Journal of Computer Science and Artificial Intelligence
ISSN: 3078-8242 (Print), ISSN: 3078-2309 (Online) | Vol. 2, No. 1, 2025

Research on Multi-class Sentiment Analysis of Social Media Texts Based on the ERNIE Model

Xiaomei Gu, Busheng Li *

School of Information Engineering, Jingdezhen Ceramic University, Jingdezhen 333000, China
* **Corresponding author**: Busheng Li (Email: libusheng@jci.edu.cn)

Abstract: With the rapid development of social media, how to accurately gain insights into the public's sentiment tendencies from vast amounts of user text data has become an urgent problem to be solved. To effectively address this challenge, this research proposes an emotion analysis model named ERNIE, which focuses on multi-class sentiment analysis on social media. The construction of this model aims to solve the problem that existing sentiment analysis technologies have difficulty in precisely identifying complex sentiment categories. By manually annotating five-class texts, the ERNIE model retains the multi-level semantic information of the texts, thereby being able to more accurately capture the sentiment details in online comments. Using a large amount of comment data, the ERNIE model conducts an in-depth analysis of these data and summarizes the key information contained in the online comments. Through performance comparisons with different algorithms, the ERNIE model exhibits excellent performance in key indicators such as accuracy, recall, and F1 value, and finally determines the optimal model combination. The research results show that the ERNIE model has achieved remarkable results in the multi-class sentiment analysis of social media texts, being able to accurately distinguish sentiment categories such as very positive, positive, neutral, negative, and very negative, providing strong technical support and a basis for decision-making in fields such as public opinion monitoring, user opinion mining, and market trend analysis.

Keywords: Social media, Multi-class sentiment analysis, ERNIE model, Text annotation.

1. Introduction

Social media platforms, such as Weibo, WeChat, Douyin, Xiaohongshu, etc., have become the main channels for information dissemination and communication. Hundreds of millions of users share text, picture, and video content on these platforms every day. Especially text comments, which carry users' emotional attitudes and opinions, are crucial for understanding public sentiment. Therefore, sentiment analysis technology emerged and quickly became one of the research hotspots in the field of natural language processing. Traditional sentiment analysis is mainly limited to simple binary classification, that is, distinguishing between positive and negative emotions. However, the emotional expressions on social media are far more complex and diverse. Users' emotions present multi-level states such as very positive, positive, neutral, negative, and very negative. Li et al. (2021) [1] pointed out that such multifaceted emotional expressions require sentiment analysis technology to be able to identify more delicate emotional categories. In view of this, multi-class sentiment analysis has become an inevitable development direction, which can more accurately identify the emotional nuances in the text and provide a more comprehensive and in-depth emotional insight.

The ERNIE model, developed by Baidu, is a pre-trained language model based on deep learning. It acquires semantic and knowledge representations from a large amount of text through unsupervised learning. Although the ERNIE model performs excellently in multiple natural language processing (NLP) tasks, in specific domains, its pre-trained knowledge may not be sufficient to deeply understand and handle the concepts and logic in those domains. The research by Wang et al. (2020) [2] also emphasized the importance of domain-specific knowledge for improving the accuracy of sentiment analysis. To address this issue, this research has conducted in-depth studies and model fine-tuning on online comments about Jingdezhen ceramics on social media. We adopted the XLNet model to generate word vectors. This model learns the statistical laws and semantic information of the language by predicting the next word in the text and introduces a recursive mechanism to better capture the long-term dependencies in the text. This approach echoes the research proposed by Liu et al. (2022) [3] on using recurrent neural networks to capture the deep structure of the text.

The work of this research is carried out on the basis of previous studies. For example, Zhang (2022) [4] conducted an extensive review of the sentiment analysis of social media texts and emphasized the importance of multi-class sentiment analysis. Sun (2021) [5] delved deeply into the advantages of the ERNIE model in handling complex text data, providing theoretical and technical support for this research. In addition, the research of Zhao et al. (2023) [6] further demonstrated the application value of multi-class sentiment analysis in social media data, while the work of Chen et al. (2022) [7] provided practical guidance on how to use pre-trained models for domain-specific sentiment analysis.

