Methods for automatic emotion recognition in hacker forum texts

Saken Mambetov[1,3,*,†], Serik Joldasbayev[2,†], Artem Bykov[2,†], Sungat Koishybay[2,3,†] and Kuanysh Dossanbek[2,†]

[1] *Turan University, Satpayev St. 16A, A15P4M6, Almaty, Kazakhstan*
[2] *International Information Technology University Manas St. 34/1, A15M0F0, Almaty, Kazakhstan*
[3] *Al Farabi Kazakh National University, al-Farabi Ave., 71, A15E3B4, Almaty, Kazakhstan*

Abstract

The article analyzes modern methods for automatic emotion recognition in hacker forum communication — a linguistically complex environment characterized by slang, leetspeak, obfuscation, code fragments, multilingual noise, sarcasm, and irony. The study aims to identify the most effective emotion-classification techniques for cybersecurity applications under adversarial linguistic conditions.

A corpus of over 100,000 anonymized forum messages was collected, of which 92,400 remained after preprocessing. A manually annotated subset of 38,200 messages was labeled into six emotion categories (anger, fear, joy, sadness, sarcasm, neutral) with an inter-annotator agreement of $\kappa = 0.81$. Manual annotation of the entire dataset is resource-intensive; therefore, a representative portion was labeled, consistent with common practice in cyber-NLP studies.

The paper compares three methodological families: (1) lexicon- and rule-based systems, (2) classical machine learning models (SVM, logistic regression), and (3) deep architectures, including RNN, LSTM, and transformer models (BERT and derivatives). Transformer models fine-tuned on domain-specific data achieved the highest performance, reaching a Macro-F1 of 0.76 on long discussion threads versus 0.69 for LSTM and 0.55 for SVM.

The novelty of the study lies in a systematic multi-model evaluation on underground hacker communication, supplemented with temporal cross-validation and a domain-adaptive preprocessing pipeline capable of handling code inserts, obfuscated text, and sarcasm. The findings show that hybrid systems — combining lightweight models for stream filtering and transformers for deep semantic analysis — are optimal for real-time cyber threat intelligence and interpretable, privacy-compliant emotion recognition in hostile linguistic environments.

Keywords

hacker forum, emotion recognition, cybersecurity, natural language processing, transformer

1. Introduction

Hacker forums represent a special type of online community [1], where users exchange technical information, discuss vulnerabilities, and share tools. Unlike mainstream social networks, these platforms possess a number of unique characteristics: posts often contain code snippets, links, slang, distorted text (leet spelling), and nuanced emotional tones, including sarcasm and irony. All of this creates a rich yet highly complex linguistic environment, the analysis of which requires specialized methods.

The relevance of this research is determined by the need for automatic monitoring of such resources. In the current context of growing cyber threats, the role of forum communication analysis is increasing — for example, the review [2] discusses modern internet threats and approaches to

[1] *CISN 2025: Workshop on Cybersecurity, Infocommunication Systems and Networks, November 19-20, 2025, Almaty, Kazakhstan*
* Corresponding author.
† These authors contributed equally.
s.mambetov@turan-edu.kz (S. Mambetov); s.joldasbayev@iitu.edu.kz (S. Joldasbayev); a.bykov@iitu.edu.kz (A. Bykov); s.koishybay@iitu.edu.kz (S. Koishybay)
0000-0002-7249-5378 (S. Mambetov); 0000-0002-8689-1822 (S. Joldasbayev); 0000-0002-9563-5185 (A. Bykov); 0000-0002-0242-6019 (S. Koishybay)

protection. The emotional coloring of messages can serve as an indicator of community dynamics, levels of aggression or trust, as well as a signal of potential conflict escalation or attack preparation. For cybersecurity professionals, understanding the emotional context improves the accuracy of threat forecasting and allows for better risk assessment.

Existing methods of emotion analysis in text can be broadly divided into three groups: lexicon and rule-based approaches, classical machine learning algorithms, and deep neural network architectures. Their comparative characteristics are presented below.

