Information Dynamics and Applications
https://www.acadlore.com/journals/IDA

# An Integrated BERT-XGBoost Framework for Open-Source Intelligence Classification in Aerospace Technology

Suping Yu, Weiwei Mao*

College of Computer and Information Engineering, Luoyang Institute of Science and Technology, 471023 Luoyang, China

* Correspondence: Weiwei Mao (mw116@lit.edu.cn)

**Received:** 10-28-2024 **Revised:** 11-22-2024 **Accepted:** 12-06-2024

**Citation:** S. P. Yu and W. W. Mao, "An integrated BERT-XGBoost framework for open-source intelligence classification in aerospace technology," *Inf. Dyn. Appl.*, vol. 3, no. 4, pp. 234–244, 2024. https://doi.org/10.56578/ida030403.

© 2024 by the author(s). Published by Acadlore Publishing Services Limited, Hong Kong. This article is available for free download and can be reused and cited, provided that the original published version is credited, under the CC BY 4.0 license.

**Abstract:** Open-source intelligence in aerospace technology often contains lengthy text and numerous technical terms, which can affect classification accuracy. To enhance the precision of classifying such intelligence, a classification algorithm integrating the Bidirectional Encoder Representations from Transformers (BERT) and Extreme Gradient Boosting (XGBoost) models was proposed. Initially, key features within the intelligence were extracted through the deep structure of the BERT model. Subsequently, the XGBoost model was utilised to replace the final output layer of BERT, applying the extracted features for classification. To verify the algorithm's effectiveness, comparative experiments were conducted against prominent language models such as Text Recurrent Convolutional Neural Network (TextRCNN) and Deep Pyramid Convolutional Neural Network (DPCNN). Experimental results demonstrate that, for open-source intelligence classification in aerospace technology, this algorithm achieved accuracy improvements of 1.9% and 2.2% over the TextRCNN and DPCNN models, respectively, confirming the algorithm's efficacy in relevant classification tasks.

**Keywords:** Text classification; BERT model; XGBoost model; Aerospace technology; Open-source intelligence

## 1 Introduction

The classification of open-source intelligence in aerospace technology is the initial step in the field of aerospace intelligence. Enhancing the classification accuracy of aerospace technology open-source intelligence through advanced machine learning techniques has become an urgent issue in the field of aerospace intelligence studies.

Earlier studies on the classification of aerospace technology open-source intelligence primarily focused on traditional machine learning methods [1]. For instance, a server-client model for the aerospace text classification system was developed by combining the Bayesian algorithm and web technology [2]. A classification method for aerospace intelligence based on the Support Vector Machine (SVM) was investigated, proposing a multi-classification process to handle various categories of data [3]. However, these methods often required manual feature engineering, and the representation of text did not accurately capture inter-textual relationships, resulting in suboptimal classification accuracy. The extensive development and application of deep learning techniques have significantly improved text classification performance. A TextRCNN-A algorithm based on the attention mechanism was proposed, which effectively captures contextual information and mitigates word ambiguity [4]. These findings highlight that accurately capturing contextual semantic information is crucial for enhancing classification accuracy in the classification of aerospace technology open-source intelligence.

Traditional text vector representation methods, such as the bag-of-words model [5], included approaches like one-hot encoding and the term frequency-inverse document frequency (TF-IDF) model [6]. However, these models failed to consider word-to-word associations and produced high-dimensional vectors, often leading to sparsity issues in matrices. Later, neural network-based text vectorisation models were introduced. For instance, the Word2Vec model [7] represents word relationships, while the Global Vectors for Word Representation

(GloVe) model [8] leverages global statistical information to obtain word vectors through matrix decomposition. Despite their advantages, these models exhibit limitations in representing polysemous words. As a solution, the Elmo model [9, 10], which dynamically adjusts pre-trained word vectors based on actual contextual data, enables the learning of complex lexical features, including syntax and semantics. The Transformer model [11, 12], incorporating an encoder-decoder and attention mechanism, offers the significant advantage of high parallelisation efficiency. The BERT model, based on the Transformer architecture, provides advanced word and sentence representation capabilities [13, 14]. BERT provides a straightforward interface for downstream natural language processing tasks while also serving as a pre-trained language model [15] for document vectorisation, demonstrating substantial transferability.

Through analysis of the textual data in aerospace technology open-source intelligence, it has been observed that the texts are often lengthy and contain numerous specialised terms unique to the field. These characteristics limit classification accuracy, as existing models for classifying aerospace technology open-source intelligence are generally unable to effectively focus on specialised terminology or extract the core content features from intelligence data. To address this, a hybrid model based on BERT and the XGBoost model was proposed in this study for the classification of aerospace technology open-source intelligence. BERT was employed to extract key features from the intelligence data, and these features were subsequently classified using the XGBoost model, which is known for its effective classification performance, thereby enhancing the accuracy of classification results.

## 2 Classification of Aerospace Technology Open-Source Intelligence Based on BERT-XGBoost

### 2.1 Overall Structure of the BERT-XGBoost Hybrid Model

The proposed classification method for aerospace technology open-source intelligence based on the BERT-XGBoost hybrid model consists of two primary components: feature extraction of aerospace technology open-source intelligence based on BERT, and feature classification based on XGBoost. After vectorising the intelligence data, it was input into the BERT model, where the output was a fixed-length feature vector representing the aerospace technology open-source intelligence.