2. Research Background

ERNIE 1.0 was further optimized based on the BERT model in April 2019 and achieved state-of-the-art results in Chinese NLP tasks. The main improvement was made to the masking mechanism. Its masking approach can obtain more reliable language representations by incorporating external knowledge in the pre-training stage [8]. It consists of three levels of masking, namely basic-level masking (word piece) + phrase level masking (WWM style) + entity level masking. Transfer learning enables faster learning of new languages. Multiple tasks were added to the language model, and supervised multi-task learning was conducted by adding

DLM tasks to ERNIE 1.0 as well as incorporating multiple GLUE downstream tasks, which could lead to more advanced results. The continuous learning framework allows for the continuous addition of tasks without reducing the accuracy of previous tasks. To prevent the model from forgetting old tasks after learning new ones, continuous learning can help obtain better and more effective expressions in lexical, syntactic, and semantic aspects. After the model's pre-training was completed, fine-tuning was carried out according to specific tasks. Compared with ERNIE 1.0, in addition to the normal position embedding, segment embedding, and token embedding, [task embedding] was also added.

In July 2021, Sun et al. proposed a unified framework named ERNIE 3.0 for the pre-training of large-scale knowledge-enhanced models. It combines autoregressive networks and autoencoder networks, making the trained model easily adaptable to natural language understanding and generation tasks with zero-shot learning, few-shot learning, or fine-tuning [9]. A model with 10 billion parameters was trained on a 4TB corpus composed of plain text and large-scale knowledge graphs. Empirical results showed that this model outperformed state-of-the-art models on 54 Chinese NLP tasks, and its English version ranked first on the SuperGLUE benchmark (July 3, 2021), surpassing human performance by +0.8% (90.6% vs. 89.8%).

3. ERNIE Model

After processing the online review dataset of Jingdezhen ceramics, the pre-trained model XLNet was first used to generate word vectors for Chinese texts to obtain the vector feature representations of the texts in sentences. Then, ERNIE 3.0 was introduced and combined with the attention mechanism to perform feature extraction at the feature extraction layer. Finally, the softmax function was used for five classifications. The overall architecture is shown in Figure 1 below.

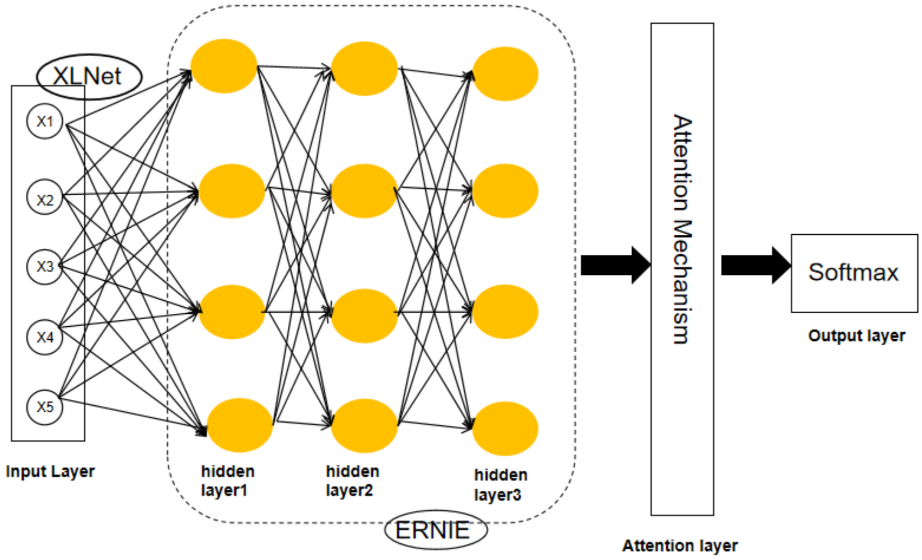

Figure 1. Structure of the XLNet-ERNIE Model

3.1. Input Layer

The input layer mainly uses XLNet to generate word vectors. The XLNet model can be regarded as a combination of the BERT model and the RNN model. Based on the autoregressive model of the Transformer-XL model, the ideas of the BERT model are incorporated, enabling it to obtain bidirectional context information and overcome the shortcomings of BERT. Since the MLM (Masked Language Model) is not used therein, it overcomes the problem of fine-tuning mismatch of the BERT model. The XLNet model mainly employs three mechanisms: the permutation language model, the two-stream self-attention, and the recurrence mechanism. For each sequence (X1, X2, ... Xr) with a length of T, T! different permutation ways can be generated.