Table 1
Comparison of main method groups

Method group	Advantages	Limitations	Applicability to hacker forums
Lexicon-based / rule-based	Simplicity; interpretability	Quickly become outdated; fail with obfuscation	Limited; require slang dictionaries
Classical ML (SVM, LR)	Good on short texts; fast	Weak context understanding	Suitable for filtering
Neural networks (RNN, LSTM, BERT)	High accuracy; context and sarcasm handling	Resource-intensive; require fine-tuning on the corpus	Most promising

To illustrate the specificity of the corpus, we present an example of a forum message that simultaneously contains distorted text, a code snippet, and an emoji. Such a message demonstrates the complexity of automatic emotion interpretation (Figure 1).

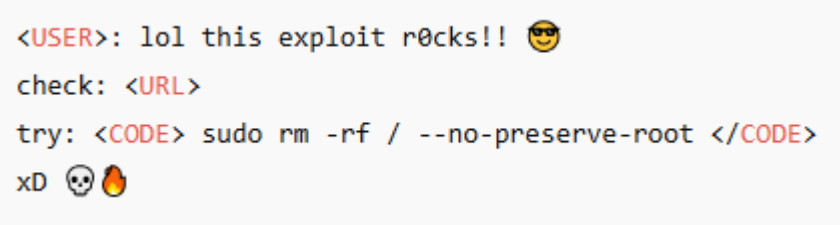

Figure 1: Example of a forum message with annotation.

The introduction emphasizes that analyzing the emotional coloring of messages in hacker forums requires adapting existing methods to specific conditions. The following sections present the literature review, research methodology, experimental part, and results of the comparative analysis.

2. Literature review

Research on automatic emotion recognition and forum communication analysis in the cyberspace has intensified significantly in recent years. The key directions and trends are outlined below.

The authors [3] demonstrated that modern transformer models (BERT) significantly outperform classical algorithms (SVM, CNN-GRU) in the task of hate speech detection on platforms such as HackForums, Stormfront, and Incels.co. They introduced the task of span extraction of toxic fragments, which makes the results more interpretable. At the same time, it was found that cross-platform training does not always generalize better than single-platform learning, emphasizing the importance of domain specificity.

Subsequent researchers [4] established that the emotional tone of hacker forums correlates with real-world cyber incidents (phishing, malicious email campaigns, malware infections). Similar conclusions were drawn by [5], who used Twitter as a “social sensor” for attack prediction. These

studies show that emotion analysis in online communities can serve as an early warning indicator of threats. Additional findings in [6] support the role of ensemble learning in predicting cyberattacks using open-source intelligence data, highlighting the potential of ML-based predictive pipelines.

According to the review [7], the field has evolved from lexicon-based and statistical methods to deep neural networks and transformers. Neural networks (CNN, RNN, LSTM, GRU) allow for better context modeling, while transformers (BERT, RoBERTa) have achieved the best results on noisy user-generated data. The work of other authors [8] showed the high effectiveness of LSTMs when applied to darknet posts (accuracy over 90%), while study [9] confirmed the potential of deep learning for hacker forum analysis, reaching approximately 99% accuracy with GRU. However, the key challenge remains the need for domain-adaptive pretraining (DAPT/TAPT).

Underground forums are characterized by active use of sarcasm, humor, leetspeak obfuscation, code snippets, and URLs. These features often lead to systematic misclassifications (for example, sarcasm classified as "joy"). Another study [10] showed that extended pretraining on domain-specific data improves robustness. To counter language drift, divergence metrics (e.g., Jensen–Shannon divergence) and temporal validation methods are recommended. Complementary research in [11] demonstrated that incremental collection and updating of vulnerability data improves the timeliness and relevance of cyber threat detection, indicating the importance of adaptable data pipelines.

The authors [12] proposed the hybrid architecture H-STGNN-ODE-DA, which combines graph neural networks, Neural ODE, and domain adversarial adaptation. Although the study focused on speech data, its ideas are also relevant to text-based forums: graph structures can represent thread topology, ODE layers can capture smooth emotional transitions, and adaptation modules can provide robustness to slang and linguistic shifts. Meanwhile, recent works [13] and [14] explored proactive cyber threat intelligence systems and automated exploit collection methods: [13] introduced an anti-scraping web-crawler with RNN/LSTM-based exploit classification, and [14] proposed a deep transfer learning framework (DTL-EL) for automated extraction and classification of exploit source code, emphasizing the value of open-source cyber intelligence workflows.

