance on Dataset 3 where many other models missed potential indicators, but also contributed to lower accuracy on the remaining datasets,

making it the lowest-performing LLM in those cases. Grok performed comparably to the trained models, while Gemini, despite having roughly half the number of parameters as Grok, achieved higher performance. GPT-5 was the most consistent model overall, demonstrating strong and reliable performance across nearly all datasets.

Conclusions & Future Work

Simpler, smaller models can perform just as well as LLMs with far more parameters. When in-domain data is available, smaller models are often a strong choice, whereas LLMs may perform better on out-of-domain data. Additionally, model performance does not necessarily scale with parameter count or complexity; models that are only a fraction of the size of LLMs were able to achieve comparable results.

These smaller models also offer significant practical advantages. They can be easily integrated into real-world systems such as social media monitoring tools, mental health applications, or crisis intervention platforms to detect suicidal ideation in real time with minimal loss of accuracy. Their efficiency reduces computational and optimization requirements, making them more cost-effective and reliable for deployment. For real-world implementation, training smaller models on in-domain datasets may therefore be the most effective approach.

Beyond suicide detection, the success of these smaller models suggests potential applicability to other domain specific tasks, such as detecting depression, self-harm, or other mental health indicators. Furthermore, some tasks may not involve highly complex relationships and could be effectively modeled using simpler architectures. Future work may

explore developing lightweight models that surpass LLM performance in specialized domains, as well as investigate new challenges associated with deploying these systems in real-world environments.

References

Bhuiyan, M. I., Kamarudin, N. S., & Ismail, N. H. (2025). *Detecting suicidal ideation in text with interpretable deep learning*. arXiv. https://arxiv.org/abs/2511.08636

Brown, N., Williamson, A., Anderson, T., & Lawrence, L. (2023). *Efficient transformer knowledge distillation: A performance review*. arXiv. https://arxiv.org/abs/2311.13657

Dzafic, A., Kavut, M., & Ulya, B. (2025). *Rethinking suicidal ideation detection*. arXiv. https://arxiv.org/abs/2507.14693

Hill, K. (2025). *A teen was suicidal. ChatGPT was the friend he confided in.*

Hsieh, C.-Y., Li, C.-L., Yeh, C.-K., Nakhost, H., Fujii, Y., Ratner, A., Krishna, R., Lee, C.-Y., & Pfister, T. (2023). *Distilling step-by-step! Outperforming larger language models with less training data and smaller model sizes*. arXiv. https://arxiv.org/abs/2305.02301

Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). *Bag of tricks for efficient text classification*. arXiv. https://arxiv.org/abs/1607.01759

Kim, J.-W., Oh, W., Yoon, H., Yoon, S.-H., Kim, D.-J., Lee, D.-H., Lee, S.-Y., & Yang, C.-M. (2025). *Language-agnostic suicidal risk detection using large language models*. arXiv. https://arxiv.org/abs/2505.20109

Kim, Y. (2014). *Convolutional neural networks for sentence classification*. arXiv. https://arxiv.org/abs/1408.5882

Nguyen, V., & Pham, C. (2024). *Leveraging large language models for suicide detection on social media with limited labels*. arXiv. https://arxiv.org/abs/2410.04501

Sankpal, S. (2025). *Detecting emotion drift in mental health text using pre-trained transformers*. arXiv. https://arxiv.org/abs/2512.13363

Shimgekar, S. R., Zhao, R., Goyal, A., Rodriguez, V. J., Bloom, P. A., Sundaram, H., & Saha, K. (2025). *Detecting early and implicit suicidal ideation via longitudinal signals on social media*. arXiv. https://arxiv.org/abs/2510.14889

World Health Organization. (2025). *Suicide*. https://www.who.int/news-room/fact-sheets/detail/suicide

Yang, Z., Leonard, R., Tran, H., Driscoll, R., & Davis, C. (2025). *Detection of suicidal risk on social media: A hybrid model*. arXiv. https://arxiv.org/abs/2505.23797