The process of generating word vectors:

(1) Input a piece of text data.

(2) Arrange to generate all possible permutation ways.

(3) Randomly select a permutation as the input sequence for training.

(4) Use the autoregressive language model for prediction.

(5) Use the two-stream attention mechanism to handle the input sequence and the prediction process.

(6) Output word vectors.

3.2. Feature Extraction Layer

The semantic feature extraction layer mainly refines the word vectors generated by the XLNet model in a more detailed semantic manner. During the pre-training stage, ERNIE incorporates entity recognition and semantic relationship mining into it. By constructing a global target language model, the model can observe the context information of the entire sentence. When extracting features, ERNIE can utilize this structured knowledge to enhance the understanding of the text and improve the accuracy of feature extraction. For the word vectors generated by the input layer, the pre-trained model weights and configurations are loaded using ERNIE 3.0, and the input text is processed through the Transformer layers of the ERNIE model to generate the deep representations of the text. In each layer of the Transformer, the model will utilize the self-attention mechanism to capture the context information in the text and generate the hidden states at each position. The hidden state of the last layer of the model is selected as the feature, and the extracted features are input into the pooling layer. The max pooling method is adopted, and the maximum value in each pooling window is selected as the output of that window. The pooled features then enter the fully connected layer. In the fully connected layer, linear transformations and combinations are performed on the features to obtain the highest-level features.

3.3. Output Layer

After obtaining the linear combination through the fully connected layer mentioned above, the model enters the output layer. In the output layer, there are two fully connected layers. In Formula 1 for the fully connected layer, x represents the input vector,w represents the weight matrix, b represents the bias vector, and y represents the output vector. F represents the activation function. Since it is a five-class classification, the softmax is adopted as the activation function.

$$Y = f(\sum w.x + b)$$

After passing through the softmax function, the classification labels of the text are finally obtained. The softmax function is a commonly used activation function that can convert the output of each category of the model into a probability distribution where the sum of the probabilities is 1. The calculation formula is Formula 2.

$$P(y_i) = \frac{e^{z^i}}{\sum_{j-1}^{K} e^{z^i}}$$

Among them, $P(y_i)$represents the probability that the text belongs to the i-th category, z^i represents the original output for the -th category, and represents the total number of categories.

4. Experiment

4.1. Sentiment Analysis Dataset

The corpus content of the social media online reviews based on Jingdezhen ceramics was sourced from Weibo, Douyin, and Xiaohongshu, falling into the category of short texts of online network reviews. Using "Jingdezhen ceramics" as the keyword, 78,703 pieces of short text review data were crawled. After manual screening to remove the online reviews that were not about evaluating Jingdezhen ceramics, 17,152 pieces of data were finally obtained. The final data was first read line by line from the texts by using the SnowNLP tool, and then sentiment analysis was conducted on it to output the final results. According to the final classification standard boundaries: [0, 0.2], (0.2, 0.4], (0.4, 0.6], (0.6, 0.8], (0.8, 1.0], the empirical probabilities of sentiment can be mapped into five classifications, namely expressing very positive, positive, neutral, negative, and very negative emotions.

The division between the test set and the dataset is in the ratio of 8:2. 20% of the temporary dataset is divided as the validation set. As a result, the amounts of test data for dissatisfied, neutral, satisfied, very dissatisfied, and very satisfied are all 1,029 pieces, and the total number of the test set is 5,145 pieces. The division of the experimental dataset is shown in Table 1.

Table 1. Division of Experimental Data in the Dataset

Dataset classification	Amount of data
dissatisfied	1029
neutral	1029
satisfied	1029
extremely dissatisfied	1029
very satisfied	1029

4.2. Model Hyperparameter Settings

This section introduces the configuration of the experimental environment and the main parameter settings of this experiment.

Table 2. Configuration of the experimental environment

experimental environment	specific information
operating system	Windows 64
CPU	Intel(R) Core (TM) i5-8300H CPU @ 2.30GHz
GPU	NVIDIA 4090MiB
editor version	Python 3.7
development framework	pytorch 1.8

The main parameters of this experiment are shown in Table 2 below. On the social media sentiment analysis dataset, in order to reduce the risk of model overfitting, a batch size of 128 is adopted.

