Hindawi
BioMed Research International
Volume 2020, Article ID 5491963, 13 pages
https://doi.org/10.1155/2020/5491963

Research Article

Application of BERT to Enable Gene Classification Based on Clinical Evidence

Yuhan Su,[1] **Hongxin Xiang**,[1] **Haotian Xie**,[2] **Yong Yu**,[1] **Shiyan Dong**,[3] **Zhaogang Yang**,[3] **and Na Zhao**[1]

[1]*National Pilot School of Software, Yunnan University, Kunming, 650091, China*
[2]*Department of Mathematics, The Ohio State University, Columbus, OH 43210, USA*
[3]*Department of Radiation Oncology, University of Texas Southwestern Medical Center, Dallas, TX 75390, USA*

Correspondence should be addressed to Zhaogang Yang; zhaogang.yang@utsouthwestern.edu and Na Zhao; zhaonayx@126.com

Received 31 July 2020; Revised 31 August 2020; Accepted 7 September 2020; Published 7 October 2020

Academic Editor: Zhiguo Zhou

The identification of profiled cancer-related genes plays an essential role in cancer diagnosis and treatment. Based on literature research, the classification of genetic mutations continues to be done manually nowadays. Manual classification of genetic mutations is pathologist-dependent, subjective, and time-consuming. To improve the accuracy of clinical interpretation, scientists have proposed computational-based approaches for automatic analysis of mutations with the advent of next-generation sequencing technologies. Nevertheless, some challenges, such as multiple classifications, the complexity of texts, redundant descriptions, and inconsistent interpretation, have limited the development of algorithms. To overcome these difficulties, we have adapted a deep learning method named Bidirectional Encoder Representations from Transformers (BERT) to classify genetic mutations based on text evidence from an annotated database. During the training, three challenging features such as the extreme length of texts, biased data presentation, and high repeatability were addressed. Finally, the BERT+abstract demonstrates satisfactory results with 0.80 logarithmic loss, 0.6837 recall, and 0.705 *F*-measure. It is feasible for BERT to classify the genomic mutation text within literature-based datasets. Consequently, BERT is a practical tool for facilitating and significantly speeding up cancer research towards tumor progression, diagnosis, and the design of more precise and effective treatments.

1. Introduction

Nowadays, genomic, transcriptomic, and epigenomic studies have been benefited from the development of inexpensive next-generation sequencing technologies, which play essential roles in exploring tumor biology [1–3]. Tumors usually possess heterogeneities, and the genomic profiling of tumors normally contains various types of genetic mutations [4–7]. However, only a small proportion of mutation genes are involved in boosting tumor growth, whereas most of them are neutral and irrelevant to tumor progression [8, 9]. Characterization and identification of cancer driver genes are important a in clinical trials to reveal tumor pathogenesis and facilitate diagnosis, prognosis, and personalized therapy [10–13]. Despite the importance of gene classification, the following analysis is challenging due to the significant amount of manual work for interpretating genomics, which is time-consuming, laborious, and subjective. With the increasing availability of electronic unstructured and semistructured data sources, automatically categorizing documents has emerged as a potential tool for information organization. Machine learning (ML), as a promising optimization tool, has been widely used in credit scoring, fraud detection, retailers, market segmentation, manufacturing, education, and healthcare [14–18]. Hence, using ML to analyze clinical contextual data automatically is favorable [19–21]. For example, in 1986, Swanson first discovered the undiscovered links in a large number of scientific literature [22]. Also, Marcotte et al. used Naive Bayesian classification to

classify the literature focusing on protein-protein interaction [23].

Despite the achievements traditional ML methods have made, potential drawbacks such as low accuracy exist when they are applied on clinical text classification. In 2018, Google proposed that the BERT method achieved state-of-the-art results in 11 projects, including text classification [24]. Descriptions about clinical research acadamic papers show high similarities , which blurs the classification boundary, increases the inconsistancy, and lows the accuracy. Consequently, the advanced ML methods, such as Light Gradient Boosting Machine (LightGBM), has been proposed to enable gene multiclassification based on complex literature [25]. Nevertheless, these methods are limited by complex calculations when applied to large-scale datasets, particularly for genomic-related literature datasets that contain millions, or billions, of annotated training examples [26, 27]. In addition, the performances of ML are dependent on feature extraction that requires professional knowledge and long-term processing [28–31].

To overcome these difficulties, deep learning (DL) has emerged to handle large-scale and complex datasets since its performance increases with the enlargement of datasets [32–34]. For example, the convolutional neural networks (CNN) [35], recurrent neural networks (RNN) [36], and their combination [37] have been applied to the sentence classification successfully. Also, In 2018, Google proposed that the BERT method achieved state-of-the-art results in 11 projects, including text classification.

Hence, we fine-tune the BERT model to classify mutation effects (9 classes) using an expert-annotated oncology knowledge base. Our BERT method is developed based on the original BERT model and is capable of obtaining different syntactic and semantic information. Three main characters of training datasets including extreme length of text entry, data imbalance, and repetitive description are engineered during training challenges. We propose three truncation methods including abstract+head, head only, and head+tail to deal with extreme length of text entry and repetitive description. Besides, data imbalance is relieved by negative sampling. Overall, we improve the BERT method to classify complex clinical texts, and obtain 0.8074 logarithmic loss, 0.6837 recall, and 0.705 F-measure scores.

2. Problem Statement

The treatment of cancer is closely related to the identification of mutant genes [38]. At present, clinicians need review and classify each mutant gene manually according to the evidence in text-based clinical literature, which is a complicated, time-consuming, and error-prone method [39–42]. To solve this problem, Memorial Sloan Kettering Cancer Center (MSKCC) has provided an expert-annotated precision oncology knowledge base with thousands of mutations manually annotated by world-class researchers and oncologists for studying gene classification using computer-based method [43]. On top of that, we design an artificial intelligence algorithm to automatically and accurately classify mutations for avoiding mistakes caused by manual classification, and provide further help for cancer treatments.

In recent years, with the rise of artificial intelligence, natural language processing, which uses linguistics, computers, mathematics, and other scientific methods to communicate between human beings and computers, has developed rapidly [44–46]. Among them, text classification is one of the most basic and critical tasks in natural language processing [47]. Text classification is the process of associating a given text within one or more categories according to characteristics of texts (content or attributes) under a predefined classification system [48–50]. The process of text classification mainly includes three steps. Firstly, the text is preprocessed, then the vector representation of the text is extracted. Finally, the classifier is trained to classify the text [48]. Text classification can be divided into single-label text classification and multilabel text classification according to the number of labels to which the text belongs. The single-label text refers to each text belonging to only one category, while multilabel text refers to each text belonging to one or more categories [51–53]. The calculation formula for text classification can be defined as follows:

$$F(D, C) = \{\text{True, False}\}. \tag{1}$$

In the formula, the collection $D = \{d_1, d_2, \cdots d_n\}$ refers to the set of texts classified, where the ith classified text is represented by $\mathrm{d_i}$, and n is the number of classified texts. The collection $C = \{c_1, c_2, \cdots, c_m\}$ is a collection of predefined classification categories, where the jth category is represented by c_j, and m is the number of predefined categories. F is a function representing a mapping relationship.