In the overall structure of the hybrid model, aerospace technology open-source intelligence texts, along with their respective classification labels, were used to train the BERT classification model. The model was iteratively adjusted using gradient descent to achieve high classification accuracy. Given that the original BERT model's output layer produces a class label for each text, a modification was made by replacing the final output layer with a linear output layer, while maintaining the parameters in the preceding layers, to extract feature vectors.

After the BERT model encoded aerospace technology open-source intelligence into feature vectors, these feature vectors, along with the corresponding classification labels, were then fed into the XGBoost model for further training. Since XGBoost is an ensemble algorithm based on decision trees [16, 17], the input feature vectors were structured into trees. Once the initial tree was constructed, subsequent trees were employed to correct errors in the existing trees. The process of tree construction halted when no further improvement was observed in the model's classification outcomes. The total score of the leaf nodes across all trees corresponded to the evaluation score for each classification category [18], and the category with the highest score was selected as the final classification.

By using the XGBoost model to classify feature vectors extracted by BERT, the hybrid model was better able to capture the underlying relationships between the data and labels, resulting in more efficient classification performance.

### 2.2 Feature Extraction of Aerospace Technology Open-Source Intelligence Based on BERT

The classification of aerospace technology open-source intelligence is significantly influenced by the presence of numerous specialised terms. As the multi-head self-attention mechanism can effectively focus on key information within a text, enabling the identification of specialised terms in intelligence data, the BERT model—incorporating the multi-head self-attention mechanism—was utilised to extract features from aerospace technology open-source intelligence, thereby achieving more accurate text classification. The structure of the model's feature extraction process is illustrated in Figure 1. A single piece of aerospace technology open-source intelligence data was input into the BERT model, which then processed the data and output a corresponding text feature vector. This feature vector was subsequently used as the input for the XGBoost model in the hybrid model.

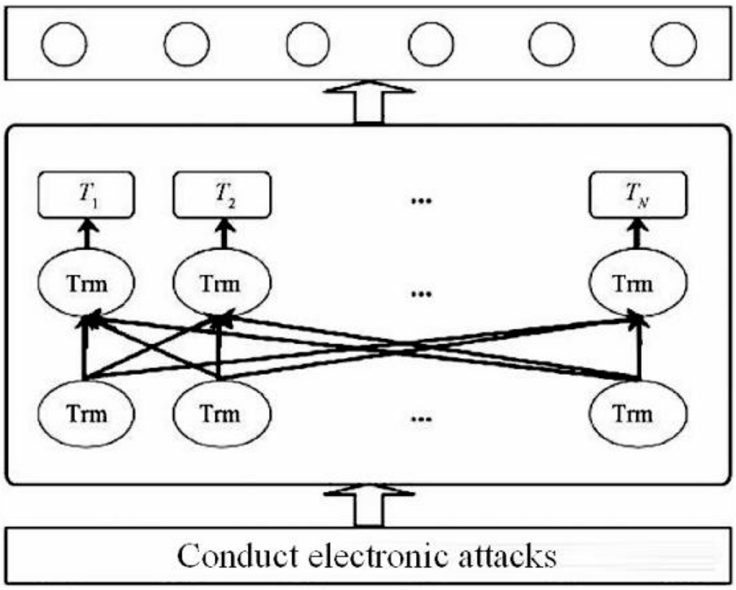


**Figure 1.** Structure of the model's feature extraction process

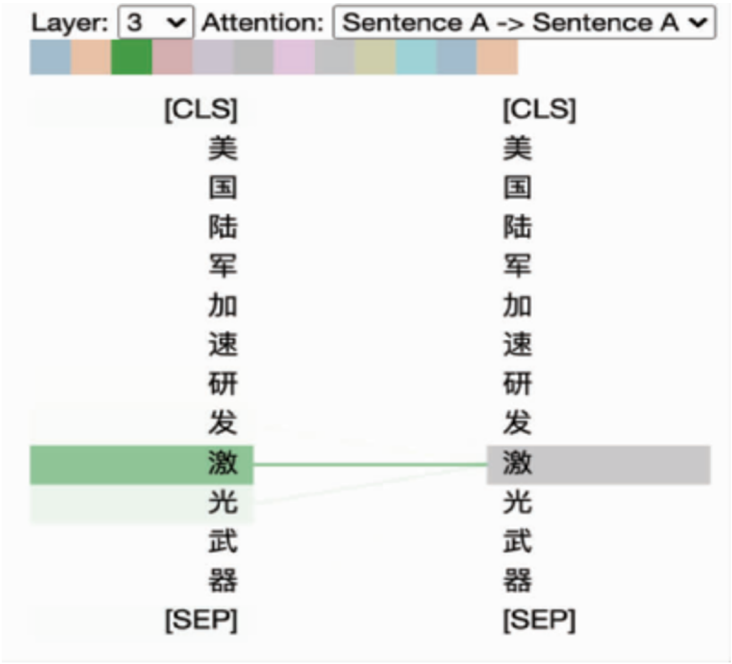


**Figure 2.** Visualisation result of one head

The BERT model contains bidirectional Transformer encoding layers, which are effective at capturing bidirectional relationships within sentences. The Transformer encoding layer employs the multi-head self-attention mechanism to enhance the diversity of attention [19]. Multi-head self-attention considers various semantic contexts, enabling different integration patterns of target word vectors with other word vectors in the text. Its computation formula is as follows:

$$MultiHead(Q, K, V) = Concat(head1, ..., headi, ...)^{*}W_O \tag{1}$$

where, *headi* = *Attention* $(Qi, Ki, Vi) = \mathrm{softmax}((Qi \cdot KT)/dk)^{*}Vi$; *headi* represents the output vector of the $i$-th head; *Concat* $(\cdot)$ denotes a concatenation function that horizontally combines matrices; $W_O$ is a weight matrix that assigns weights to the output vectors; $Qi$, $Ki$, and $Vi$ are matrices formed by linearly mapping input vectors; $dk$ is the dimension of the $K$ vector.

