



IRSTI 50.41.29

Article

<https://doi.org/10.32523/2616-7263-2026-154-1-249-262>

## Enhancing user experience through advanced sentiment analysis

A.K. Aitim  , M.A. Abdulla  \*

*International Information Technology University, Almaty, Kazakhstan*

*E mail: [a.aitim@iitu.edu.kz](mailto:a.aitim@iitu.edu.kz), [muslima.abaykyzy@gmail.com](mailto:muslima.abaykyzy@gmail.com)*

**Abstract.** The rapid growth of user-generated textual content on social media platforms, messaging applications, e-commerce websites, and online services has increased the importance of automated sentiment analysis for understanding public opinion and improving digital services. The aim of this research is to investigate modern Natural Language Processing (NLP) approaches for sentiment analysis and to develop an effective method for analyzing and summarizing consumer reviews in order to enhance user experience. The scientific significance of the study lies in exploring advanced machine learning and deep learning techniques for processing large volumes of unstructured textual data and identifying emotional patterns in user feedback. The research methodology includes text preprocessing, tokenization, vectorization using transformer-based representations, and sentiment classification using Long Short-Term Memory (LSTM) models and machine learning algorithms. The results show that deep learning models improve the accuracy of sentiment detection and enable more reliable analysis of user opinions expressed in textual data. The findings contribute to the development of efficient analytical tools for sentiment analysis and provide practical support for businesses in understanding consumer preferences and improving the quality of digital services.

**Keywords:** sentiment analysis, natural language processing, user experience, LSTM networks, text classification.

## Introduction

Sentiment analysis systems apply natural language processing and machine learning algorithms to text data. Preprocessing steps like tokenization and part-of-speech tagging prepare the data. For machine learning techniques, systems are first trained on annotated sentiment data. Then, they apply that knowledge to determine the sentiment in the new text. Systems might examine word choices, grammar constructs, and linguistic patterns to gauge sentiment. Various methods and models exist to leverage in sentiment analysis, each has its pros and cons, and most suitable cases to be efficient [1].

BERT (Bidirectional Encoder Representations from Transformers): Used for tasks like text classification, named entity recognition, and question answering [2]. Additionally, RoBERTa, DistilBERT, and XLNet were examined for performance in recognizing emotions in text. The models successfully recognized emotions expressed in written text, such as joy, humiliation, guilt, fear, anger, disgust, and sadness. RoBERTa achieved the top accuracy score of 0.74. The model needs to be made more general [3].

A hybrid method based on attributes to analyze consumers' intelligence by detecting characteristics through POS tags. The method needs to identify a greater number of subjects that cover all attributes while reducing computational expenses. Additionally, the research should encompass short text categorization and hybrid approaches, while also removing human labeling of attribute specific words in current lexicons [4].

A classification system based on specific factors with the use of Long-Short-Term Memory (LSTM) models [5]. LSTM by incorporating a combination of LSTM and a recurrent add-on network.

A combination of ARM techniques to extract characteristics is an option as well, with the possible five different strategies to utilize them. Due to the context disregarding implicit words, only explicit elements were retrieved [6]. The F-measure reached 75.51 percent. Challenges in this approach: Certain elements of the hybrid association rule mining prove challenging to manage in real-world scenarios.

## The methodology

Sentiment analysis (SA) techniques, such as the SentiDiff algorithm, integrate sentiment diffusion patterns with textual information to improve sentiment analysis outcomes on Twitter datasets [7]. A process called sentiment reversal and identified multiple intriguing features linked to sentiment reversals [8]. Then examining the links between text content in Twitter posts and how emotions spread to forecast sentiment orientations from Twitter posts. Extensive tests on an actual dataset demonstrate that the suggested method enhances the area under the Precision-Recall curve for Twitter sentiment analysis compared to current SA techniques that are based on textual information [9].

Zhu et al. propose SentiVec as a method for optimizing sentiment word embedding kernels [10]. The initial stage of the study includes supervised learning, while the subsequent stage involves unsupervised updating models such as O2SR and C2OR [11]. Experimental results show that sentiment vectors with the best performance can accurately capture both semantics and sentiment in comparison to baseline methods, exceeding their performance on tasks like word analogy, similarity, and sentiment analysis [12].

Aspect-based sentiment analysis (ABSA) techniques includes a framework for text processing that could summarize reviews [13]. Within this framework, the goal is to identify the broad

categories mentioned in review sentences. There are two methods to achieve this. The initial method suggested is an autonomous process that identifies aspect categories by utilizing association rule mining on co-occurrence frequency data derived from a corpus. The proposed unsupervised technique exceeds many simple baselines with a 67 percent F1-measure. Alternative ABSA hybrid method for examining smart app review entities by combining domain lexicons and rules. This approach involves the use of language processing methods, rules, and lexicons to tackle various difficulties in sentiment analysis, leading to the creation of condensed outcomes [14]. The accuracy of aspect extraction sees a significant improvement when implicit factors are considered [15].