Currently, the most common methods for text classification are statistical ML and DL-based methods. Statistical ML methods usually preprocess texts in the first place, then manually extract high-dimensional sparse features. Consequently, they use statistical ML algorithms to obtain classification results. In 1998, Joachims first employed support for vector machine (SVM) in text classification and achieved favorable results [54]. In the following research, many methods based on statistical ML are used in text classification, including Naïve Bayes classifier [55], K-nearest Neighbor method (KNN) [56], decision tree [57], boosting [58], and LightGBM [59]. Among them, LightGBM is widely used in classification problems due to its fast speed, low memory consumption, and relatively high accuracy [60]. Although LightGBM gets good classification results in some scenes, research related to this approach runs basically into bottleneck due to its strong dependence on the effectiveness of features. Also, it is time-consuming and labor-intensive during feature extraction process.

Although the traditional statistical ML models can classify texts faster than the manual method, they require manual feature extraction, which leads to a large amount of labor cost and is difficult to obtain effective features [61–63]. On the other hand, the DL methods are superior to traditional statistical ML methods in terms of text feature expression and automatic acquisition of feature expression capabilities, thus eliminating complex manual feature engineering processes and

reducing possible application costs [64]. As we all know, large-scale pretraining language models have become a new driving force for various natural language processing tasks [65]. For example, BERT models can significantly improve model performance by fine-tuning downstream tasks. Google first proposed the BERT model, and it completely subverted the logic of training word vectors before training specific tasks in natural language processing [24]. Methods of fine-tuning the BERT model, such as extended text preprocessing and layer adjustment, have been proved to improve the results substantially [66]. Wu et al. proposed a conditional BERT method, which can enhance the text classification ability of original BERT method by predicting the conditions of masked words [67]. To sum up, it is feasible to employ the fine-tuned model based on the original BERT to classify genetic mutations.

Hence, we propose an improved BERT model with high classification accuracy after analyzing the MSKCC mutation gene interpretation database thoroughly. We believe this method can be successfully applied to genetic mutation classification. The main contributions of our work are summarized as follows:

(1) The text description of the individual sample shows considered lengths. There are differences in text lengths between different categories of samples. Some categories contain shorter words, while others contain miscellaneous descriptions. Generally, texts in a dataset range from hundreds to thousands of words in length. However, the lengths of the gene mutation in this paper are much longer than usual. We use the BERT method to truncate texts and extract valuable information in the texts using different methods, thus avoiding adverse impacts of excessive differences in text lengths on the results.

(2) There is a deviation of total gene number in all categories. Individual genes are unevenly distributed in different categories. Some genes belong to five or more groups, while others only present in two categories. To solve the vast differences in the number of samples between different categories in the dataset, we choose an undersampled data processing method to balance the data deviations between different categories.

(3) The whole dataset has a high repeated description. Different examples belong to different categories share the same text entry. Some categories show a high correlation, which may lead to low accuracy. To solve this problem, we improve the BERT model and splice the last three layers of the initial model, which increases the accuracy of the model and reduces the running time.

(4) To a certain extent, we illustrate the effectiveness of using DL in the classification of genetic clinical texts. As the data set increases, the DL model represented by BERT will learn the characteristics of the sample better to achieve exceptional results. In the future, DL models will have better performances on similar tasks.

3. Materials and Methods

3.1. Description of Datasets. MSKCC sponsored the training and test datasets in this study for method development and validation. For the past several years, world-class experts have created a clinical evidence annotated precision oncology knowledge database. The annotations contain information about which genes are oncology clinically actionable. We sum up three characteristics of the MSKCC datasets mentioned below:

(i) Textual descriptions of individual samples exhibit considerable lengths. The text lengths among different classes show variabilities. Some of the classes contain shorter words while other classes contain redundant descriptions.

(ii) The overall gene numbers presented among the whole classes show biases. The distribution of individual genes in different classes is unequal. Some genes belong to five classes or more, and some of the genes only fit in two classes.

(iii) High repetitive descriptions exist in the whole datasets. Different samples belong to different classes that share the same text entry. Classes demonstrate high correlations.

3.1.1. Length of Entry Text. It is reasonable to analyze the length of the entry text as a prior task for textual-based classification. We find that extremely long descriptions with massive irrelevant information are correlated with samples (Figure 1). We plot the distribution of text lengths (Figure 2), and our datasets contain more counted words than the normal classification datasets in reviews [68]. Consequently, we examine the distribution of text lengths among different target classes to better understand the uniformity of datasets. Variabilities are demonstrated among different classes (Figure 3). Comparing the density of the length distributions, we divide the classes into three groups. Classes 3, 5, and 6 contain the shortest counted words; classes 1, 2, 4, and 7 exhibite medium counted words; and classes 8 and 9 show the most counts. Overall, two features that increase the task difficulty are attracted: considerable lengths of words and the unequal text length distribution among different classes.

3.1.2. Analysis of the Data Distribution. Analyzing the composition of datasets can help us construct algorithms at an early stage. We sum up the frequency of genes among 9 classes (Figure 4). The 9 classes correspond to mutation effects but are annotated using numbers instead of real textural information to avoid artificial labeling, thus improving the reliability of our algorithms during the training. The true information of these labels is listed in Table 1. The distribution of genes among 9 classes exhibited bias. Genes in class 7 are significantly higher than genes in classes 3, 8, and 9.

We also examin the interactions among different features within target classes. To reduce calculations, we select the top 20 gene types to illustrate the interrelations instead of the whole gene types (Table 2). Selected genes are sorted by

EGFR || we have conducted an analysis of EGFR mutations in glioblastoma by sequencing cDNAs that represent the entire EGFR coding region for each member of a series of tumors containing equal numbers of cases with and without EGFR amplification. A majority of the tumors exhibited common mutations, i.e., Del 19 (40%) or L858R (47%).

KRAS || we note that, surprisingly, this method was able to detect impactful mutations in oncogenes, including KRAS, despite the presence of an endogenous, activating KRAS mutation in A549 cells. KRAS (exon 2) was carried out by fragment analysis and Sanger sequencing.

BRCA1 || interestingly, BRCA1-associated cancers have an altered spectrum of p53 mutations Which may reflect changes in mutagenesis and/or selection for the acquired mutations (17). By contrast, five different BRCA1 constructs (P1749R, M1775R, Y1853X, 5382insC, and▲1751) that contain single amino acid mutations or short deletions (including removal of only the last 11 amino acids in Y1853X) within the C-terminal tandem BRCT domains shifted BRCA1 from the nucleus to the cytoplasm (Figure 1).

Figure 1: The cut-off document views of the datasets.

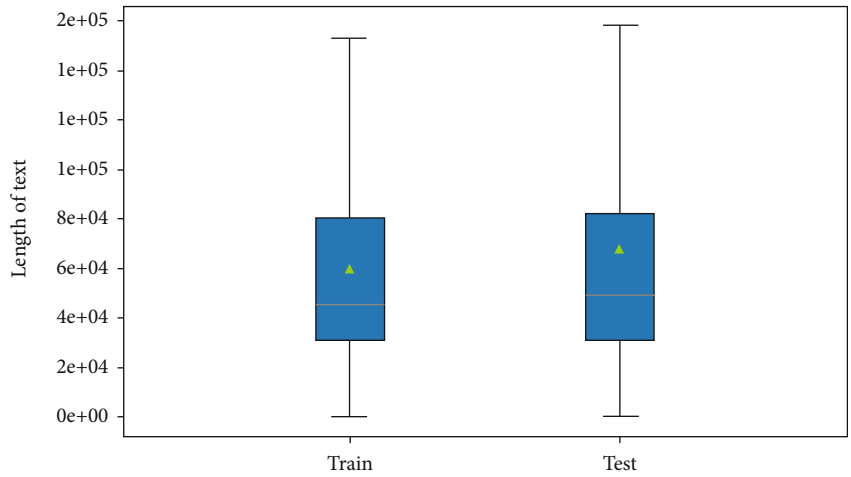

Figure 2: Distribution of the text entry lengths.

classes (Figure 5). The distribution of genes demonstrate huge variabilities among different classes. We find that classes 8 and 9 contain almost none of these genes, and class 3 contain a few of these genes. These distribution biases are in accordance with our previous gene frequency summary based on the whole gene types. Similarly, the trends in classes 1, 2, 4, and 7 correspond to our previous results. These comparable results indicate that the whole datasets are highly associated with selected genes. Consequently, discriminatory differences among classes can impede the feature learning performances of our algorithms and low the accuracy of the text classification.