The multi-head attention mechanism can apply multiple linear transformations and dot-product computations

iteratively to achieve the multi-head structure. This approach enables the model to learn contextual information in various subspaces, thus capturing key information in the text more comprehensively. For example, the Chinese phrase in Figure 2 and Figure 3, which means “The U.S. Army accelerated the development of laser weapons,” was used to visualise attention, demonstrating the ability of multi-head attention to capture contextual information. The visualisation results are shown in Figure 2 (one head) and Figure 3 (two heads).

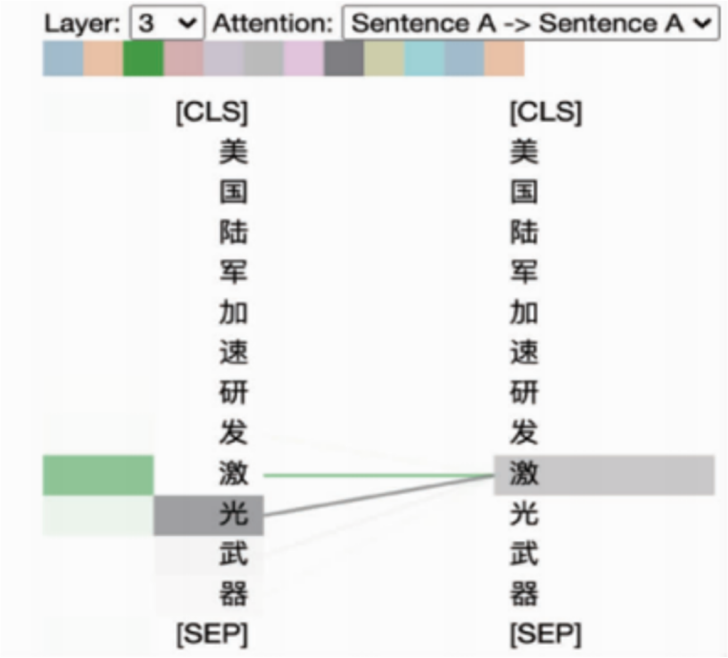


**Figure 3.** Visualisation result of two heads

In Figure 2 and Figure 3, differently coloured blocks represent the results of distinct attention heads. The darker the shade of a particular colour, the higher the attention value, indicating a stronger dependency between the characters at each end of the line. As shown in Figure 2 (one head), the character “激” is only able to learn dependencies with itself. In contrast, in Figure 3 (two heads), the character “激” captures dependencies with both characters of “激” and “光”, thereby demonstrating that multi-head attention can extract information across various representational subspaces.

**Table 1.** Key parameters of the XGBoost model

| Parameter | Value | Description |
|---|---|---|
| Booster | Gbtree | Model selection for each iteratione |
| Num_eslass | 5 | Number of categoriese |
| Max_depth | 8 | Maximum depth-of each tree |
| Objective | Multi: softmax | Definition-of theloss function |
| Gamma | 0.1 | Minimum decrease-in-loss function fornode classificatione |
| Subsample | 0.6 | Proportion of samples randomly’selected per tree |
| Colsample bytree | 0.7 | Control-of therropgrtion-of columns randomly-selected for-each tree |
| Eta | 0.1 | Weight reduction-atreach step to improyemodel-stability |

These results indicate that the multi-head self-attention mechanism effectively focuses on specialised terminology in aerospace technology open-source intelligence, enhancing feature extraction within the model. This process prepares the features for subsequent classification using the XGBoost model.

### 2.3 Feature Classification of Aerospace Technology Open-Source Intelligence Based on XGBoost

The XGBoost model can process features in parallel at a granular level. Compared to softmax and linear classifiers, the XGBoost model leverages data features more effectively to fit the data. Therefore, XGBoost was utilised to classify feature vectors extracted by the BERT model. XGBoost operates within a gradient-boosting framework, aiming to minimise the residuals of the $k$-th tree. The objective function was solved using the second-order Taylor expansion as follows:

$$Obj^{(\mathrm{t})} \approx \sum_{i=1}^{n} l\left(yi, yi^{(\mathrm{t}-1)}\right) + gift(xi) + \frac{1}{2} hift^{2}(xi) \Bigg] + \Omega(ft) + constant \tag{2}$$

where, $l\left(yi, yi^{(t-1)}\right)$ represents the loss between the true and predicted values; $gi$ and $hi$ are coefficients in the second-order Taylor expansion and are known constants during the training of the $k$-th tree; $ft(xi)$ is the prediction of the $k$-th tree; $\Omega(ft)$ denotes the tree complexity term to prevent excessive tree growth during training; *constant* represents the sum of all constant terms in the expansion.