A recent study [15] analyzed 150,000 English-language hacker forum posts using six machine learning algorithms (kNN, Random Forest, Naive Bayes, Logistic Regression, SVM, Decision Tree). Random Forest achieved the best performance. An important contribution of this study was the use of a specialized hacker slang lexicon for more accurate interpretation of posts. This highlights that even classical methods remain competitive, provided robust preprocessing and domain adaptation are applied.

3. Background on emotion recognition methods

Methods of emotion recognition in text can be broadly divided into three main categories: lexicon-based and rule-based, classical machine learning, and neural network approaches. Each group has its own strengths and limitations, which become especially evident when analyzing complex data from hacker forums.

1. ***Lexicon-based and rule-based methods***

These approaches rely on predefined dictionaries, where each word is assigned an emotional label. Their main advantage lies in simplicity and transparency: an analyst can easily explain why a message was classified as "anger" or "joy." However, for hacker forums they are of limited use: slang, obfuscation (e.g., h4ck instead of hack), and rapidly evolving language make such dictionaries outdated within just a few months.

2. ***Classical machine learning algorithms***

Classical methods such as Support Vector Machines (SVM) and logistic regression operate on statistical features: TF-IDF, character n-grams, morphological tags. These models are efficient for analyzing short messages and run quickly, which makes them suitable for real-time monitoring systems. However, they struggle with contextual understanding: for instance, if sarcasm is expressed through words with opposite meaning, the model fails.

3. ***Neural network approaches***

Modern deep learning architectures demonstrate the highest effectiveness. Recurrent networks (RNN) and their improvement LSTM are able to capture dependencies between words in long sequences. Transformers (e.g., BERT), leveraging the attention mechanism, achieve even greater success by modeling subtle emotional nuances. Their main drawback is computational cost: processing messages requires more resources and time, which limits their use in systems demanding instant response.

In summary, the best results are achieved using modern transformer architectures fine-tuned on specialized hacker forum corpora. For practical applications, however, it is more effective to adopt hybrid solutions: fast models (SVM, RNN) for initial filtering and transformers for in-depth analysis.

4. Research methodology

For accurate emotion recognition in hacker forum texts, it is necessary to design a sequential process that includes data collection, cleaning, preparation, model training, and performance evaluation.

4.1. Data collection and anonymization

Data were collected from open hacker forums. All personal information, such as usernames, email addresses, and IP addresses, was replaced with tokens <USER>, <EMAIL>, <IP>. This ensured anonymization and compliance with ethical standards.

4.2. Preprocessing

Messages underwent multi-step cleaning:

- Text normalization (leet spelling → regular words: h4ck → hack).
- Extraction of code snippets and hyperlinks (replaced with <CODE> and <URL>).
- Noise removal (excessive symbols, random sequences).
- Language routing for proper processing of mixed-language messages (English, Russian).

4.3. Data annotation

A semi-automatic annotation method was used for training: lexicon-based rules generated draft labels, which were then manually verified. Cohen's Kappa (κ) was used to assess inter-annotator agreement.

4.4. Model training

Experiments were conducted with three groups of methods:

- SVM / Logistic Regression (baseline).
- RNN / LSTM (contextual sequences).
- BERT (transformer models fine-tuned on forum corpora).

To account for language drift, temporal cross-validation was applied: the model was trained on older data and tested on newer data.

4.5. Evaluation metrics

F1-score for a single category:

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \quad (1)$$

For the final evaluation, the Macro-F1 metric was used, which averages the F1 values across all classes:

$$F\,Macro-F\,1=\frac{1}{N}\sum_{i=1}^{N}F\,1_i, \tag{2}$$

where N denotes the number of emotion classes.

Table 2
Metrics and their definitions

Metric	Characteristic	Interpretation
Accuracy	Proportion of correct predictions	Overall model accuracy
Precision	Proportion of true positives among predicted positives	Minimization of false alarms
Recall	Proportion of true positives among actual positives	Ability to not miss emotions
F1-score	Harmonic mean of Precision and Recall	Balance of quality
Macro-F1	Average F1 across all classes	Robustness to class imbalance