A method called Cross-Lingual Sentiment Classification to create a detailed aspect-level system by aligning topics across groups [16]. The AOS model combines aspects, opinions, and sentiments in reviews across different domains using a coarse alignment method to improve latent feature representation [17]. To enhance AOS further, a partly supervised AOS model that utilizes labeled data from the source language to minimize the discrepancy in feature representations between two language domains through logistic regression. Findings from extensive experiments on various multilingual product review datasets show that this method surpasses multiple cutting-edge methods by a wide margin [18]. There was a major challenge in sentiment analysis by addressing the presence of bipolar words in SA to enhance aspect-based sentiment analysis. The root is analyzing how words shift in meaning based on context, how this affects product ratings, and specific aspects, yielding notable findings [19].

Another automated approach to analyzing sentiments of evolving elements in customer reviews collected from various sources via web scraping to address the cold start problem. The accuracy of the system was enhanced by the addition of new stop words. Another method relies on three broad stages divided into multiple detailed operations. In the initial phase, the preliminary polarity lexicon and aspect word sets, serving as representations of aspects, are chosen to extract existing domain knowledge from the dataset [20]. These two resources are used as basic input for an expectation-maximization algorithm that calculates the likelihood of a word considering its aspect and emotion.

Deep learning-based SA methods have been utilized in the field of sentiment analysis to enhance effectiveness in recent times [21]. The initial end-to-end semi-supervised Multi-task Learning framework (SEML) for conducting ABSA on user reviews. Aspect mining and aspect sentiment classification in ABSA are both taught simultaneously in a combined session. SEML generate unified representations of reviews from a combination of three stacked and bidirectional neural layers. MAGRU enhances GRU by incorporating the moving-window attention mechanism to capture extensive contextual information. The suggested model incorporates Cross-View Training (CVT) to train additional prediction modules on unlabeled reviews, improving representation learning.

The proposed novel model, named Attention Emotion Enhanced (AEC)-LSTM enhances the LSTM network performance by integrating an attention mechanism and emotional intelligence. Initially, a new type of LSTM network is introduced, called ELSTM, that incorporates emotions to improve learning by utilizing an emotion modulator and estimator to boost the feature learning capacity of LSTM networks. With combined ELSTM with additional processes such as convolution, pooling, and concatenation it offers an improved depiction of diverse structural patterns in textual sequences. Next, the weight of hidden representations of text is adaptively adjusted through the use of the topic-level attention mechanism.

Pre-processing, identification of aspects, analysis of polarity, and categorization of sentiment are all stages of the model being suggested. Initially, the demonetization tweets from the Kaggle data set was collected and prepared. Aspect extraction is utilized to extract sentiment words from processed data. Through the use of polarity measure calculation and Word2vec, these identified aspect words are converted into attributes. The polarity measures were improved by combining the FireFly Algorithm (FF) and Multi-Verse Optimization (MVO) to create a new algorithm named FireFly-oriented Multi-Verse Optimizer (FF-MVO). Next, the combined features are fed into a Recurrent Neural Network (RNN) to classify sentiments as either positive or negative.

A common use for a sentiment analysis model is to determine if a text conveys positivity, negativity, or neutrality. However, this process typically requires access to plain text, which can potentially raise privacy concerns. Fully Homomorphic Encryption (FHE) permits performing calculations on encrypted data without requiring decryption beforehand. This makes it ideal for situations where a user's personal and possibly confidential information is in danger (such as analyzing emotions in private messages). Initiating the process by bringing in the necessary materials for transformers. To utilize the well-known library by Hugging Face to swiftly obtain a transformer. The BERT transformer model was selected, as shown in Fig. 1.

```
import torch
from transformers import
AutoModelForSequenceClassification, AutoTokenizer
device = "cuda:0" if torch.cuda.is_available() else
"cpu"
# Load the tokenizer (converts text to tokens)
tokenizer =
AutoTokenizer.from_pretrained("cardiffnlp/twitter-rober
ta-base-sentiment-latest")
# Load the pre-trained model
transformer_model =
AutoModelForSequenceClassification.from_pretrained(
    "cardiffnlp/twitter-roberta-base-sentiment-latest"
)
```

Figure 1. Downloading the model that ready to be used [22]