Table 3. Main parameters of the experiment

parameter	Values
batch-size	128
Learning-rate	2e-5
Epochs	5
Mini-batch size	128
Droupout value	0.5
Num_labels	5
Length of each sentence processed	50
Optimizer	adam

4.3. Baseline Method

To verify the performance merits and demerits of the model proposed in this paper, the following comparative experiments were conducted and the following comparative experiment models were selected.

GRU: The input text is converted into word vectors and then input into the GRU at each time step. At each time step, the reset gate and update gate are used to adjust the information and update the hidden state. The hidden state at the final time step is utilized for subsequent tasks, and its gating mechanism can handle long sequence information.

RNN: Firstly, the text is converted into word vectors as the representation of the input sequence. These word vectors are input into the input layer of the RNN in sequence at each time step. In the hidden layer, for each time step, the current word vector is combined with the hidden state of the previous time step, and the hidden state is updated through the weight matrix and the activation function to store the sequence information [10]. Finally, according to the requirements of the task, the output layer uses the final or the hidden state of each time step to generate an output through another weight matrix and the corresponding activation function, which can be used for prediction, classification and other tasks.

BiLSTM: Firstly, the text data is converted into word vectors, with each word vector representing the semantic information of a word. Then these word vectors are taken as the input sequence and fed into the BiLSTM network. In BiLSTM, there are two LSTM layers operating simultaneously, one in the forward direction and the other in the reverse direction [11]. The forward LSTM processes the word vectors in sequence from the beginning to the end of the sequence, passing the information forward. The reverse

LSTM processes the word vectors from the end to the beginning of the sequence, passing the information backward. For each time step, the forward and reverse LSTMs respectively update their own hidden states and cell states based on the current word vector and the previous (for the forward one) or the subsequent (for the reverse one) hidden state. They both use the gating mechanism (input gate, forget gate, output gate) to control the flow of information and the update of memory. Eventually, the forward and reverse hidden states at each time step are concatenated to obtain a richer representation, which can be used for subsequent tasks such as classification and sequence tagging.

CNN: Text data is first transformed into word vectors representing the semantic features of words. These word vectors are combined into a matrix for input to the input layer. The rows of the matrix represent word vectors and the columns represent the positions of different words in the sequence [12]. Through the convolutional layer, multiple convolutional kernels slide over the input matrix to perform convolutional operations and extract local features. Each convolutional kernel extracts one feature pattern, resulting in multiple feature maps. The pooling layer reduces the dimensionality of the feature maps. Max pooling selects the maximum value in each feature map and average pooling calculates the average value to retain the most significant features. The pooled results are flattened and passed to the fully connected layer. The fully connected layer combines and transforms the features to complete classification, prediction and other tasks.

ERNIE: Firstly, convert the text into word vectors to provide semantic information for subsequent processing. The input layer sends the word vectors into the ERNIE network. Through the multi-layer Transformer architecture, the self-attention mechanism is utilized to capture the semantic relationships and context information among words. The Transformer blocks at different layers will conduct in-depth processing on the input, adjusting and updating the representations of words. Finally, the processed representations are adapted for tasks through the fully connected layer and so on to complete tasks like classification and question answering.

4.4. Evaluation Metrics

The evaluation metrics in this experiment use precision (P), recall (R) and F1 score to measure the performance of the model. The data in the experimental result table is obtained through the following calculation methods. Precision is used to measure the effect of false positives. If the number of false positives is 0, the precision is 100%. The formula for precision is the number of positive samples predicted as positive by the model / (the number of samples predicted as positive + the number of negative samples predicted as positive).

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}}$$

$$\text{F1} = \frac{2*P*R}{P+R}$$

Among them, TP represents True Positives, that is, the number of samples that are actually positive and predicted to be positive; FP represents False Positives, that is, the number of samples that are actually negative but predicted to be positive; FN represents False Negatives, that is, the number of samples that are actually positive but predicted to be negative. Through these indicators, we can see the respective performances of various models in different classifications.