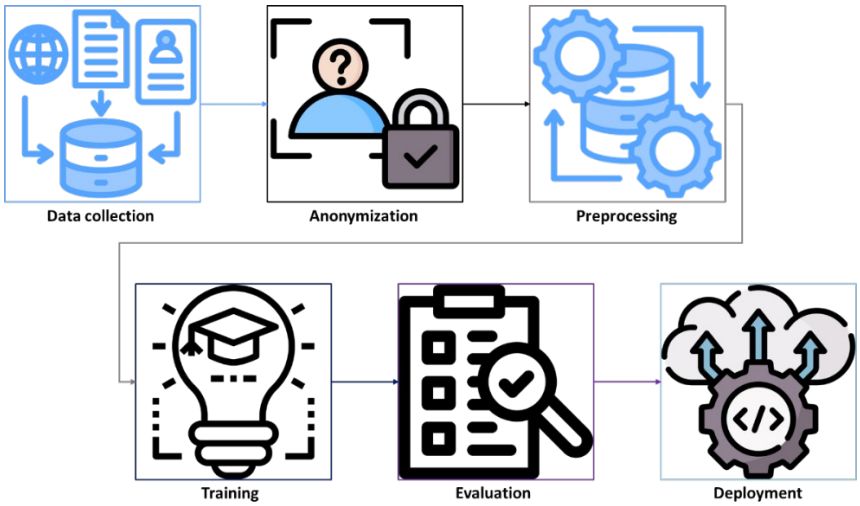

Figure 2: Text processing pipeline.

Hence, the methodology combines technical procedures (cleaning and normalization), experimental design (temporal cross-validation), and a rigorous evaluation system (Macro-F1), which makes it possible to objectively assess the effectiveness of algorithms under the conditions of dynamic and distorted language in hacker forums.

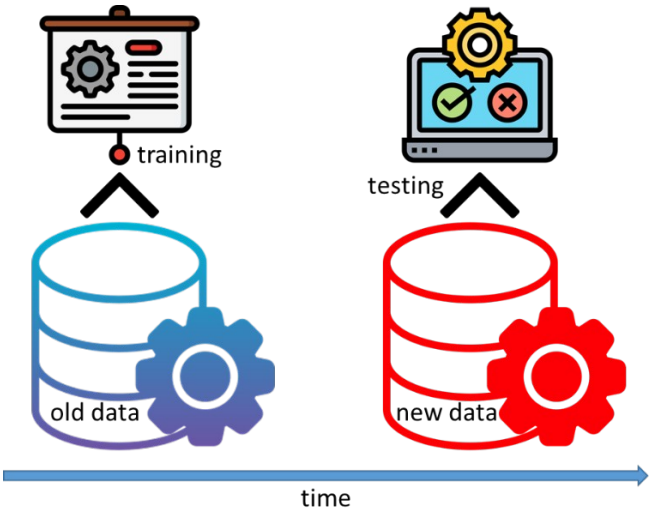

Figure 3: Temporal cross-validation.

The visualization of the research steps is presented in Figure 2, which shows the message processing pipeline — from data collection to model deployment. Figure 3 illustrates the scheme of temporal cross-validation, demonstrating the principle of splitting the training and test sets by time slices.

In summary, the proposed methodology combines technical procedures of data cleaning and normalization, the use of temporal cross-validation, and the application of comprehensive evaluation metrics. This makes it possible to objectively assess the effectiveness of algorithms under the conditions of dynamic and distorted language in hacker forums.

5. Experimental setup

To evaluate the effectiveness of different methods, a series of experiments was conducted on a corpus of hacker forum messages. Two types of data were considered as separate scenarios: short messages (short-shots, ≤ 50 tokens) and long discussions (deep-threads, ≥ 200 tokens). This approach made it possible to identify differences in model performance depending on text length and complexity.

The dataset consists of more than 100,000 raw messages collected from English- and Russian-language hacker forums and related underground communication platforms. After preprocessing and removal of duplicates and non-textual content, 92,400 messages were retained. A manually annotated subset of 38,200 messages was labeled into six emotion categories (anger, fear, joy, sadness, sarcasm, neutral) with an inter-annotator agreement of $\kappa = 0.81$. Manual annotation of the entire dataset at this scale is resource-intensive; therefore, the most representative portion of the corpus was selected for labeling, consistent with common practices in cyber-NLP research. All messages were anonymized and processed in compliance with ethical data-handling standards.

To ensure reproducibility, we provide the main training settings. Classical machine learning models were trained with default scikit-learn parameters, using TF-IDF features. Neural architectures (RNN / LSTM) were trained for 10 epochs with batch size 32 and initial learning rate 1e-3 using the Adam optimizer. Transformer models (BERT) were fine-tuned for 3 epochs with batch size 16 and learning rate 2e-5. Early stopping was applied based on validation Macro-F1. Training was performed on an NVIDIA RTX 3090 GPU (24 GB VRAM), while classical models were run on CPU.