Tokenizing involves dividing the text into tokens and assigning a numerical value to each one as demonstrated in Figure 2. After that, we input the tokenized text into the transformer model, which generates a hidden representation, which result of the self-attention layers that are frequently utilized as input in the classification layers for each individual word. Ultimately, we calculate the mean of the word representations to obtain a representation at the text level. The outcome is a matrix with dimensions –number of samples, hidden size. The hidden size refers to the number of dimensions in the hidden representation. BERT has a hidden size of 768. The concealed representation is a numerical vector that depicts the text and can be applied to various tasks as shown in Fig. 3. In this instance, we'll employ it for classification using XGBoost later.

```
def text_to_tensor(
    list_text_X_train: list,
    transformer_model:
    AutoModelForSequenceClassification,
    tokenizer: AutoTokenizer,
    device: str,
) -> np.ndarray:
    # Tokenize each text in the list one by one
    tokenized_text_X_train_split = []
    tokenized_text_X_train_split = [
        tokenizer.encode(text_x_train,
            return_tensors="pt")
        for text_x_train in list_text_X_train
    ]
```

Figure 2. Downloading the model that ready to be used [26]

```
# Let's vectorize the text using the transformer
list_text_X_train = text_X_train.tolist()
list_text_X_test = text_X_test.tolist()
```

Figure 3. Vectorizing the text using transformer [22]

After preparing our training and test sets for classifier training, the next step is to train our Fully Homomorphic Encryption (FHE) model. It will be simple to do with a hyper-parameter tuning tool like GridSearch from scikit-learn. The output is as follows in Fig.4:

```
Best score: 0.8378111718275654
Best parameters: {'max_depth': 1, 'n_bits': 3,
'n_estimators': 50, 'n_jobs': -1}
```

Figure 4. Received output [22]

Now, we will make predictions on encrypted text. The concept is to encrypt the transformer's output instead of the original text. By enabling `execute_in_fhe=True` in the `predict` function in Concrete-ML, you can accomplish this task swiftly which is demonstrated in Fig. 5. This feature is specifically for developers to test the execution time of the FHE model. Further along, we will explore how to successfully implement this in a deployment environment and get the results as shown in Fig. 6.

```
start = time.perf_counter()
decrypted_proba =
best_model.predict_proba(X_tested_tweet,
execute_in_fhe=True)
end = time.perf_counter()
fhe_exec_time = end - start
print(f"FHE inference time: {fhe_exec_time:.4f}
seconds")
```

Figure 5. Predicting with FHE over a single tweet, including the estimated time [22]

```
Probabilities from the FHE inference: [0.08434131
0.05571389 0.8599448]
Probabilities from the clear model: [0.08434131
0.05571389 0.8599448 ]
```

Figure 6. Read output [22]

The total settlement was (1.64) mm at a load of (100) tons (working load) and the total settlement was (6.46) mm at a load of (200) tons (equal to 200% of working load) in the second

In today's world, we are encompassed by a variety of gadgets for recording, such as smartphones, computers, tablets, smartwatches, and video cameras, among others. Yet, traditional methods for handling signals from multiple devices cannot be used together due to the requirement of simultaneous recording. Currently, microphone array processing available for purchase is constrained to using a single device with all microphones mounted on it.

Depending on your preferred method of analysis, sound can be broken down into several parts. We will divide sound into two primary categories for the purposes of this article: amplitude and frequency. Pitch, Formant, Bandwidth, Sampling Rate, and other components (overtones, harmonics, etc.) can be further separated out of the frequency components. Amplitude, as previously mentioned, controls a sound's volume. Sound wave strength is measured by loudness. The amount of energy a sound has over a space is its intensity. If you hear the same sound in a smaller space, it sounds louder. Higher intensity noises are generally referred to as louder. Therefore, amplitude is a measurement of energy. The amplitude of a wave increases with its energy. Intensity rises in tandem with amplitude. The number of samples a sound contains per second is its sample rate, also known as sampling frequency. For instance, a recording lasting five seconds will have 20,000 samples if the sampling rate is 4000 hertz. Pitch refers to the frequency at which the waveform repeats itself, which is the fundamental component of sound. As an example, there is a tool DeepSpeech, which provides qualitative performance. It starts with initiation, as in Fig. 7, it is important to consider that only 16 kilohertz (kHz) .wav files are used.

```
# Install DeepSpeech
pip3 install deepspeech

# Download pre-trained English model files
curl -LO
https://github.com/mozilla/DeepSpeech/releases/download/v0.9.3/d
eepspeech-0.9.3-models.pbmm
curl -LO
https://github.com/mozilla/DeepSpeech/releases/download/v0.9.3/d
eepspeech-0.9.3-models.scorer
```

Figure 7. Installation [22]

The `__init__` function of the `VADAudio` class initializes with parameters for a callback, device, input rate, and file, with default values for everything but the input rate. The class sets up an instance method `proxy_callback` that returns the `PyAudio` signal after calling the callback.