The XGBoost model includes numerous parameters, and proper tuning is essential for effective model training. The primary parameters are listed in Table 1. The "Booster" parameter was set to gbtree, indicating a tree structure suitable for feature classification in this study. Given that there are five categories in the aerospace technology open-source intelligence data, the "Num_class" parameter was set to 5. "Max_depth," which indicates tree depth, is typically between 5 and 10; in this study, a value of 8 was selected. The multi-classification objective determines the "Objective" parameter to be multi:softmax, while other parameters were configured according to standard settings.

**Algorithm 1** Training process of the BERT-XGBoost hybrid model applied to aerospace technology open-source intelligence classification

Input $\mathrm{data}_{\mathrm{original}}$: raw text data of aerospace technology open-source intelligence; $\mathrm{model}_{\mathrm{pre}}$: pre-trained model BERT-Base, Chinese; Output test set error rate: $\mathrm{Test}_{\mathrm{error}}$; $\mathrm{model}_{\mathrm{BERT}}$: the BERT model after training; $\mathrm{model}_{\mathrm{XGBoost}}$: the XGBoost model after training;

model = Load($\mathrm{model}_{\mathrm{pre}}$); // Load the pre-trained Chinese model
data,label =DataProcessing($\mathrm{data}_{\mathrm{original}}$); // Data pre-processing
$\mathrm{data}_{\mathrm{input}}$ =DataVectorization($\mathrm{model}_{\mathrm{pre}}$,data); // Handle the text data for input into the BERT model
$\mathrm{data}_{\mathrm{train}}$,$\mathrm{data}_{\mathrm{dev}}$,$\mathrm{data}_{\mathrm{test}}$ = SplitData($\mathrm{data}_{\mathrm{input}}$); // Split the dataset
$\mathrm{model}_{\mathrm{BERT}}$ = Train(model,$\mathrm{data}_{\mathrm{train}}$,$\mathrm{data}_{\mathrm{dev}}$)
features =OutputFeature($\mathrm{data}_{\mathrm{train}}$,$\mathrm{model}_{\mathrm{BERT}}$) ; // BERT outputs feature vectors in categories from the training set
training the XGBoost model $\mathrm{model}_{\mathrm{XGBoost}}$ with features;
set the maximum number of iterations epochs;
set the initial best error rate $\mathrm{best}_{\mathrm{error}}$;
initial epoch = 0;
repeat
Train($\mathrm{data}_{\mathrm{train}}$,$\mathrm{data}_{\mathrm{dev}}$,$\mathrm{model}_{\mathrm{BERT}}$); // Train the BERT model
features =OutputFeature($\mathrm{data}_{\mathrm{test}}$,$\mathrm{model}_{\mathrm{BERT}}$); // BERT outputs feature vectors for the test set
result = Prediction(features,$\mathrm{model}_{\mathrm{XGBoost}}$); // XGBoost outputs classification result
$\mathrm{test}_{\mathrm{error}}$ =ComputError( result); // Compute the error rate on the test set
if $\mathrm{test}_{\mathrm{error}} <=$ $\mathrm{best}_{\mathrm{error}}$ then
$\mathrm{best}_{\mathrm{error}} \leftarrow \mathrm{test}_{\mathrm{error}}$;
Save($\mathrm{model}_{\mathrm{BERT}}$);
features =OutputFeature($\mathrm{data}_{\mathrm{train}}$,$\mathrm{model}_{\mathrm{BERT}}$);
training the XGBoost model $\mathrm{model}_{\mathrm{XGBoost}}$ with features;
Save($\mathrm{model}_{\mathrm{XGBoost}}$);
end if
Save($\mathrm{test}_{\mathrm{error}}$);
until (epoch$>$epochs).

Given the complexity of feature vectors extracted by the BERT model from aerospace technology open-source intelligence, the XGBoost model was selected for classifying these vectors to better account for the

impact of these complex features on classification accuracy. The aerospace technology open-source intelligence data were vectorised and represented as $x$, which was then used to train the BERT model, ultimately outputting the feature vector $Output_{vec}$. The original classification label $y$ and the feature vector produced by BERT together formed the input for the XGBoost model, denoted as $Input_x$. The final classification result was output by the XGBoost model. The specific formulas are as follows:

$$Output_{vec} = BERT(x) \tag{3}$$

$$Input_x = (\ Output_{vec}, y) \tag{4}$$

$$Output_{final} = XGBoost(Input_x) \tag{5}$$

The pseudocode for the training process of the hybrid model is shown.

## 3 Experimental Analysis

### 3.1 Experimental Data

The experimental data used in this study consist of publicly available scientific and industrial information texts provided by a platform, sourced from various national defence technology websites. The total dataset contains 61,027 entries, primarily categorised into aerospace, shipbuilding, ordnance, aviation, and electronics industries. Following the preprocessing methods outlined [20], irrelevant information in the aerospace technology open-source intelligence datawas removed, and the cleaned text content was utilised for subsequent classification tasks.