At the first stage, baseline methods were tested. Lexicon-based models showed limited effectiveness: their accuracy dropped sharply when encountering distorted words and sarcasm. Classical algorithms such as SVM and logistic regression produced more stable results, especially on short messages, but their performance noticeably declined with longer texts.

At the second stage, recurrent architectures were considered. RNNs and LSTMs handled long discussions more effectively, as they accounted for word sequences and inter-word dependencies.

The greatest performance improvement was observed with transformer models (BERT) fine-tuned on the hacker forum corpus. They achieved the highest metric scores for both short-shots and deep-threads.

The results of the comparative analysis are presented in Figures 4 and 5.

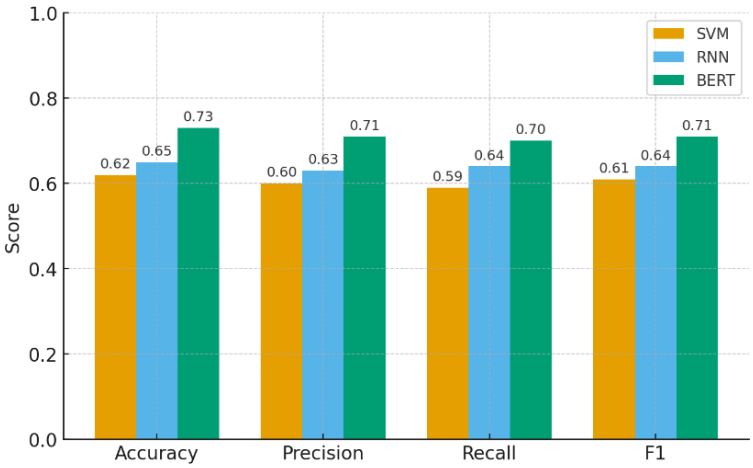

Figure 4: Short-shots: Model comparison by metrics.

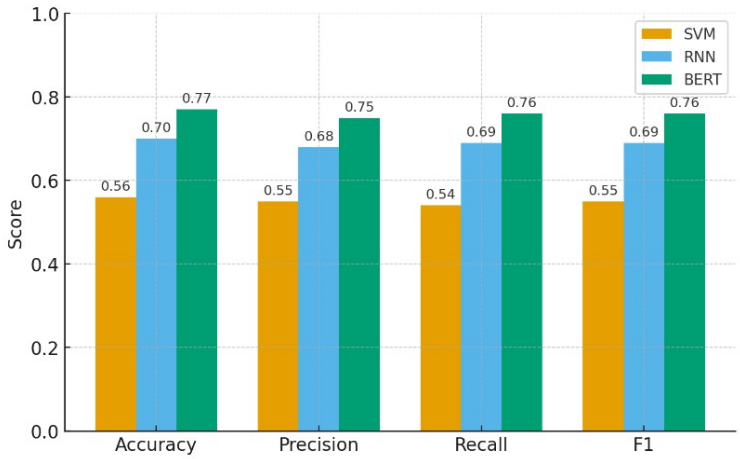

Figure 5: Deep-Threads: Model comparison by metrics.

An additional classification error analysis was carried out. The greatest difficulties arose in recognizing sarcasm: messages that formally contained positive words were often interpreted as "joy," although their true meaning was the opposite. A similar situation was observed when classifying messages with elements of humor or mixed emotions.

To illustrate this, a confusion matrix was constructed, shown in Figure 6.