During initialization, it assigns the callback to handle audio data, creates a buffer queue, sets the device and input rate, and calculates the block size based on the sample rate. A `PyAudio` stream

is created with parameters like format, channels, and rate, and the aggressiveness of background noise filtering is set. If a device is provided, it is used as the input device, or if a file is passed, the chunk size is set, and the file is opened for reading.

The class contains six functions: `resample`, `read_resampled`, `read`, `write_wav`, a frame generator, and a voice activity detection (VAD) collector. The `resample` function adjusts audio data to a 16 kHz sample rate. The `read` and `read_resampled` functions handle reading raw and resampled audio data, respectively. The `write_wav` function writes audio data to a WAV file, which is shown in Fig. 8. The frame generator yields raw or resampled data based on the input rate.

```
def write_wav(self, filename, data):
    logging.info("write wav %s", filename)
    wf = wave.open(filename, 'wb')
    wf.set_channels(self.CHANNELS)
    # wf.setsampwidth(self.pa.get_sample_size(FORMAT))
    assert self.FORMAT == pyaudio.paInt16
    wf.setsampwidth(2)
    wf.setframerate(self.sample_rate)
    wf.writeframes(data)
    wf.close()
```

Figure 8. Function that writes data to an output file [22]

The utilization of the spatial diversity provided by multiple audio devices without the need for wired networking is a significant challenge, leading to extensive research efforts in the past decade due to its potential practical and commercial advantages.

WASNs have emerged as a fresh approach to acoustic sensing, aiming to address the constraints of stand-alone devices. By addressing new issues in managing asynchronous channels, unknown microphone positions, and distributed computing, in addition to enabling wireless communication between microphone nodes, the WASN allows us to distribute multiple recording devices spatially. These could encompass a broader range and make use of the nodes to create a larger microphone array. It guarantees a major enhancement in the efficiency of different audio assignments, like improving speech quality, recognizing speech, identifying speakers, analyzing scenes, and detecting unusual sounds.

Although it might be challenging to assign a strict classification to spatial audio techniques, many of them fall into one of two general categories: model-based or data-based. In the past, model-based techniques created sound scopes using discrete virtual sound sources that are powered by a set of audio input signals and analytically defined by mathematical or physical models. Model-based techniques include vector base amplitude panning (VBAP), stereophonic amplitude panning (WFS), and others. Conversely, data-driven spatial audio techniques use sound scene representations, where the audio waves contain spatial information stored in them.

The source of the spatial data may come from simulations, acoustic measurements, or array recordings. Social media is a vital tool for community discussion and information sharing about a wide range of events. Creating datasets from three specific social media platforms-Instagram, Snapchat, and Twitter is used to investigate the quality of the sentiment analysis (SA) of impressions about. WhatsApp is the most appropriate platform for expressing negative emotions, followed by Facebook, Twitter, and Instagram, according to the analysis of the samples and the methods. Perceived fit for WhatsApp was highest for conveying pleasant emotions, followed by Instagram, Facebook, and Twitter. All that the system offers is a comparison between these four

platforms. The RF methodology was the main method used, and it had a 94% accuracy rate in predicting results. The fact that the data collection period was so brief, just one month, came with certain restrictions. Taking twitter data set as an example, a visualization of sentiment analysis. The first step is exploring the dataset, as shown in Fig.9.

```
import pandas as pd

data = pd.read_csv('training.csv', encoding='ISO-8859-1', header=None)
column_names = ['target', 'ids', 'date', 'flag', 'user', 'text']
data.columns = column_names
head = data.head()
info = data.info()
describe = data.describe()
head, info, describe
```

Figure 9. Exploring dataset [22]

Insights received as:

- The dataset has 1.6 million tweets, with no missing values in any column.
- Each tweet has a target sentiment (0 for negative, 2 neutral, 4 for positive), an ID, a timestamp, a flag (query or 'NO\_QUERY'), the username, and the text.
- The sentiment targets are balanced, with an equal number of positive and negative labels [8]. Further step is visualising received insights and dividing positive and negative ones, which is demonstrated in Fig. 10. Visualized picture of the output is provided below in Fig. 11.

```
column
positive_tweets = data[data['target'] == 4]['text']
negative_tweets = data[data['target'] == 0]['text']

sample_positive_text = " ".join(text for text in
positive_tweets.sample(frac=0.1, random_state=23))
sample_negative_text = " ".join(text for text in
negative_tweets.sample(frac=0.1, random_state=23))

sentiments
wordcloud_positive = WordCloud(width=800, height=400,
max_words=200,
background_color="white").generate(sample_positive_text)
wordcloud_negative = WordCloud(width=800, height=400,
max_words=200,
background_color="white").generate(sample_negative_text)

plt.figure(figsize=(15, 7.5))
# Positive word cloud
plt.subplot(1, 2, 1)
plt.imshow(wordcloud_positive, interpolation='bilinear')
plt.title('Positive Tweets Word Cloud')
plt.axis("off")
# Negative word cloud
plt.subplot(1, 2, 2)
plt.imshow(wordcloud_negative, interpolation='bilinear')
plt.title('Negative Tweets Word Cloud')
plt.axis("off")

plt.show()
```