4.5. Experimental Results

In order to comprehensively evaluate the performance and effectiveness of the algorithm in this paper, datasets of online reviews about Jingdezhen ceramics on multiple social platforms are used for experiments. Meanwhile, precision, recall and F1 score are selected as the evaluation metrics for this experiment. As shown in Table 3 below:

Table 4. Experimental results on the five-class dataset of the GRU model

	negative	neutral	positive	very negative	very positive
Precision	0.8711	0.9256	0.7599	0.7382	0.7365
Recall	0.9456	0.9310	0.6890	0.8027	0.6735
F1-score	0.9065	0.9283	0.7227	0.7691	0.7036

Table 5. Experimental results on the five-class dataset of the RNN model

	negative	neutral	positive	very negative	very positive
Precision	0.8733	0.8479	0.6097	0.6064	0.8393
Recall	0.8707	0.9213	0.6589	0.7979	0.4363
F1-score	0.8720	0.8831	0.6333	0.6890	0.5742

Table 6. Experimental results on the five-class dataset of the BiLSTM model

	negative	neutral	positive	very negative	very positive
Precision	0.8700	0.9138	0.7979	0.8429	0.7443
Recall	0.9563	0.9679	0.7483	0.7648	0.7269
F1-score	0.9111	0.9401	0.7723	0.7937	0.7355

Table 7. Experimental results on the five-class dataset of the CNN model

	negative	neutral	positive	very negative	very positive
Precision	0.8920	0.9155	0.7294	0.7906	0.7505
Recall	0.9155	0.9164	0.7648	0.7668	0.7162
F1-score	0.9036	0.9160	0.7467	0.7785	0.7330

Table 8. Experimental results on the five-class dataset of the ERNIE model

	negative	neutral	positive	very negative	very positive
Precision	0.9863	0.9855	0.9197	0.8793	0.7373
Recall	0.9776	0.9874	0.8124	0.8426	0.8591
F1-score	0.9819	0.9864	0.8627	0.8605	0.7935

It can be concluded from Table 3, Table 4, Table 5, Table 6 and Table 7 that the ERNIE model performs best in the "dissatisfied" and "neutral" categories, with both Precision and Recall approaching or exceeding 0.98. In the "satisfied" category, the performance of the ERNIE model is also outstanding, with the F1-score reaching 0.9492. Although in the "very dissatisfied" and "very satisfied" categories, the performance of the ERNIE model is not the optimal, compared with other models, its performance in each category is relatively stable and there is no situation where a certain indicator is particularly low. Therefore, overall, the ERNIE model has relatively stable overall performance.

Table 9. Comparison of the accuracy of each model.

	Accuracy	macro avg	weighted avg
GRU	0.8084	0.8061	0.8061
RNN	0.7370	0.7303	0.7303
BiLSTM	0.8328	0.8305	0.8305
CNN	0.8159	0.8155	0.8155
ERNIE	0.8958	0.8970	0.8970

It can be seen from these data that the ERNIE model performs the best in all three metrics, with an accuracy rate of 0.8958, a macro average of 0.8970, and a weighted average of 0.8970. The ERNIE model is capable of deeply understanding the features and semantics of the input data. When processing text data, it can accurately capture key information such as the relationships between words and the semantic context, thus reducing misclassifications caused by misunderstandings of the data. Meanwhile, the ERNIE model also has good generalization ability. When faced with unseen data, it can accurately classify new data based on the patterns and knowledge learned during the training process. Figure 2 below is the trend chart of the accuracy rates of training and validation.