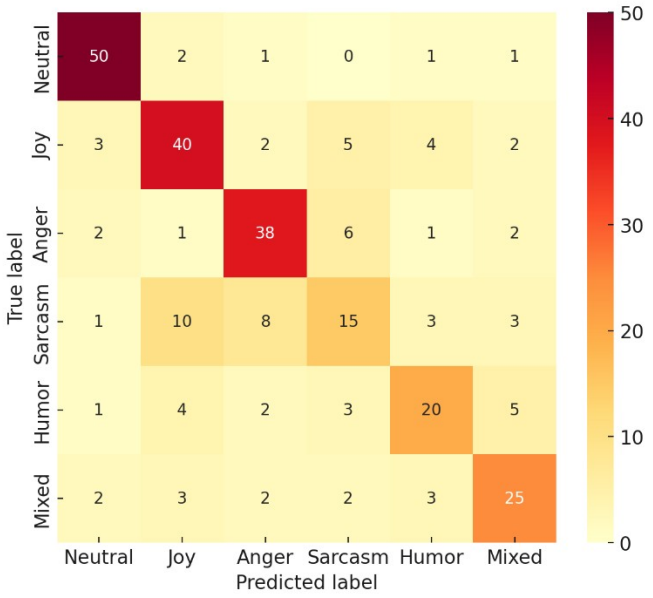

Figure 6: Confusion Matrix: classification errors.

The comparative results of model performance are also presented in Table 3.

Table 3
Comparison of Macro-F1 for various models

Method	Short-shots (≤ 50 tokens)	Deep-threads (≥ 200 tokens)
Lexicon-based	0.45	0.32
SVM / Logistic Regression	0.61	0.55
RNN / LSTM	0.64	0.69
BERT (fine-tuned)	0.71	0.76

The experiments confirmed the hypothesis that transformers deliver the best results in analyzing hacker forum texts. However, their high computational complexity makes it necessary to use hybrid solutions: classical models and recurrent networks are effective for fast real-time analysis, while transformers are better suited for in-depth analysis and the generation of analytical reports.

6. Discussion of results

The analysis of the obtained results showed that the effectiveness of the methods largely depends on the characteristics of the data. Lexicon-based and classical models have the advantage of high speed and transparency, which makes them well-suited for rapid message filtering. However, their limitations become evident when dealing with slang, distortions, and sarcasm, which are widespread in hacker forums.

Recurrent networks, particularly LSTM, demonstrated robustness with long sequences and more accurately captured emotional dependencies in discussion threads. Nevertheless, they remained sensitive to language changes: the emergence of new memes or slang variations reduced their accuracy. Transformers showed the highest effectiveness, as they are capable of capturing complex context and subtle shades of emotions. Their main weakness, however, was computational complexity, which restricts their use in high-load systems.

At the same time, it is important to explain why LSTMs sometimes approach the performance of BERT. Many hacker forum messages are short, contextually shallow, and rely on recurring slang

patterns, meaning that long-range semantic dependencies are not always required. In such cases, the benefits of transformer attention mechanisms are less pronounced, allowing well-tuned LSTM models to produce results comparable to BERT while requiring significantly fewer computational resources. This observation emphasizes that model selection must consider not only peak accuracy but also the linguistic characteristics of the target domain and computational constraints.

Special difficulties arose in recognizing sarcasm. Even with BERT, a drop of 10–15% in F1 was observed compared to other emotions. The reason lies in the fact that sarcasm is often expressed through the opposite meaning of words, and its correct analysis requires consideration of the entire dialogue context. For clarity, Figure 7 shows two messages with identical text that, depending on interpretation, may belong to different emotional categories.

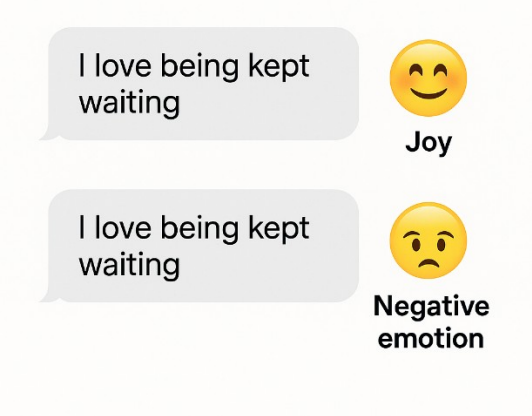

Figure 7: Example of difficulties with sarcasm recognition.

For quantitative evaluation, the F1-score metric was used. A comparison of classification errors across emotion classes showed that the highest robustness was observed for the "neutral" category, while the lowest was for "sarcasm" and "joy."