Figure 10. Selecting positive and negative tweets and combine them together [22]

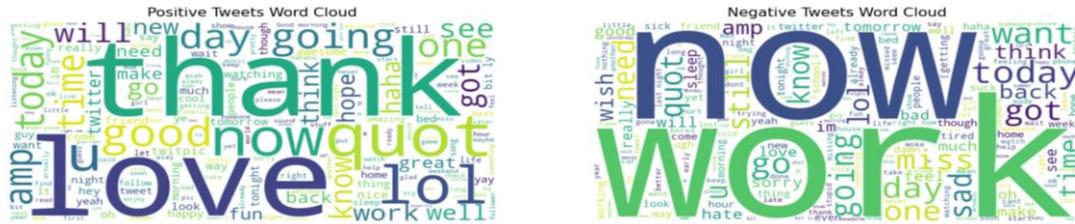


Figure 11. Sentiment analysis in social media

### Findings/Discussion

The study explored several sentiment analysis techniques and models, each offering different levels of performance and efficiency across various text data sets. RoBERTa demonstrated the highest accuracy in recognizing emotions in text, achieving a score of 0.74. However, there remains a need to generalize this model further to ensure consistent performance across diverse datasets. Meanwhile, hybrid methods, including a combination of part-of-speech tagging and attribute detection, offered insights into consumer intelligence. These methods showed potential for improving sentiment analysis by expanding the range of detectable attributes and reducing computational costs, though challenges persist in short text categorization. Deep learning models, such as LSTM combined with additional network layers, improved sentiment classification, particularly in capturing long-term dependencies in text. The hybrid approach of integrating association rule mining (ARM) techniques and recurrent neural networks (RNNs) also showed promise in extracting explicit sentiment elements from text data, reaching an F-measure of 75.51%. However, managing implicit elements remains a challenge in real-world applications, as these methods tend to overlook nuanced contextual words. The ABSA techniques explored in this research provided a framework for summarizing reviews by categorizing text into broad aspects. The proposed unsupervised method, which utilized association rule mining for aspect category identification, outperformed simple baselines, achieving a 67% F1 score. However, incorporating implicit factors into aspect extraction led to significant improvements, particularly in enhancing accuracy. The cross-lingual sentiment classification approach successfully aligned topics across different languages and domains, further improving latent feature representation. This model surpassed cutting-edge methods in multilingual datasets, though bipolar word presence continues to be a major challenge. One of the key challenges identified in sentiment analysis is the need for robust methods to handle privacy concerns. Fully Homomorphic Encryption (FHE) offers a promising solution by allowing sentiment analysis on encrypted data, addressing the risks of analyzing personal information. Additionally, the study highlighted the difficulties in analyzing evolving elements in customer reviews, particularly in the cold start problem, where limited data availability hampers model performance. In the realm of spatial audio processing, wireless acoustic sensor networks (WASNs) emerged as a novel approach to improve speech quality, speaker recognition, and sound detection. By distributing recording devices spatially, WASNs overcome the limitations of stand-alone devices, enhancing the efficiency of various audio-related tasks. Data-driven spatial audio techniques were noted for their ability to capture and utilize spatial information from sound scenes. Finally, sentiment analysis on social media platforms revealed varying performance across different channels. WhatsApp emerged as the most effective platform for expressing negative emotions, while Instagram excelled in conveying pleasant emotions. The random forest methodology used in the analysis achieved a high accuracy rate of 94%, though the brief data collection period posed limitations.

## Conclusion

Overall, the results indicate that while several sentiment analysis techniques and models demonstrate high performance, challenges remain in generalization, managing implicit contextual elements, and addressing privacy concerns. Future research should focus on refining hybrid methods, improving cross-lingual sentiment analysis, and leveraging advanced encryption techniques to ensure the safe and effective application of sentiment analysis across various domain.

## Acknowledgment

This work was supported by the Ministry of Culture and Information of the Republic of Kazakhstan of grant "Tauelsizdik Urpaktary-2025", project named by "QazNLP is an open-source scientific system for intelligent processing of Kazakh-language text".