Statistical analysis revealed a substantial disparity in the data volume across categories. Therefore, two experimental groups—balanced and imbalanced datasets—were designed for comparison. In the balanced dataset, each category (aerospace, shipbuilding, ordnance, aviation, and electronics industries) contains 4,053 entries, and the total number of entries is 20,265. In the imbalanced dataset, the data volumes are 13,454, 15,804, 4,053, 20,846, and 6,870 entries, respectively, for a total of 61,027. The experimental data were split into training, validation, and test sets in an 8:1:1 ratio.

### 3.2 Experimental Procedure

During model training, comparative experiments were conducted based on variations in the length of feature vectors extracted by BERT. Considering the significant differences in data volume across categories, experiments were designed for both balanced and imbalanced datasets. Finally, the hybrid model was compared with other mainstream language models on the same dataset to evaluate its classification performance.

#### 3.2.1 Training of the hybrid model

In this study, the BERT module within the hybrid model uses the “BERT-Base, Chinese” model, with a network structure comprising 12 layers, 768 hidden neurons, a 12-head attention mechanism, and a total of 110 million parameters. During model training, the imbalanced dataset was utilised. The model parameters were set as follows: a dropout rate of 0.1, 3 epochs, and a learning rate of $5e^{-5}$. The extracted feature vectors were input into the XGBoost model to obtain accurate classification results. The training loss and accuracy for BERT, as well as the training error rate for XGBoost, are illustrated in Figure 4 and Figure 5.

From Figure 4, it can be observed that after more than 5,400 iterations, the loss and accuracy for both the training and validation sets of the BERT model exhibit smaller fluctuations and gradually stabilise. The accuracy on the validation set remains nearly constant, indicating that the model has achieved optimal training fit. As shown in Figure 5, the training error rate of the XGBoost model is relatively low from the beginning of training, likely due to the BERT model’s ability to accurately extract features from the aerospace technology open-source intelligence data, which allows the XGBoost model to achieve good classification performance in subsequent stages. The final accuracy of the hybrid model on the test set reached 90.01%, representing a 1.5% improvement compared to the accuracy obtained when using the BERT model alone under the same parameters. This result indicates that the combination of BERT and XGBoost contributes to enhanced classification accuracy for aerospace technology open-source intelligence.

Since the length of the text feature vectors extracted by the BERT module in the hybrid model varies, the amount of information also differs, potentially affecting classification accuracy. Therefore, a comparative experiment was conducted to explore how classification accuracy changes with different feature vector lengths. The results are shown in Figure 6.