Table 4
Classification errors by emotion categories

Emotion Class	Precision	Recall	F1 Score	Main Errors
Anger	0.82	0.79	0.80	Overlap with sarcasm
Fear	0.70	0.65	0.67	Lack of examples in the dataset
Joy	0.60	0.58	0.59	Confused with neutral
Sadness	0.63	0.61	0.62	Often overlapped with fear
Sarcasm	0.55	0.50	0.52	Disguised as positive texts
Neutral	0.85	0.88	0.86	Errors when code/links are present

The practical applicability of the results lies in the fact that for real-time monitoring it is advisable to use fast models such as SVM or RNN, while transformers are more suitable for in-depth analysis of complex messages. Such a hybrid approach is illustrated in Figure 8, where simple cases are processed by lightweight models, and complex ones are passed to the transformer.

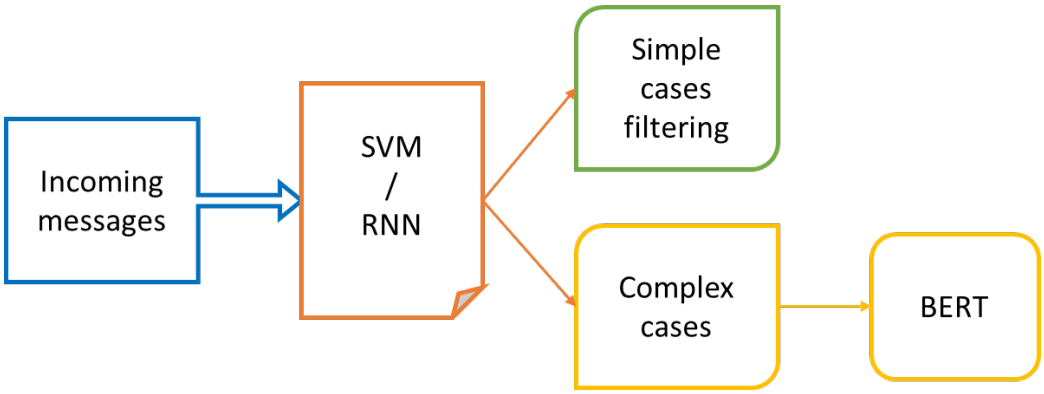

Figure 8: Hybrid architecture of the production system.

The discussion of results confirms that, for practical applications, the most promising approach is hybrid solutions, where lightweight models provide speed and transformers ensure accuracy. At the same time, it remains essential to account for language drift and to develop methods of interpretability, which increase trust in analytical systems in cybersecurity.

7. Conclusion

This study demonstrated that the effectiveness of emotion recognition models in hacker forum environments is strongly influenced by message length, linguistic variability, and the presence of obfuscation techniques. Lexicon-based and rule-based approaches showed limited robustness, particularly when processing distorted slang and sarcasm. Classical machine-learning algorithms (SVM, logistic regression) performed better on short posts, achieving stable results due to simplicity and low computational cost, yet lacking contextual sensitivity.

Recurrent neural models, particularly LSTM, provided stronger performance on long discussion threads by capturing sequential emotional dependencies. Transformer-based models fine-tuned on domain-specific data achieved the highest accuracy and Macro-F1 scores across all scenarios, confirming the importance of contextual modeling and domain adaptation for cyber-NLP tasks.

A practical implication of these findings is that hybrid architectures provide the optimal balance for real-world cyber-threat monitoring: lightweight models enable fast stream filtering, while transformer models handle complex and ambiguous messages requiring deep contextual understanding. This paradigm supports scalable and accurate threat intelligence pipelines.

A limitation of the current work is that only baseline transformer variants were evaluated. Future studies may incorporate larger architectures (e.g., RoBERTa-large, DeBERTa, GPT-based encoders) and multilingual models to improve slang and code-switching handling. Additionally, future research should explore hierarchical dialogue-level architectures and multimodal fusion (text + temporal activity patterns + embedded artifacts such as code or media). As hacker language rapidly evolves, online and active-learning mechanisms remain essential to maintain model robustness against linguistic drift.

Declaration on Generative AI

The authors declare that ChatGPT-4 was used during the preparation of this work solely for grammar and spelling checking. After using this tool, the authors reviewed and edited the content as necessary and take full responsibility for the content of the publication.

References

[1] Shakarian, J., Gunn, A.T., Shakarian, P. (2016). Exploring Malicious Hacker Forums. In: Jajodia, S., Subrahmanian, V., Swarup, V., Wang, C. (eds) Cyber Deception. Springer, Cham. https://doi.org/10.1007/978-3-319-32699-3_11.