## The contribution of the authors:

**A.K.Aitim** - general guidance, supervising, idea, problem setting, concept, methodology, critical review of content, final decision, final editing.

**M.A.Abdulla** - solving theoretical issues, data collection, article design, analysis, visualization, interpretation, conclusion, editing.

## References

1. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
2. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of NAACL-HLT*, 4171–4186. <https://doi.org/10.18653/v1/N19-1423>
3. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30. <https://doi.org/10.48550/arXiv.1706.03762>
4. Tang, D., Qin, B., & Liu, T. (2015). Document modeling with gated recurrent neural network for sentiment classification. *Proceedings of EMNLP*, 1422–1432. <https://doi.org/10.18653/v1/D15-1167>
5. Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1), 1–167. <https://doi.org/10.2200/S00416ED1V01Y201204HLT016>
6. Aitim, A. (2024). Developing methods for automatic processing systems of Kazakh language. *KazATC Bulletin*, 133(4), 254–265. <https://doi.org/10.52167/1609-1817-2024-133-4-254-265>
7. Aitim, A., & Satybaldiyeva, R. (2024). A systematic review of existing tools to automated processing systems for Kazakh language. *Bulletin of Abai KazNPU. Series of Physical and Mathematical Sciences*, 87(3), 106–122. <https://doi.org/10.51889/2959-5894.2024.87.3.009>
8. Hoang, M., Bihorac, O. A., & Rouces, J. (2019). Aspect-based sentiment analysis using BERT. *Proceedings of the Nordic Conference on Computational Linguistics*, 187–196. <https://doi.org/10.48550/arXiv.1904.05064>

9. Bello, A., et al. (2023). A BERT framework to sentiment analysis of tweets. *Sensors*, 23, 506. <https://doi.org/10.3390/s23010506>
10. Liu, N., Li, Y., & Liu, Y. (2022). A BERT-based aspect-level sentiment analysis algorithm for cross-domain text. *Applied Sciences*, 12, 6637. <https://doi.org/10.3390/app12136637>
11. Atlas, L. G., et al. (2025). A novel sentence-level sentiment analysis approach for mining online product reviews. *Scientific Reports*, 15, 1104. <https://doi.org/10.1038/s41598-025-01104-0>
12. Zhang, H., & Li, J. (2025). Hybrid CNN–LSTM model for social media sentiment analysis. *Scientific Reports*, 15, 1104. <https://doi.org/10.1038/s41598-025-01104-0>
13. Mandhasiya, D. G., Murfi, H., & Bustamam, A. (2024). The hybrid of BERT and deep learning models for Indonesian sentiment analysis. *Indonesian Journal of Electrical Engineering and Computer Science*, 33(1), 591–602. <https://doi.org/10.11591/ijeecs.v33.i1.pp591-602>
14. Aitim, A. (2025). Building a high-quality annotated corpus for Kazakh NLP: A pipeline approach. *Vestnik KazUTB*, 4(29). <https://doi.org/10.58805/kazutb.v.4.29-1092>
15. Zhang, X., Liu, Y., & Chen, H. (2025). A BERT–BiLSTM–attention framework for robust sentiment classification. *Systems*, 13, 964. <https://doi.org/10.3390/systems13110964>
16. Acheampong, F. A., Nunoo-Mensah, H., & Chen, W. (2020). Comparative analyses of BERT, RoBERTa, DistilBERT, and XLNet for text-based emotion recognition. *IEEE Access*, 8, 83632–83643. <https://doi.org/10.1109/ACCESS.2020.2992247>
17. Mao, Y., Liu, Q., Zhang, H., & Deng, Z. (2024). Sentiment analysis methods, applications, and challenges: A systematic literature review. *Journal of King Saud University – Computer and Information Sciences*, 36, 102048. <https://doi.org/10.1016/j.jksuci.2024.102048>
18. Zhang, L., Wang, S., & Liu, B. (2018). Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4), e1253. <https://doi.org/10.1002/widm.1253>
19. Sun, C., Huang, L., & Qiu, X. (2019). Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. *Proceedings of NAACL-HLT*, 380–385. <https://doi.org/10.18653/v1/N19-1035>
20. Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C., Ng, A., & Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. *Proceedings of EMNLP*, 1631–1642. <https://doi.org/10.18653/v1/D13-1170>
21. Aitim, A., & Abdulla, M. (2024). Data processing and analysing techniques in UX research. *Procedia Computer Science*, 251, 591–596. <https://doi.org/10.1016/j.procs.2024.11.154>
22. QNLP – Full Kazakh NLP Suite GitHub repository. (2025). Retrieved from <https://github.com/Aigerimhub/qnlp>