We further explore the distribution of individual genes within classes, which demonstrates inequitable distributions. For instance, genes such as CDKN2A, PTEN, and TSC2 only present in a limited number of classes (lesser than three). In contrast, BRCA1, ERBB2, FGFR2, and RET are possessed in the majority of classes. Compared with genes only present in a few groups, genes that spread among classes are generally difficult to classify because elaborate texture descriptions can blur the classification standard. Hence, the accuracy of classifications is dependent on the gene compositions. Commonly, genes distributed in lesser classes can show more satisfactory results.

3.1.3. Characteristics of the Datasets. Using typical genes as samples, we find that these typical genes presented in classes demonstrated variabilities. To better recognize these biases and complete potential influences behind them, we conduct a statistical analysis of the whole datasets from the text entry aspects. We find that different samples share the same text entries after extracting common words. The highly repetitive descriptions increase the difficulties of classification, especially when samples in different classes share the same sketches. The worst scenario is the fact that samples belong to different classes that have the same name, but other clue information is missing. For example, five possible mutations of gene BRCA1, the mutation P1749R, M1775R, Y1853X, 5382insC, and Δ1751, may belong to different classes, but their descriptions are close, even in the same sentence. Similarly, two mutations of EGFR, such as Del 19 and L858R, also show in pairs (Figure 1). Hence, we can assume that it is tough to categorize the samples into correct classes by relying on the name of mutations with limited or without other valuable information.

Also, class-dependent word similarities are evaluated using full word lists (Figure 6). Correlation coefficients exhibited high connections (higher than 60%) between classes. Among them, classes 2 and 7 and classes 1 and 4 demonstrate

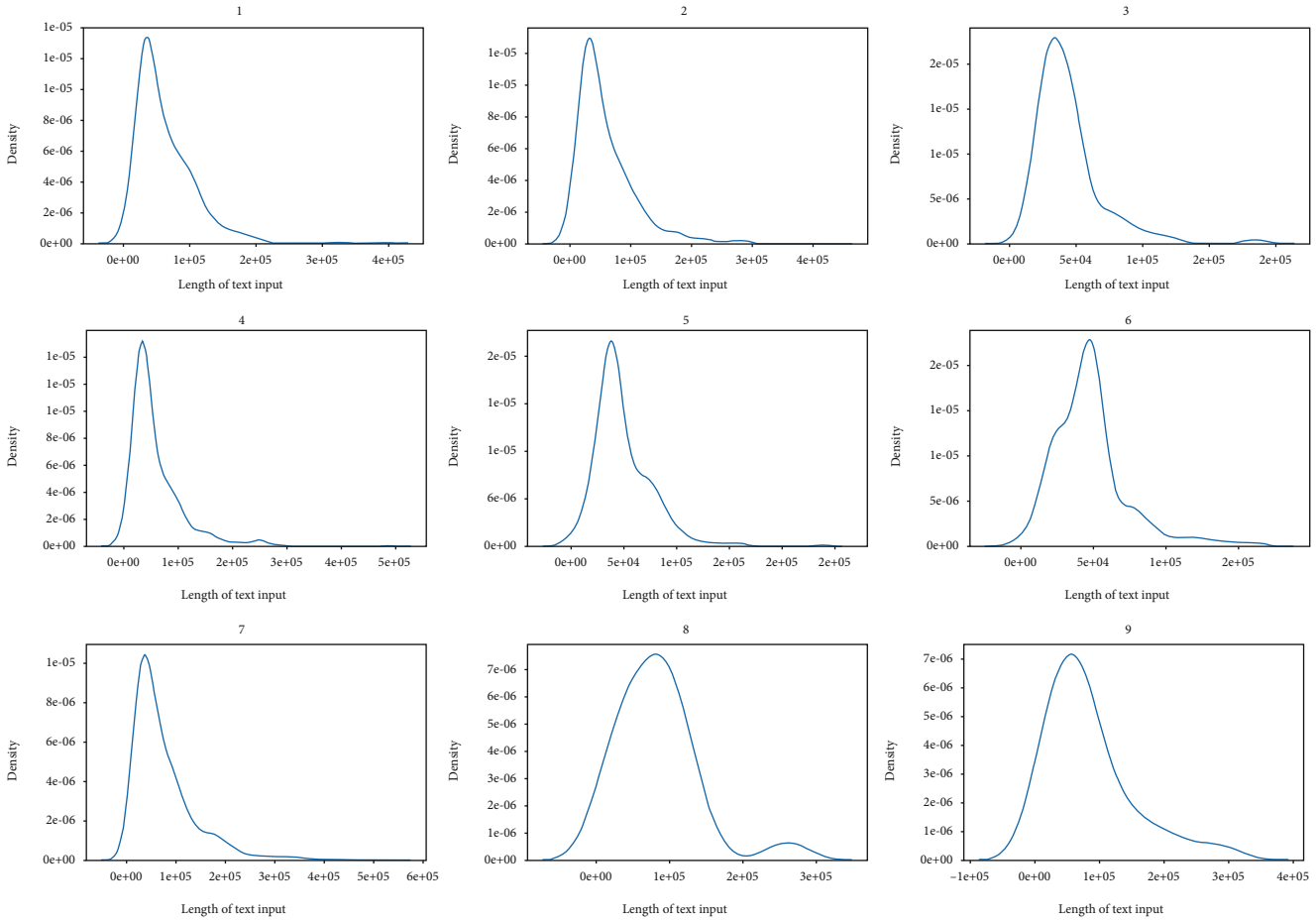

Figure 3: Distribution of the text entry lengths among different classes.

extremely high correlations with 97% and 93% coefficients, respectively. Therefore, we think substantial work needs to be done to clarify samples that share similar descriptions in high correlative classes. Besides, we can not expect high accuracy when classifying samples with these properties.

3.2. BERT. Compared with traditional ML methods, DL demonstrates better performances in text feature expression and automatically obtains feature expression capabilities, thus removing the complicated manual feature engineering process and decreasing its application cost. BERT is a new language representation model based on DL, which was released by the AI team of Google company in October 2018. The BERT model is divided into two parts: pretraining and fine-tuning.