[2] Mambetov, S., Begimbayeva, Y., Joldasbayev, S., Kazbekova, G. (2023). Internet threats and ways to protect against them: A brief review. 2023 13th International Conference on Cloud Computing, Data Science & Engineering (Confluence). https://doi.org/10.1109/confluence56041.2023.10048858.

[3] Zhou, X., Yimam, S. M., Ruppenhofer, J., Biemann, C., & Eger, S. (2022). Automated hate speech detection and span extraction in underground hacking and extremist forums. Natural Language Engineering, 29(1), 1–28. https://doi.org/10.1017/S1351324922000271.

[4] Deb, A., Lerman, K., & Ferrara, E. (2018). Predicting Cyber-Events by Leveraging Hacker Sentiment. Information, 9(11), 280. https://doi.org/10.3390/info9110280.

[5] Hernandez-Suarez, A., Sanchez-Perez, M., Toscano-Medina, K., Martinez-Hernandez, V., & Nakano-Miyatake, M. (2018). Using Twitter as a social sensor for cyber-attack prediction. Sensors, 18(5), 1380. https://doi.org/10.3390/s18051380.

[6] Alserhani, F., & Aljared, A. (2023). Evaluating Ensemble Learning Mechanisms for Predicting Advanced Cyber Attacks. Applied Sciences, 13(24), 13310. https://doi.org/10.3390/app132413310.

[7] Colnerič, N., & Demšar, J. (2021). Emotion recognition in text: Recent advances and future directions. Information Fusion, 64, 127–145. https://doi.org/10.1016/j.inffus.2020.07.006.

[8] Adewopo, A., AlEroud, A., Alsmadi, I., & Al-Qurishi, M. (2022). Deep learning approach for threat detection in deep web forums. arXiv preprint arXiv:2202.01448. https://doi.org/10.48550/arXiv.2202.01448.

[9] Gautam, A.S., Gahlot, Y., Kamat, P. (2020). Hacker Forum Exploit and Classification for Proactive Cyber Threat Intelligence. In: Smys, S., Bestak, R., Rocha, Á. (eds) Inventive Computation Technologies. ICICIT 2019. Lecture Notes in Networks and Systems, vol 98. Springer, Cham. https://doi.org/10.1007/978-3-030-33846-6_32.

[10] Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., & Smith, N. A. (2020). Don't stop pretraining: Adapt language models to domains and tasks. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020) (pp. 8342–8360). Association for Computational Linguistics. https://doi.org/10.48550/arXiv.2004.10964.

[11] R. Williams, S. Samtani, M. Patton, H. Chen, Incremental Hacker Forum Exploit Collection and Classification for Proactive Cyber Threat Intelligence: An Exploratory Study, 2018 IEEE International Conference on Intelligence and Security Informatics (ISI), Miami, FL, USA, pp. 94–99, doi: 10.1109/ISI.2018.8587336.

[12] Mambetov, S., Ormanbekova, A., Joldasbayev, S., Duissembayeva, L., & Mailykhanova, B. (2025). Graph-based neural networks with neural ODEs for robust speech processing in environmental and human-centric systems. In E3S Web of Conferences (Vol. 627, p. 04017). EDP Sciences. https://doi.org/10.1051/e3sconf/202562704017.

[13] Kaur, P., Singh, A., Kumar, A. (2024). Advanced Hacker Forum Use Collection and Classification Methods for Preventive Cyber Threat Intelligence, 2024 4th International Conference on Technological Advancements in Computational Sciences (ICTACS), Tashkent, Uzbekistan, pp. 850–855, doi: 10.1109/ICTACS62700.2024.10841221.

[14] B. Ampel, S. Samtani, H. Zhu, S. Ullman, H. Chen, Labeling Hacker Exploits for Proactive Cyber Threat Intelligence: A Deep Transfer Learning Approach, 2020 IEEE International Conference on Intelligence and Security Informatics (ISI), Arlington, VA, USA, pp. 1–6, doi: 10.1109/ISI49825.2020.9280548

[15] Mambetov, S., Begimbayeva, Y., Gurko, O., Doroshenko, H., Joldasbayev, S., Fridman, O., ... & Neronov, S. (2024). Detection And Classification Of Threats And Vulnerabilities On Hacker Forums Based On Machine Learning. Eastern-European Journal of Enterprise Technologies, (9). https://doi.org/10.15587/1729-4061.2024.306522.