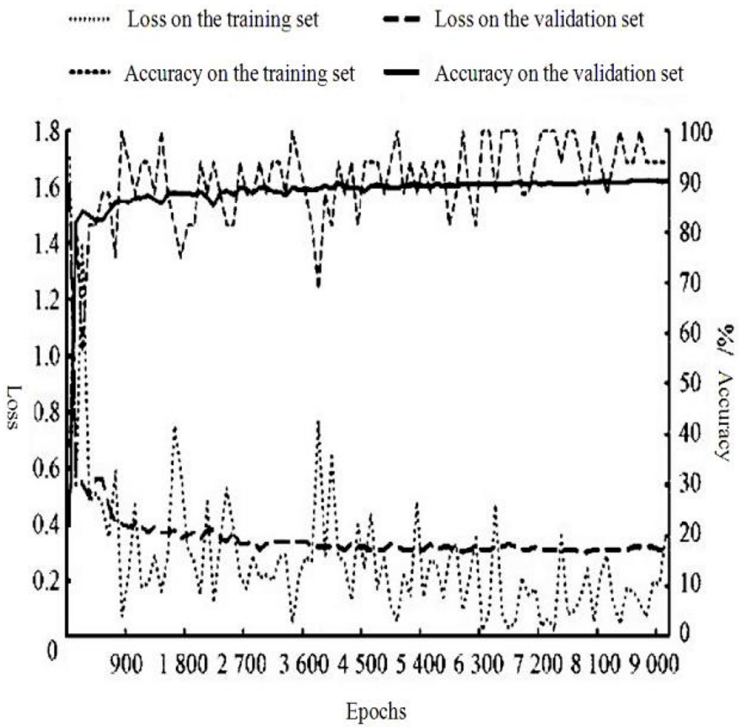


**Figure 4.** Training loss and accuracy of the BERT module in the hybrid model

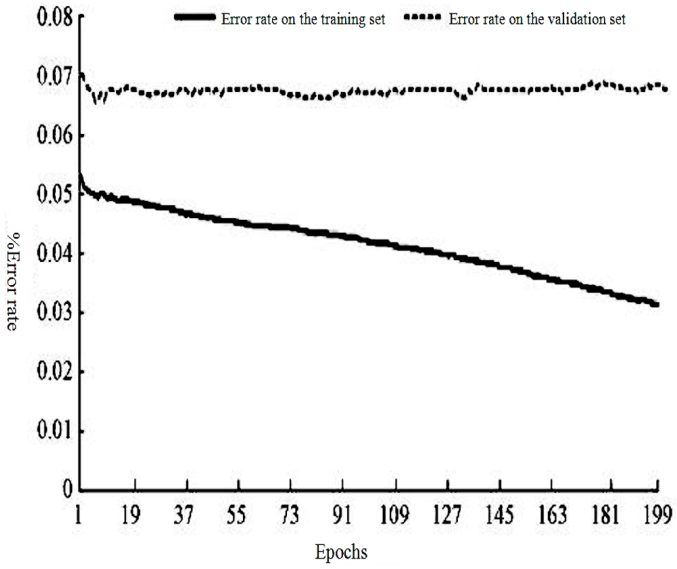


**Figure 5.** Training error rate of the XGBoost module in the hybrid model

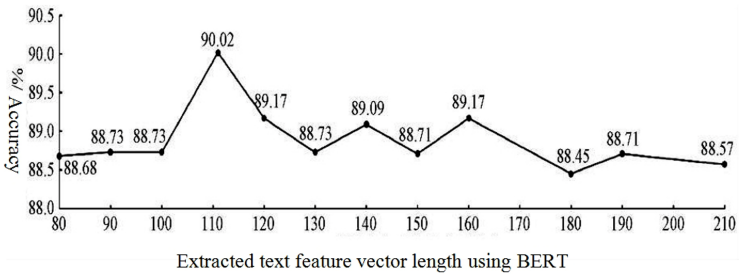


**Figure 6.** Classification accuracy of the model with different text feature vector lengths

As shown in Figure 6, as the length of the extracted text feature vectors increases, the classification accuracy of the model initially rises and then fluctuates downward. The highest classification accuracy, reaching 90.02%, is achieved when the feature vector length is approximately 110. When the feature vector length exceeds 120, it is speculated that the excessive length may lead to feature saturation, thereby impacting the model's classification performance and causing the accuracy to exhibit a fluctuating decline.

### 3.3 Comparison of Balanced and Imbalanced Datasets

To investigate the impact of dataset balance on the classification of aerospace technology open-source intelligence, experiments comparing F1-scores under balanced and imbalanced datasets were designed. Based on the findings in Section 3.2.1, the best classification accuracy is achieved when the feature vector length extracted by the BERT module is 110 dimensions. To conduct a comprehensive comparison, feature vector lengths both greater than and less than 110 dimensions were selected for experimentation, setting maximum text feature vector lengths to 100, 110, and 128 dimensions, respectively. The test set accuracy of the hybrid model on both balanced and imbalanced datasets is shown in Figure 7. The accuracy of the hybrid model is higher under the imbalanced dataset compared to the balanced condition, likely due to the larger data volume in the imbalanced dataset, which may have enhanced the model's learning effectiveness and, consequently, its classification accuracy. The F1-score, a metric used to evaluate model performance, was also used for comparison, with results shown in Table 2. For the categories "ordnance industry" and "electronics industry," the F1-scores were higher under the balanced dataset than under the imbalanced dataset. This is likely because these categories have the least data, and the presence of larger data volumes in other categories under the imbalanced condition may have introduced classification interference. In contrast, for the categories "aviation industry" and "shipbuilding industry," the F1-scores were higher under the imbalanced dataset, likely because the larger data volumes for these categories under the imbalanced condition allowed for more comprehensive model training and richer feature learning, resulting in higher F1-scores.

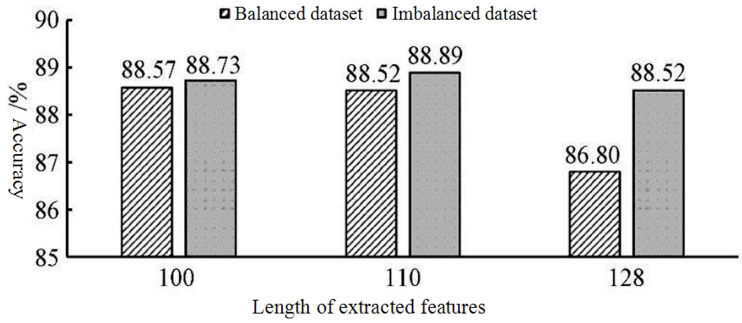


**Figure 7.** Classification accuracy of the hybrid model on different dataset types

**Table 2.** Comparison of F1-scores across different categories and dimensions

| Category | 100 Dimensions | | 110 Dimensions | | 128 Dimensions | |
|---|---|---|---|---|---|---|
| | Balanced F1-score | Imbalanced F1-score | Balanced F1-score | Imbalanced F1-score | Balanced F1-score | Imbalanced F1-score |
| Aerospace industry | 0.894 | 0.886 | 0.891 | 0.882 | 0.878 | 0.884 |
| Ordnance industry | 0.903 | 0.861 | 0.899 | 0.860 | 0.900 | 0.862 |
| Electronics industry | 0.886 | 0.825 | 0.888 | 0.829 | 0.857 | 0.833 |
| Aviation industry | 0.853 | 0.887 | 0.856 | 0.891 | 0.823 | 0.881 |
| Shipbuilding industry | 0.893 | 0.921 | 0.893 | 0.924 | 0.883 | 0.919 |

Note: In the table header, "100 dimensions," "110 dimensions," and "128 dimensions" refer to the length of the text feature vectors extracted by the BERT module.

A comprehensive analysis of the hybrid model's accuracy on the test set and the F1-scores for each category reveals that the F1-score differences between the balanced and imbalanced datasets are not significant. However,

the test set accuracy is consistently higher for the imbalanced dataset compared to the balanced dataset. This result indicates that the imbalanced dataset is more advantageous for the model's ability to learn the features of aerospace technology open-source intelligence texts.