3.2.1. Pretraining of Modified BERT Model. In the pretraining process, a large-scale unlabeled text corpus is used to complete the deep vector representation of text content in the deep bidirectional neural network through an unsupervised training method, thus forming the corresponding text pretraining model. Google has trained two pretrained models. One is the BERT-base model, which includes 12 transformers, 12 self-attention heads, and 768 hidden sizes. The other is the BERT-large model, which contains 24 transformers, 16 self-attention heads, and 1024 hidden sizes. Parameters of BERT-base methods are loaded into the downstream BERT classification model so that our model parameters can be fine-tuned based on these pretrained models, which significantly reduces the convergence time of the model and increases the accuracy of the model. During the pretraining process, BERT randomly masks out, replaces some words, and predicts these missing or replaced words through the remaining ones. The transformer must maintain a distributed representation of each input token. The transformer is likely to remember the word masked without this masking and predicting procedure.

3.2.2. Fine-Tuning of Modified BERT Model. Since the generalization ability of the pretrained model is powerful, the BERT pretrained model can be applied to various downstream tasks after fine-tuning the parameters of the pretrained model. For example, it is possible to meet the needs of a text classification task by adding pooling, full connect, and Softmax function to the output layer sequence of fine-tuned BERT model. The fine-tuning process requires much lesser training resources compared to the pretraining process. The method of fine-tuning BERT model, such as truncation and layer adjustment, has been proved to be capable of improving the result [18]. It implements the process of unsupervised learning through the mask, thereby predicting the vocabulary that will appear in the sentence and

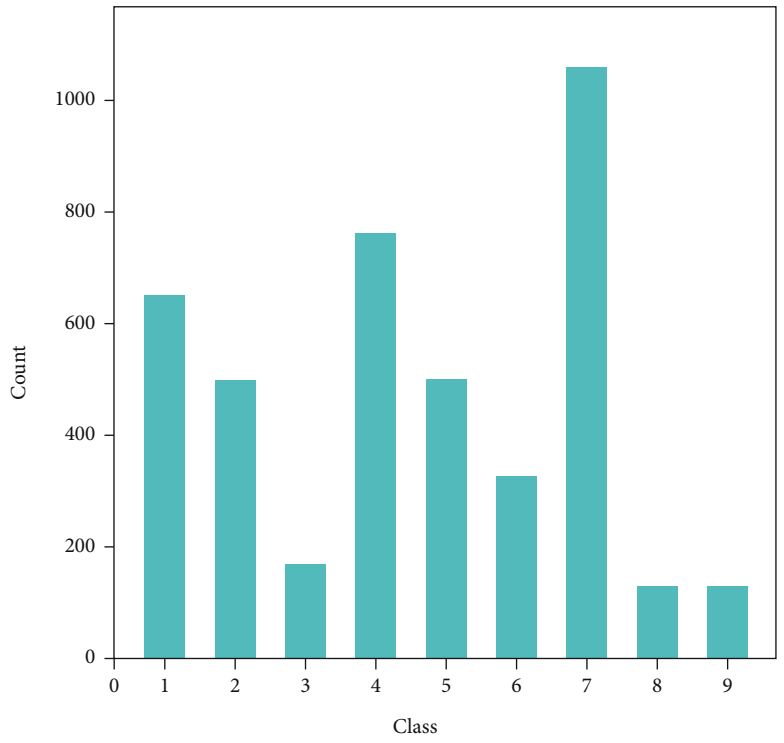

Figure 4: Distribution of the number of genes among 9 classes.

Table 1: Class information corresponds to the annotated number.

Annotated number	Class information
1	Likely loss of function
2	Likely gain of function
3	Neutral
4	Loss of function
5	Likely neutral
6	Inconclusive
7	Gain of function
8	Likely switch of function
9	Switch of function

Table 2: List of top 20 genes in the datasets.

Rank	Gene name	Rank	Gene name
1	EGFR	11	FLT3
2	TP53	12	MTOR
3	CDKN2A	13	MAP2K1
4	ERBB2	14	PTEN
5	PDGFRA	15	BRCA1
6	TSC2	16	BRAF
7	PIK3CA	17	BRCA2
8	FGFR2	18	KIT
9	ALK	19	KRAS
10	VHL	20	RET

understanding the specific meaning of the sentence according to the context.

3.3. Evaluation Equation. This paper evaluates the performances of the model using several evaluation indicators: Logloss, recall (REC), precision (PRE), F1 score, receiver operating characteristic (ROC) curve, and confusion matrix. True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) can be used to calculate some of the indicators mentioned above. TP is the number of categories that are correctly predicted. TN is the number of categories that are correctly predicted as another class. FP is the number of categories that are wrongly predicted. FN is the number of categories that are wrongly predicted as another class.

In multiclassification tasks, Logloss is one of the most common loss functions, where the predicted input is a probability value distribution between 0 and 1, and it can be defined as follows:

$$\text{Logloss} = -\frac{1}{S_n}\sum_{m=1}^{S_n}\sum_{n=1}^{N} y_{mn}\log\left(p(y_{mn})\right), \tag{2}$$

where M is the number of samples and N is the number of classifications. y_{mn} is the predicted result of classification, such as 0 and 1. $p(y_{mn})$ is the predicted probability of y_{mn}.

PRE defines the proportion of genes identified correctly belonging to this type of mutation:

$$\text{PRE} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{3}$$

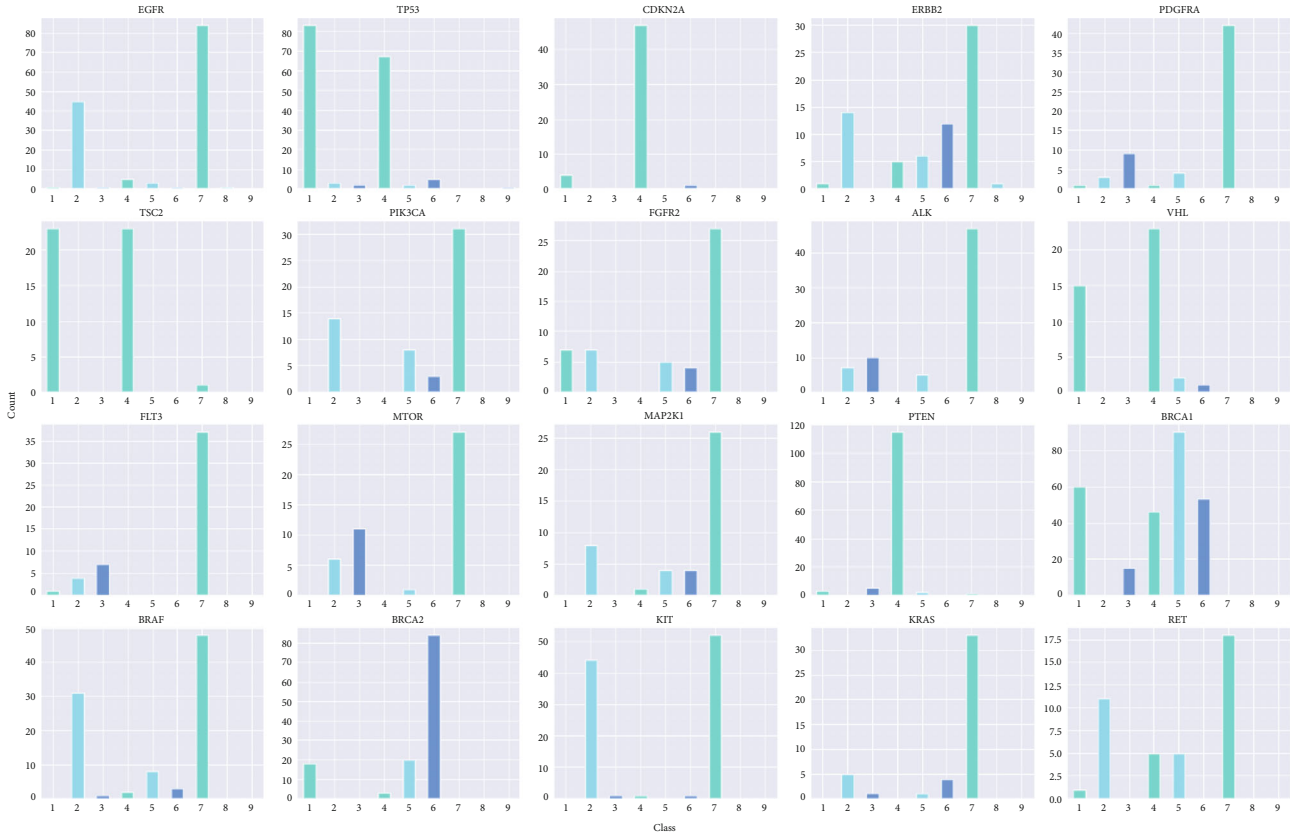

Figure 5: Distribution of genes among classes.