**Ә.Қ. Әйтiм, М.А. Абдулла\***

Халықаралық ақпараттық технологиялар университетi, Алматы, Қазақстан

**Жетiлдiрiлген көңiл-күйдi талдау арқылы пайдаланушы тәжiрибесiн жақсарту**

**Аңдатпа.** Әлеуметтік медиа платформаларында, хабар алмасу қосымшаларында, электрондық коммерция сайттарында және басқа да онлайн сервистерде пайдаланушылар жасайтын мәтіндік контенттің жылдам өсуі қоғамдық пікірді түсіну және цифрлық

қызметтердің сапасын жақсарту үшін автоматтандырылған көңіл-күйді талдаудың маңыздылығын арттырды. Бұл зерттеудің мақсаты – көңіл-күйді талдау үшін заманауи табиғи тілді өңдеу (Natural Language Processing, NLP) тәсілдерін зерттеу және пайдаланушылардың пікірлерін талдау мен қорытындылау үшін пайдаланушы тәжірибесін жетілдіруге арналған тиімді әдіс ұсыну. Зерттеудің ғылыми маңыздылығы үлкен көлемдегі құрылымдалмаған мәтіндік деректерді өңдеу және пайдаланушылардың пікірлеріндегі эмоциялық заңдылықтарды анықтау үшін машиналық оқыту мен терең оқыту әдістерін зерттеуде көрінеді. Зерттеу әдістемесі мәтінді алдын ала өңдеуді, токенизацияны, трансформерлік модельдер негізінде векторизацияны, сондай-ақ Long Short-Term Memory (LSTM) модельдері мен машиналық оқыту алгоритмдерін қолдану арқылы көңіл-күйді жіктеуді қамтиды. Зерттеу нәтижелері терең оқыту модельдері көңіл-күйді анықтау дәлдігін арттырып, мәтіндік деректердегі пайдаланушылар пікірін неғұрлым сенімді талдауға мүмкіндік беретінін көрсетеді. Алынған нәтижелер көңіл-күйді талдауға арналған тиімді аналитикалық құралдарды дамытуға үлес қосады және тұтынушылардың қалауын түсіну мен цифрлық қызметтердің сапасын арттыруда бизнеске практикалық қолдау көрсетеді.

**Түйін сөздер:** көңіл-күйді талдау, табиғи тілді өңдеу, пайдаланушы тәжірибесі, LSTM желілері, мәтінді жіктеу.

**Ә.Қ. Әйтiм, М.А. Абдулла\***

*Международный университет информационных технологий, Алматы, Казахстан*

### **Повышение пользовательского опыта за счет расширенного анализа настроений**

**Аннотация.** Стремительный рост пользовательского текстового контента на платформах социальных сетей, в приложениях для обмена сообщениями, на сайтах электронной коммерции и других онлайн-сервисах значительно повысил значимость автоматического анализа тональности для понимания общественного мнения и улучшения цифровых сервисов. Цель данного исследования заключается в изучении современных подходов к обработке естественного языка (Natural Language Processing, NLP) для анализа тональности и разработке эффективного метода анализа и обобщения пользовательских отзывов с целью улучшения пользовательского опыта. Научная значимость работы заключается в исследовании современных методов машинного обучения и глубокого обучения для обработки больших объемов неструктурированных текстовых данных и выявления эмоциональных закономерностей в пользовательских отзывах. Методология исследования включает предварительную обработку текста, токенизацию, векторизацию с использованием трансформерных представлений, а также классификацию тональности с применением моделей Long Short-Term Memory (LSTM) и алгоритмов машинного обучения. Результаты исследования показывают, что модели глубокого обучения повышают точность определения тональности и обеспечивают более надежный анализ мнений пользователей, выраженных в текстовых данных. Полученные результаты способствуют развитию эффективных аналитических инструментов для анализа тональности и имеют практическое значение для бизнеса при изучении предпочтений потребителей и повышении качества цифровых сервисов.

**Ключевые слова:** анализ тональности, обработка естественного языка, пользовательский опыт, сети LSTM, классификация текста.