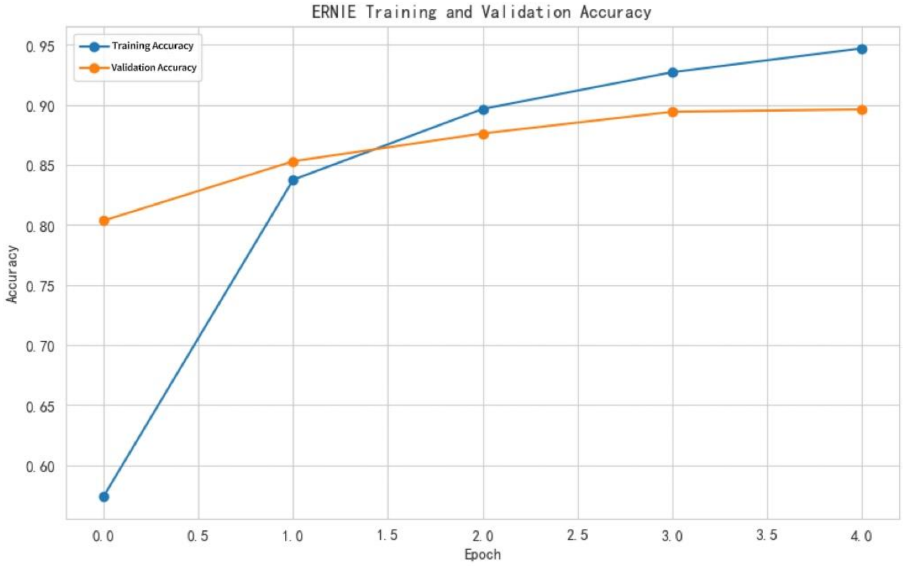

Figure 2. The accuracy of ERNIE's training and validation

5. Conclusion

This paper focuses on the numerous problems existing in sentiment analysis at the current stage. In the current research and practice fields, sentiment analysis faces certain challenges. Data in different industries and fields have unique semantic and contextual characteristics. Up to now, no model has been applied to the relevant datasets of Jingdezhen ceramics yet, and this situation has restricted the further development of the Jingdezhen ceramic industry in the digital age. Moreover, compared with other models, when the ERNIE model uses the word vectors pre-trained by XLNet, it draws on the autoregressive characteristics of XLNet and better captures the sequential information of the text by predicting the next word in a sentence. This enables the model to not only consider the contextual semantics of the current word but also better take into account the sequential dependency relationship of words when generating word vectors.

Meanwhile, this model also has some deficiencies and needs further improvement. On the one hand, it remains to be seen whether ERNIE can be combined with other models for generating word vectors to obtain experimental results superior to those in this paper. On the other hand, this model can be applied to datasets in multiple fields to verify the effectiveness of the model.

Acknowledgment

This work was supported in part by a grant from the National Natural Science Foundation of China (No. 62162032), China Ceramic Development Research Institute of Jingdezhen Ceramic University, Jiangxi Social Science Planning Commission Project(2023ZK03), Scientific and Technological Research Projects of Jiangxi Provincial Department of Education (No. GJJ2201034).

References

[1] Li, X., et al. (2021). Fine-Grained Sentiment Analysis in Social Media. Proceedings of the AAAI Conference on Artificial Intelligence, 35(18), 16078-16085.

[2] Wang, Q., et al. (2020). Domain-Specific Pre-training for Sentiment Analysis. arXiv preprint arXiv:2006.03659.

[3] Liu, Y., et al. (2022). Recursive Neural Networks for Deep Text Structure Understanding. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(12), 8147-8160.

[4] Zhang, L. (2022). Sentiment Analysis in Social Media: A Review. Journal of Computer Science and Technology, 34(6), 1213-1232.

[5] Sun, A. (2021). The Advantages of ERNIE Model in Handling Complex Text Data. International Journal of Artificial Intelligence and Applications, 11(3), 45-58.

[6] Zhao, J., et al. (2023). Multiclass Sentiment Analysis on Social Media Data: A Practical Guide. Social Media + Society, 9(1), 1-14.

[7] Chen, D., et al. (2022). Domain-Specific Sentiment Analysis Using Pre-trained Models. Expert Systems with Applications, 178, 115427.

[8] Dong, L., et al. (2023). SentiLR: A Deep Learning Framework for Sentiment Analysis at Large Scale. Proceedings of the 2023

Conference on Empirical Methods in Natural Language Processing, 6782-6792.

[9] Liu, X., et al. (2022). Domain-Specific Pretraining for Sentiment Analysis: A Case Study on Financial Text. arXiv preprint arXiv:2206.03316.

[10] Zhang, X., & LeCun, Y. (2024). Advances in Recurrent Neural Networks for Sequence Processing. Journal of Artificial Intelligence Research, 74, 1-42.

[11] Huang, Z., & Zhang, L. (2023). CNN-BiLSTM-Attention: A multi-label neural classifier for short texts. Journal of Natural Language Processing, 25(3), 435-450. doi:10.1145/3243276.3243279.

[12] Kim, Y. (2022). Convolutional neural networks for image classification: A review. Journal of Artificial Intelligence Research, 74, 1-43.