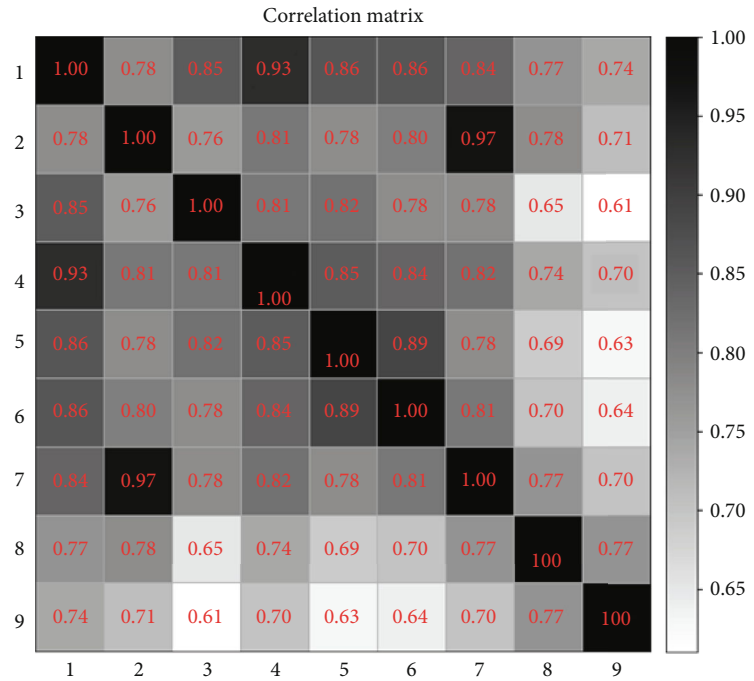

Figure 6: Confusion matrix analysis of the similarity of the texts in different classes.

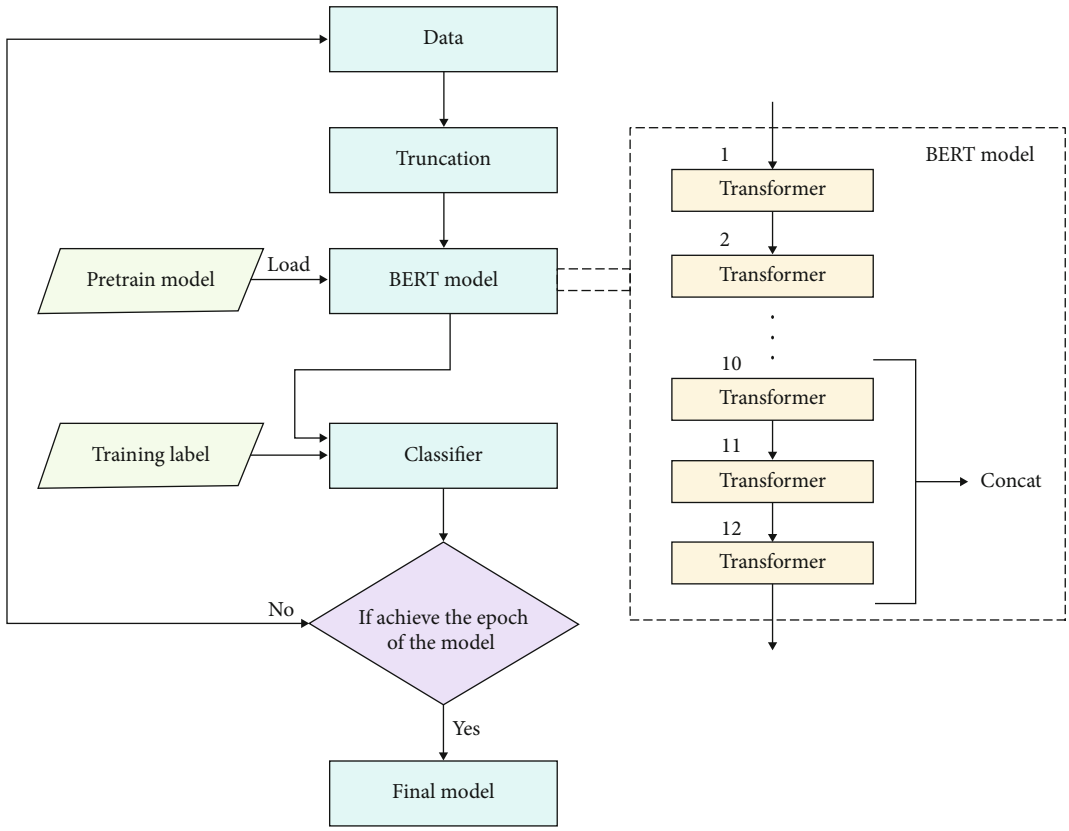

Figure 7: Scheme of the training.

REC calculates the proportion of genes identified correctly belonging to this type of mutation in all this type of gene:

$$\mathrm{REC} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}. \tag{4}$$

F1 score takes into account the factors of PRE and REC. F1 is the standard metric for this task. It combines precision and recall. Macro-F1 is a parameter index that can best reflect the effectiveness and stability of the model:

$$\mathrm{F1} = \frac{2\mathrm{PRE} * \mathrm{REC}}{\mathrm{PRE} + \mathrm{REC}}. \tag{5}$$

The ROC curve is created by plotting the TP against the FP at various threshold settings.

The confusion matrix is a specific table capable of visualizing the performance of an algorithm. Individual rows of the matrix represent the predicted gene classses, while each column represents the genes in the actual classes.

4. Experiments

For easier comparison with other methods, our training process uses the GPU of the server in the lab for training. There are 3136 training sets and 553 verification sets in total. The Python language is selected as the programming language in this experiment. The experiment is completed on Tensorflow's open-source framework and BERT-base. We use the parameters on BERT-base trained by Google through a large number of corpus on Wikipedia as pretraining parameters to accelerate the convergence speed and reduce the convergence difficulty. Our experimental parameters are batch size 128, learning rate $3e-5$, and warmup period 0.06; the whole experiment runs for 30 cycles; the maximum sequence length of BERT input is 512; and the optimizer is Adam optimizer, while other model parameters remain unchanged.

4.1. Experiment Procedure. The BERT model can automatically complete the process of converting each word in the text into a one-dimensional vector by querying the word vector table and inputting it in the model. The input of the model contains three sections: the token embeddings, the segmentation embeddings, and position embeddings.