### References

1. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
2. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of NAACL-HLT*, 4171–4186. <https://doi.org/10.18653/v1/N19-1423>
3. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30. <https://doi.org/10.48550/arXiv.1706.03762>
4. Tang, D., Qin, B., & Liu, T. (2015). Document modeling with gated recurrent neural network for sentiment classification. *Proceedings of EMNLP*, 1422–1432. <https://doi.org/10.18653/v1/D15-1167>
5. Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1), 1–167. <https://doi.org/10.2200/S00416ED1V01Y201204HLT016>
6. Aitim, A. (2024). Developing methods for automatic processing systems of Kazakh language. *KazATC Bulletin*, 133(4), 254–265. <https://doi.org/10.52167/1609-1817-2024-133-4-254-265>
7. Aitim, A., & Satybaldiyeva, R. (2024). A systematic review of existing tools to automated processing systems for Kazakh language. *Bulletin of Abai KazNPU. Series of Physical and Mathematical Sciences*, 87(3), 106–122. <https://doi.org/10.51889/2959-5894.2024.87.3.009>
8. Hoang, M., Bihorac, O. A., & Rouces, J. (2019). Aspect-based sentiment analysis using BERT. *Proceedings of the Nordic Conference on Computational Linguistics*, 187–196. <https://doi.org/10.48550/arXiv.1904.05064>
9. Bello, A., et al. (2023). A BERT framework to sentiment analysis of tweets. *Sensors*, 23, 506. <https://doi.org/10.3390/s23010506>
10. Liu, N., Li, Y., & Liu, Y. (2022). A BERT-based aspect-level sentiment analysis algorithm for cross-domain text. *Applied Sciences*, 12, 6637. <https://doi.org/10.3390/app12136637>
11. Atlas, L. G., et al. (2025). A novel sentence-level sentiment analysis approach for mining online product reviews. *Scientific Reports*, 15, 1104. <https://doi.org/10.1038/s41598-025-01104-0>
12. Zhang, H., & Li, J. (2025). Hybrid CNN–LSTM model for social media sentiment analysis. *Scientific Reports*, 15, 1104. <https://doi.org/10.1038/s41598-025-01104-0>
13. Mandhasiya, D. G., Murfi, H., & Bustamam, A. (2024). The hybrid of BERT and deep learning models for Indonesian sentiment analysis. *Indonesian Journal of Electrical Engineering and Computer Science*, 33(1), 591–602. <https://doi.org/10.11591/ijeecs.v33.i1.pp591-602>
14. Aitim, A. (2025). Building a high-quality annotated corpus for Kazakh NLP: A pipeline approach. *Vestnik KazUTB*, 4(29). <https://doi.org/10.58805/kazutb.v.4.29-1092>
15. Zhang, X., Liu, Y., & Chen, H. (2025). A BERT–BiLSTM–attention framework for robust sentiment classification. *Systems*, 13, 964. <https://doi.org/10.3390/systems13110964>

16. Acheampong, F. A., Nunoo-Mensah, H., & Chen, W. (2020). Comparative analyses of BERT, RoBERTa, DistilBERT, and XLNet for text-based emotion recognition. *IEEE Access*, 8, 83632–83643. <https://doi.org/10.1109/ACCESS.2020.2992247>
17. Mao, Y., Liu, Q., Zhang, H., & Deng, Z. (2024). Sentiment analysis methods, applications, and challenges: A systematic literature review. *Journal of King Saud University – Computer and Information Sciences*, 36, 102048. <https://doi.org/10.1016/j.jksuci.2024.102048>
18. Zhang, L., Wang, S., & Liu, B. (2018). Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4), e1253. <https://doi.org/10.1002/widm.1253>
19. Sun, C., Huang, L., & Qiu, X. (2019). Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. *Proceedings of NAACL-HLT*, 380–385. <https://doi.org/10.18653/v1/N19-1035>
20. Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C., Ng, A., & Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. *Proceedings of EMNLP*, 1631–1642. <https://doi.org/10.18653/v1/D13-1170>
21. Aitim, A., & Abdulla, M. (2024). Data processing and analysing techniques in UX research. *Procedia Computer Science*, 251, 591–596. <https://doi.org/10.1016/j.procs.2024.11.154>
22. QNLP – Full Kazakh NLP Suite GitHub repository. (2025). Retrieved from <https://github.com/Aigerimhub/qnlp>

### ***Information about the authors:***

Aitim A.K. – PhD, Associate Professor, Department of Information Systems, International Information Technology University, Almaty, Kazakhstan

Abdulla M.A. – corresponding author, Bachelor of Engineering and Technology, International Information Technology University, Almaty, Kazakhstan

Әйтiм Ә.Қ. – PhD, ассоциированный профессор кафедры Информационных систем, Международный университет информационных технологий, Алматы, Казахстан.

Абдулла М.А. – автор для корреспонденции, бакалавр техники и технологий, Международный университет информационных технологий, Алматы, Казахстан.

Әйтiм Ә.Қ. – PhD, Ақпараттық жүйелер кафедрасының қауымдастырылған профессоры, Халықаралық ақпараттық технологиялар университеті, Алматы, Қазақстан

Абдулла М.А. – хат-хабар авторы, техника және технология ғылымдарының бакалавры, Халықаралық ақпараттық технологиялар университеті, Алматы, Қазақстан



**Copyright:** © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY NC) license (<https://creativecommons.org/licenses/by-nc/4.0/>).