### 3.4 Comparison of the Hybrid Model with Mainstream Language Models

Before the hybrid model proposed in this study, substantial research was conducted on language models, including models such as TextCNN based on Convolutional Neural Networks (CNNs) and TextRNN based on Bidirectional Long Short-Term Memory (BiLSTM). Using the datasets described in previous sections, key hyperparameters were set uniformly for several mainstream language models: dropout rate of 0.5, 20 iterations, learning rate of e-3, and maximum text length of 256. The performance of these mainstream language models on the aerospace technology open-source intelligence classification task was then compared with that of the proposed hybrid model. In addition to the identical parameter settings mentioned, the BERT module in the hybrid model was set to extract feature vectors with a maximum length of 110 dimensions, as concluded in Section 3.2.1.

The accuracy results on the test set for the BERT_XGBoost, BERT, TextRCNN, DPCNN, TextRNN, TextCNN, FastText, and Transformer models were 90.01%, 88.50%, 88.16%, 87.83%, 87.27%, 85.54%, 85.06%, and 80.57%, respectively. It can be observed that the BERT_XGBoost hybrid model achieves the highest accuracy on the test set compared to other mainstream models. Furthermore, the classification accuracy of the hybrid model is 1.5% higher than that of the BERT model alone, indicating that the hybrid model provides better classification performance and enhances the classification accuracy of aerospace technology open-source intelligence. Additionally, the F1-scores of each model across different categories were compared, as shown in Table 3.

Table 3 demonstrates that the BERT_XGBoost hybrid model achieves the highest F1-scores across all five categories, indicating that its classification performance is superior in these categories. In summary, the comparison of accuracy on the test set and F1-scores across categories shows that the hybrid model outperforms other mainstream language models. This finding suggests that the hybrid model is more suitable for classifying aerospace technology open-source intelligence texts and improves classification performance in this domain.

**Table 3.** Comparison of F1-scores for models across these categories

| Category | F1-scores of Each Model | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | BERT_XGBoost | BERT | TextRCNN | DPCNN | TextRNN | TextCNN | FastText | Transformer |
| Aerospace industry | 0.898 | 0.890 | 0.869 | 0.876 | 0.871 | 0.850 | 0.850 | 0.773 |
| Ordnance industry | 0.862 | 0.846 | 0.852 | 0.844 | 0.834 | 0.807 | 0.804 | 0.677 |
| Electronics industry | 0.860 | 0.814 | 0.821 | 0.812 | 0.794 | 0.792 | 0.774 | 0.726 |
| Aviation industry | 0.904 | 0.886 | 0.882 | 0.877 | 0.874 | 0.854 | 0.852 | 0.808 |
| Shipbuilding industry | 0.924 | 0.917 | 0.922 | 0.916 | 0.913 | 0.898 | 0.893 | 0.887 |

## 4 Conclusion

The primary focus of this study is to address the classification of aerospace technology open-source intelligence. Given the lengthy nature of these texts and the presence of numerous specialised terms, a classification method based on the BERT-XGBoost hybrid model was proposed. The BERT model was employed to extract features from the open-source intelligence texts, and the XGBoost model was utilised to classify these features. A comparison of accuracy with several mainstream language models on the same dataset verified that the hybrid model effectively enhances classification accuracy for aerospace technology open-source intelligence. However, this study did not investigate the impact of image data present in the dataset on classification performance, and the dataset used was relatively small. In future work, it is intended to integrate image data into the hybrid model to enrich sentence feature representations. Additionally, the classification performance of the hybrid model will be explored on larger-scale datasets.

**Funding**

This study was funded by the Postgraduate Education Reform and Quality Improvement Project of Henan Province (Grant No.: YJS2024AL134).

**Data Availability**

The data used to support the findings of this study are available from the corresponding author upon request.

**Conflicts of Interest**

The authors declare no conflict of interest.

**References**

[1] S. Soussi, G. S. Collins, P. Jüni, A. Mebazaa, E. Gayat, and Y. Le Manach, "Evaluation of biomarkers in critical care and perioperative medicine: A clinician' s overview of traditional statistical methods and machine learning algorithms," *Anesthesiology*, vol. 134, no. 1, pp. 15–25, 2020. https://doi.org/10.1097/ALN.0000000000003600

[2] P. O. Prakash and A. Jaya, "WS-BD-based two-level match: Interesting sequential patterns and Bayesian fuzzy clustering for predicting the web pages from weblogs," *Comput. J.*, vol. 63, no. 2, pp. 322–336, 2020. https://doi.org/10.1093/comjnl/bxz132

[3] X. C. Li, X. L. Tang, S. C. Qiu, X. Deng, H. M. Wang, and Y. Tian, "Subdomain adversarial network for motor imagery EEG classification using graph data," *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 8, no. 1, pp. 327–336, 2023. https://doi.org/10.1109/TETCI.2023.3301385

[4] Z. T. Hu, C. H. Hu, H. R. Yang, and W. W. Shuai, "Unsupervised multi-modal image translation based on the squeeze-and-excitation mechanism and feature attention module," *High Technol. Lett.*, vol. 30, no. 1, pp. 23–30, 2024.