Because BERT is a pretraining model with high generalization ability, the output layer of BERT can be externally connected with corresponding layers to complete downstream tasks. For example, in this experiment, the processed data is substituted into the BERT model for training, and the output layer will connect Softmax function for classification tasks (Figure 7).

BERT is an unsupervised model that uses whether the sentences are related to each other as labels and masks some words to make the masked words as labels, thus avoiding the tedious process of manually labeling data. Generally, the data in the dataset are not balanced. Take the samples in the 7th and 8th categories of the dataset as an example. The

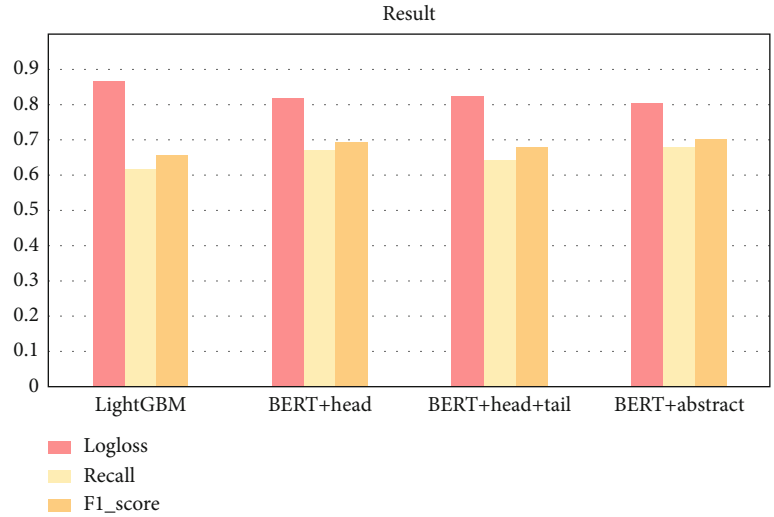

Figure 8: Evaluation of four methods.

difference between their numbers even reaches more than 10 times. In this case, the default classification method makes classifiers pay too much attention to the category with a larger number of samples, thus making the generalization ability of the model weak and unable to obtain satisfactory results. Therefore, we use random sampling to eliminate the imbalance between data and extract only a part of samples from the category within a larger number of samples to balance the sample number differences between classes.

Simultaneously, because the length of the gene text in the dataset is greater than 512 tokens, which is the longest length that can be retained by BERT, we need to use the truncation method to intercept part of the information in the text. We take three ways to solve this problem. The head only truncation method intercepts the first 512 tokens (at most) as input, the head+tail method intercepts part of the head and part of the tail to form 512 tokens (at most) as input, and the abstract +head method sorts the gene text according to importance, then select the most important 512 tokens (at most) as input.

Finally, the processed data are substituted into the BERT model for training. Numerous previous works have shown that fine-tuning a pretrained model which has been trained with a large amount of corpus can significantly improve the classification result. As BERT can learn different contents in different layers, stitching some of the layers together can make the model get richer information, thereby improving the accuracy of the model, so the last three layers in the BERT model are concatenated. Max pooling, fully connected, and Softmax function are added after the concatenated output layer to realize the classification of gene text to improve the classification accuracy of the model.

4.2. Experiment Results and Discussions. It can be seen from the figure that compared with the LightGBM method, the BERT methods using three types of truncation have higher ACC, REC, and F1 score. The confusion matrix shows our classification situation in a visual way (Figure 6). The red numbers are nonzero values. It can be observed that type 1 is easy to be confused with types 4 and 5. There are more

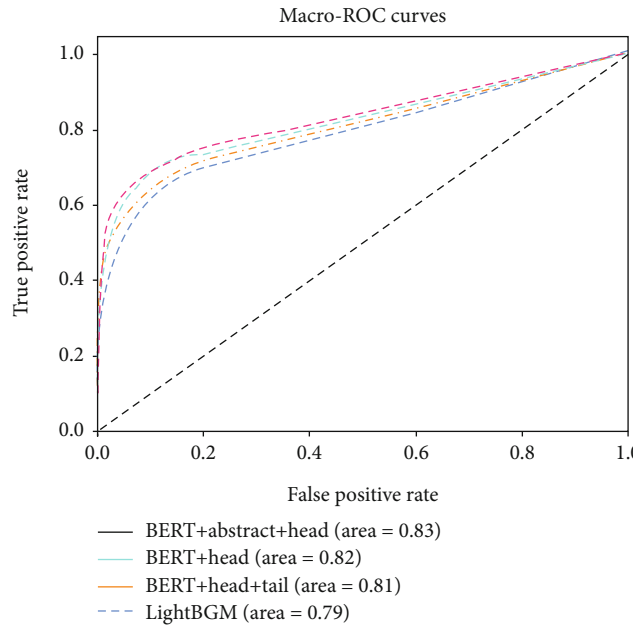

Figure 9: ROC curves of the proposed methods.

machine judgment errors of texts between type 7 and type 2. Overall, the classification of data-lacking types 8 and 9 is more complicated than other types, possibly because there are fewer samples of types 8 and 9, and these two types have fewer intersections with other types of mutation. The lack of intersection leads to difficulties in distinguishing types 8 and 9 from different types of mutations. The ROC curve can evaluate the accuracy of the model prediction.

The performance and ranking of the entries for the proposed four methods are shown in Figure 8. All methods share the same setting of hyperparameters for an unbiased comparison. Overall, deep learning-based algorithms (BERT) perform slightly better than machine learning-based methods (LightGBM). Among the three models using BERT, the BERT+abstract truncation method has the best performance

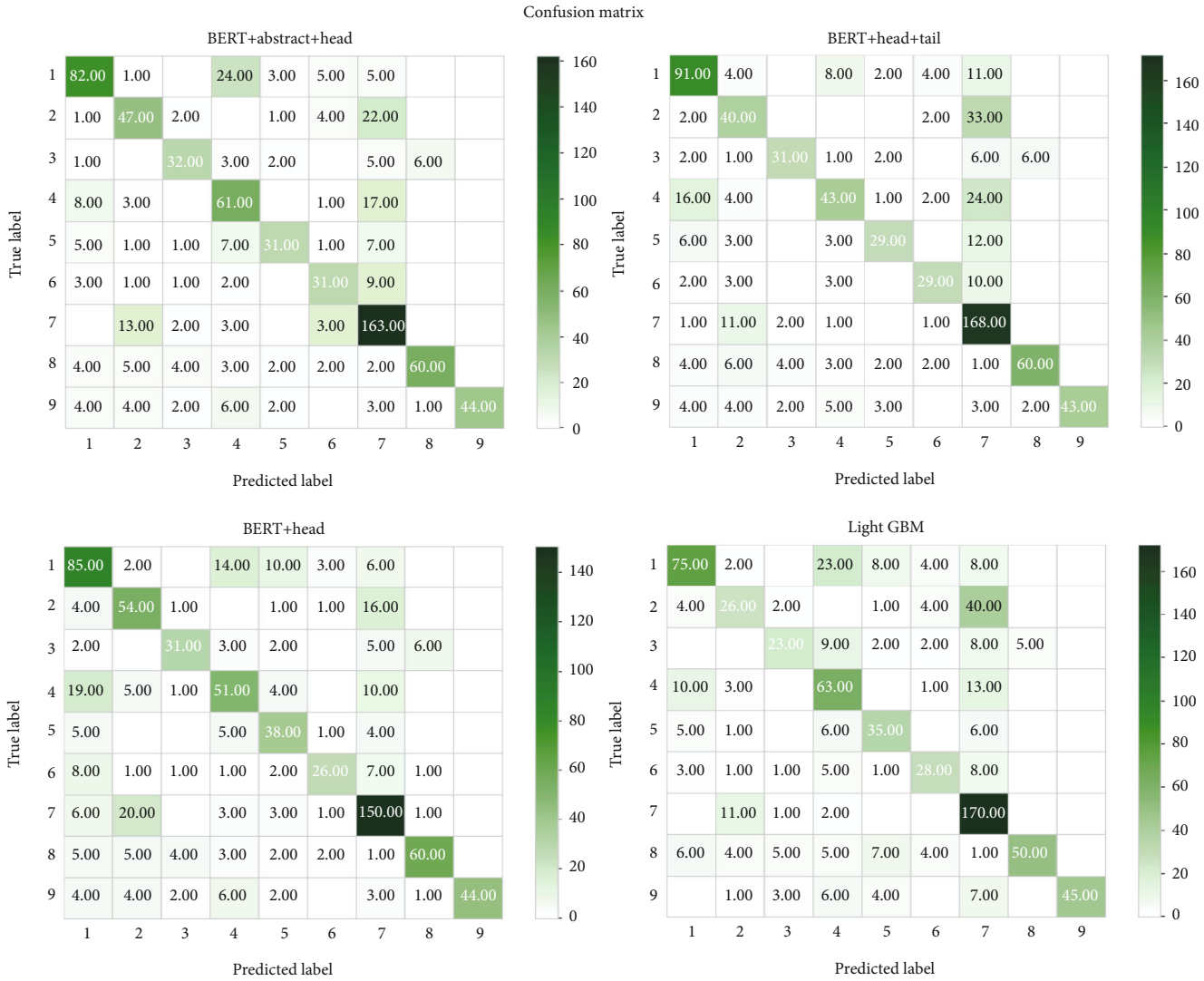

Figure 10: Confusion matrix tables of proposed four methods.

as a single model with 0.8074 logarithmic loss and 0.6837 recall. The 0.705 *F*-measure score is limited by the extreme shortage of training data. Better performance should be obtained when it is applied to large-scale datasets.

Besides, the ROC curves of the other three methods are below the ROC curve of the BERT+abstract (Figure 9). The ROC curves for the BERT+abstract, the BERT+head, the BERT+head+tail, and the LightGBM with the highest and lowest AUCs are also shown in Figure 9. Compared results indicate better performance of the BERT+abstract since the AUC assesses the algorithm's inherent validity using an effective and combined measure of sensitivity and specificity. The accuracy of predicted results is highly dependent on the datasets. The performances of our model are limited by the size of available datasets in our case. However, the capabilities of deep networks can be improved using expanded data. Our proposed model is a proof-of-concept, and we believe it is applicable when applied on large-scale datasets.

Moreover, we compare confusion matrix tables using predict classes versus true classes among different methods (Figure 10). The confusion matrix table is an error matrix which can be used to evaluate the performance of the algorithm. In summary, individual classes of genes are predicted precisely using the BERT+abstract method, corresponding with results of Logloss and F1 measurements.

It can be observed that class 1 is easy to be confused with classes 4 and 5. Furthermore, there are more machine judgment errors of texts between type 7 and type 2. These phenomena can be easily attributed to the similarity of texts among these classes as we previously described. Also, it is apparent that classifying classes 8 and 9 is complicated. The computer may misjudge mutation texts with real labels of 8 or 9 as other types but hardly underestimate other types of mutation texts as type 8 or 9 since there are fewer samples in classes 8 and 9. The shortage of samples in classes 8 and 9 also fails to provide sufficient data to distinguish themselves from other classes since there are no intersections. Contrastingly, the classification of class 7 is easier due to the abundant samples. Therefore, the abundance of data plays essential roles in improving the efficiency of classification.

5. Conclusion

In this study, we propose a deep learning algorithm to identify genomic information within texture-based literature abstracts. Aiming to address the classification problem in an extremely long, imbalanced, and repetitive dataset, we test four methods, including LightGBM and three different truncation BERT methods. By analyzing their Logloss, recall, F1 score, ROC curve, and AUC scores, we notice that the abstract+head truncation BERT method has superior results than other algorithms in all indicators.

In this study, our BERT method is limited due to the shortage of datasets, and its performance can be improved dramatically with the size of datasets increasing. Moreover, our approach will be potentially applied on diagnosing and treating more than 120,000 patients every year around the world based on the announcement of the MSKCC, which will provide our opportunity to enhance our methods further when large-scale datasets are available. We believe BERT is a promising tool for accelerating tumor genomic-related research and facilitating tumor diagnosis and treatments. Besides, this text-based classifier algorithm demonstrated high universality, and it is applicable not only in tumor-specific research but also in other types of diseases and in other nonacademic areas.

Data Availability

All data generated or analyzed during this study are included in this published article.

Conflicts of Interest

The authors declare that they have no competing interests.

Authors' Contributions

N Zhao and ZG Yang conceived and designed the experiments. YH Su, HX Xiang, HT Xie, and Y Yu analyzed and extracted data. YH Su and HT Xie constructed the figures. YH Su, HT Xie, and ZG Yang participated in table construction. All authors participated in the writing, reading, and revising of the manuscript and approved the final version of the manuscript.

Acknowledgments

This work has been supported by the National Key Research and Development Program No.2018YFB2100100, Data-Driven Software Engineering innovation team of Yunnan province of China No.2017HC012, Postdoctoral Science Foundation of China No.2020M673312, Innovation and Entrepreneurship training projects for College Students of Yunnan University No.20201067307, Postdoctoral Science Foundation of Yunnan Province, Project of the Yunnan Provincial Department of Education scientific research fund No. 2019J0010, and DongLu Young and Middle-aged backbone Teachers Project of Yunnan University.

References

[1] M. L. Metzker, "Sequencing technologies — the next generation," *Nature Reviews Genetics*, vol. 11, no. 1, pp. 31–46, 2010.

[2] Z. Yang, J. Shi, J. Xie et al., "Large-scale generation of functional mRNA-encapsulating exosomes via cellular nanoporation," *Nature Biomedical Engineering*, vol. 4, no. 1, pp. 69–83, 2020.

[3] M. Masseroli, A. Canakoglu, P. Pinoli et al., "Processing of big heterogeneous genomic datasets for tertiary analysis of next generation sequencing data," *Bioinformatics*, vol. 35, no. 5, pp. 729–736, 2019.

[4] M. R. Stratton, P. J. Campbell, and P. A. Futreal, "The cancer genome," *Nature*, vol. 458, no. 7239, pp. 719–724, 2009.

[5] P. J. Stephens, The Oslo Breast Cancer Consortium (OSBREAC), P. S. Tarpey et al., "The landscape of cancer genes and mutational processes in breast cancer," *Nature*, vol. 486, no. 7403, pp. 400–404, 2012.

[6] C. Ma, F. Jiang, Y. Ma, J. Wang, H. Li, and J. Zhang, "Isolation and detection technologies of extracellular vesicles and application on cancer diagnostic," *Dose-response : a publication of International Hormesis Society*, vol. 17, no. 4, pp. 155932581989100–1559325819891004, 2019.