[5] T. C. Zhang, Q. Li, and X. Liu, "An object detection and classification method for underwater visual images based on the bag-of-words model," *Proc. Inst. Mech. Eng. M J. Eng. Marit. Environ.*, vol. 237, no. 2, pp. 487–497, 2023. https://doi.org/10.1177/14750902221096984

[6] J. Sulaksono, R. A. Ramadhani, and R. K. Niswatin, "Automatic article summary with the term frequency-inverse document frequency algorithm for information on elderly health," *J. Comput. Theor. Nanosci.*, vol. 17, no. 2–3, pp. 1511–1513, 2020. https://doi.org/10.1166/jctn.2020.8833

[7] Q. Gao, X. Huang, K. Dong, Z. T. Liang, and J. Wu, "Semantic-enhanced topic evolution analysis: A combination of the dynamic topic model and word2vec," *Scientometrics*, vol. 127, no. 3, pp. 1543–1563, 2022. https://doi.org/10.1007/s11192-022-04275-z

[8] M. Q. Zhou, D. Liu, Y. H. Zheng, Q. S. Zhu, and P. Guo, "A text sentiment classification model using double word embedding methods," *Multimed. Tools Appl.*, vol. 81, no. 14, pp. 18 993–19 012, 2022. https://doi.org/10.1007/s11042-020-09846-x

[9] D. P. Schmidt, M. Haghshenas, P. Mitra, C. Wang, P. K. Senecal, F. Tagliante, and L. M. Pickett, "The Eulerian Lagrangian Mixing-Oriented (ELMO) model," *Int. J. Multiph. Flow*, vol. 152, p. 104041, 2022. https://doi.org/10.1016/j.ijmultiphaseflow.2022.104041

[10] E. A. Hassan, A. M. Elsaid, M. M. Abou-Elzahab, A. M. El-Refaey, R. Elmougy, and M. M. Youssef, "The potential impact of MYH9 (rs3752462) and ELMO1 (rs741301) genetic variants on the risk of nephrotic syndrome incidence," *Biochem. Genet.*, vol. 62, no. 2, pp. 1304–1324, 2024. https://doi.org/10.1007/s10528-023-10481-y

[11] W. X. Sima, D. X. Peng, M. Yang, P. T. Sun, B. Y. Zou, and Z. Xiong, "Reversible wideband hybrid model of two-winding transformer including the core nonlinearity and EMTP implementation," *IEEE Trans. Ind. Electron.*, vol. 68, no. 4, pp. 3159–3169, 2021. https://doi.org/10.1109/TIE.2020.2977544

[12] F. L. Xu, H. K. Zhao, F. Y. Hu, M. F. Shen, and Y. F. Wu, "A road segmentation model based on mixture of the convolutional neural network and the transformer network," *Comput. Model. Eng. Sci.*, vol. 135, no. 2, pp. 1559–1570, 2023. https://doi.org/10.32604/cmes.2022.023217

[13] Y. T. Feng, Z. F. Ma, P. F. Duan, and S. S. Luo, "Automated vulnerability detection of blockchain smart contacts based on BERT artificial intelligent model," *China Commun.*, vol. 21, no. 7, pp. 237–251, 2024. https://doi.org/10.23919/JCC.ja.2023-0189

[14] K. Ma, Y. J. Tan, M. Tian, X. J. Xie, Q. J. Qiu, S. F. Li, and X. Wang, "Extraction of temporal information from social media messages using the BERT model," *Earth Sci. Inform.*, vol. 15, no. 1, pp. 573–584, 2022. https://doi.org/10.1007/s12145-021-00756-6

[15] X. W. Liao, Y. Z. Huang, P. Yang, and L. Chen, "A statistical language model for pre-trained sequence labeling: A case study on Vietnamese," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 21, no. 3, pp. 1–21, 2021. https://doi.org/10.1145/3483524

[16] S. M. Mousavi, V. Majidnezhad, and A. Naghipour, "A new intelligent intrusion detector based on ensemble of decision trees," *J. Ambient Intell. Humaniz. Comput.*, vol. 13, no. 7, pp. 3347–3359, 2022. https://doi.org/10.1007/s12652-019-01596-5

[17] Á. Delgado-Panadero, J. A. Benítez-Andrades, and M. T. García-Ordás, "A generalized decision tree ensemble based on the NeuralNetworks architecture: Distributed Gradient Boosting Forest (DGBF)," *Appl. Intell.*, vol. 53, no. 19, pp. 22 991–23 003, 2023. https://doi.org/10.1007/s10489-023-04735-w

[18] M. Li, C. Chen, Y. Z. Cao, P. Y. Zhou, X. Deng, P. Liu, Y. L. Wang, X. Y. Lv, and C. Chen, "CIABNet: Category imbalance attention block network for the classification of multi-differentiated types of esophageal cancer," *Med. Phys.*, vol. 50, no. 3, pp. 1507–1527, 2023. https://doi.org/10.1002/mp.16067

[19] S. Aathilakshmi, A. K. Britto, and R. Vimala, "Digital filtering using multipliers on intelligence extraction and power-line removal from electromyogram signals," *J. Med. Imaging Health Inform.*, vol. 10, no. 1, pp. 30–37, 2020. https://doi.org/10.1166/jmihi.2020.2835

[20] S. Shambhu, D. Koundal, P. Das, V. T. Hoang, K. Tran-Trung, and H. Turabieh, "Computational methods for automated analysis of malaria parasite using blood smear images: Recent advances," *Comput. Intell. Neurosci.*, vol. 2022, no. 1, p. 3626726, 2022. https://doi.org/10.1155/2022/3626726