[7] N. Walters, L. T. H. Nguyen, J. Zhang, A. Shankaran, and E. Reátegui, "Extracellular vesicles as mediators ofin vitroneutrophil swarming on a large-scale microparticle array," *Lab on a Chip*, vol. 19, no. 17, pp. 2874–2884, 2019.

[8] M. H. Bailey, C. Tokheim, E. Porta-Pardo et al., "Comprehensive characterization of cancer driver genes and mutations," *Cell*, vol. 173, pp. 371–385.e18, 2018.

[9] J. R. Pon and M. A. Marra, "Driver and passenger mutations in cancer," *Annual Review of Pathology: Mechanisms of Disease*, vol. 10, no. 1, pp. 25–50, 2015.

[10] A. Youn and R. Simon, "Identifying cancer driver genes in tumor genome sequencing studies," *Bioinformatics*, vol. 27, no. 2, pp. 175–181, 2011.

[11] D. Tamborero, A. Gonzalez-Perez, C. Perez-Llamas et al., "Comprehensive identification of mutational cancer driver genes across 12 tumor types," *Scientific Reports*, vol. 3, no. 1, p. 2650, 2013.

[12] B. Vogelstein, N. Papadopoulos, V. E. Velculescu, S. Zhou, L. A. Diaz, and K. W. Kinzler, "Cancer genome landscapes," *Science*, vol. 339, no. 6127, pp. 1546–1558, 2013.

[13] Y. Liu, Y. Ma, J. Zhang, Y. Yuan, and J. Wang, "Exosomes: a novel therapeutic agent for cartilage and bone tissue regeneration," *Dose Response*, vol. 17, no. 4, article 1559325819892702, 2019.

[14] C.-F. Tsai and J.-W. Wu, "Using neural network ensembles for bankruptcy prediction and credit scoring," *Expert Systems with Applications*, vol. 34, no. 4, pp. 2639–2649, 2008.

[15] B. Norgeot, B. S. Glicksberg, and A. J. Butte, "A call for deep-learning healthcare," *Nature Medicine*, vol. 25, no. 1, pp. 14-15, 2019.

[16] Z.-L. Sun, T.-M. Choi, K.-F. Au, and Y. Yu, "Sales forecasting using extreme learning machine with applications in fashion retailing," *Decision Support Systems*, vol. 46, no. 1, pp. 411–419, 2008.

[17] C. Shen, D. Nguyen, Z. Zhou, S. B. Jiang, B. Dong, and X. Jia, "An introduction to deep learning in medical physics: advantages, potential, and challenges," *Physics in Medicine & Biology*, vol. 65, no. 5, p. 05TR01, 2020.

[18] R. Wang, Y. Weng, Z. Zhou, L. Chen, H. Hao, and J. Wang, "Multi-objective ensemble deep learning using electronic health records to predict outcomes after lung cancer radiotherapy," *Physics in Medicine & Biology*, vol. 64, no. 24, p. 245005, 2019.

[19] H. Wu, G. Toti, K. I. Morley et al., "SemEHR: a general-purpose semantic search system to surface semantic data from clinical notes for tailored care, trial recruitment, and clinical research," *Journal of the American Medical Informatics Association*, vol. 25, no. 5, pp. 530–537, 2018.

[20] C. Gulden, M. Kirchner, C. Schüttler et al., "Extractive summarization of clinical trial descriptions," *International Journal of Medical Informatics*, vol. 129, pp. 114–121, 2019.

[21] S. Li, P. Xu, B. Li et al., "Predicting lung nodule malignancies by combining deep convolutional neural network and handcrafted features," *Physics in Medicine & Biology*, vol. 64, no. 17, p. 175012, 2019.

[22] D. R. Swanson, "Fish oil, Raynaud's syndrome, and undiscovered public knowledge," *Perspectives in Biology and Medicine*, vol. 30, no. 1, pp. 7–18, 1986.

[23] E. M. Marcotte, I. Xenarios, and D. Eisenberg, "Mining literature for protein-protein interactions," *Bioinformatics*, vol. 17, no. 4, pp. 359–363, 2001.

[24] M.-W. C. Jacob Devlin, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," 2019, https://arxiv.org/abs/1810.04805.

[25] D. Wang, Y. Zhang, and Y. Zhao, "Association for Computing Machinery," in *Proceedings of the 2017 International Conference on Computational Biology and Bioinformatics*, pp. 7–11, Newark, NJ, USA, 2017.

[26] Y. Oytam, F. Sobhanmanesh, K. Duesing, J. C. Bowden, M. Osmond-McLeod, and J. Ross, "Risk-conscious correction of batch effects: maximising information extraction from high-throughput genomic datasets," *BMC Bioinformatics*, vol. 17, no. 1, p. 332, 2016.

[27] A. E. Woerner, M. P. Cox, and M. F. Hammer, "Recombination-filtered genomic datasets by information maximization," *Bioinformatics*, vol. 23, no. 14, pp. 1851–1853, 2007.

[28] S. Raschka and V. Mirjalili, *Python machine learning: machine learning and deep learning with Python, scikit-learn, and TensorFlow 2*, Packt Publishing Ltd, 2019.

[29] L. Li, H. Ruan, C. Liu et al., "Machine-learning reprogrammable metasurface imager," *Nature Communications*, vol. 10, pp. 1–8, 2019.

[30] X.-D. Zhang, *A Matrix Algebra Approach to Artificial Intelligence*, Springer, 2020.

[31] A. Rajkomar, J. Dean, and I. Kohane, "Machine learning in medicine," *New England Journal of Medicine*, vol. 380, no. 14, pp. 1347–1358, 2019.

[32] A. Messac and X. Chen, "EMBC 2019 Speakers," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 2182–2185, Berlin, Germany, 2019.

[33] Z. Tian, A. Yen, Z. Zhou, C. Shen, K. Albuquerque, and B. Hrycushko, "A machine-learning–based prediction model of fistula formation after interstitial brachytherapy for locally advanced gynecological malignancies," *Brachytherapy*, vol. 18, no. 4, pp. 530–538, 2019.

[34] M. Al-Ayyoub, A. Nuseir, K. Alsmearat, Y. Jararweh, and B. Gupta, "Deep learning for Arabic NLP: a survey," *Journal of Computational Science*, vol. 26, pp. 522–531, 2018.

[35] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1746–1751, Doha, Qatar, 2014.

[36] J. Y. Lee and F. Dernoncourt, "Sequential short-text classification with recurrent and convolutional neural networks," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 515–520, San Diego, California, 2016.

[37] C. M. Shiou Tian Hsu, P. Jones, and N. Samatova, "A hybrid CNN-RNN alignment model for phrase-aware sentence classification," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 443–449, Valencia, Spain, 2017.

[38] P. D. Stenson, M. Mort, E. V. Ball et al., "The human gene mutation database: 2008 update," *Genome Medicine*, vol. 1, pp. 1–6, 2009.

[39] P. D. Stenson, E. V. Ball, M. Mort et al., "Human gene mutation database (HGMD®): 2003 update," *Human Mutation*, vol. 21, no. 6, pp. 577–581, 2003.

[40] P. D. Stenson, M. Mort, E. V. Ball, K. Shaw, A. D. Phillips, and D. N. Cooper, "The Human Gene Mutation Database: building a comprehensive mutation repository for clinical and molecular genetics, diagnostic testing and personalized genomic medicine," *Human Genetics*, vol. 133, no. 1, pp. 1–9, 2014.

[41] G. J. G. Prelich, "Gene overexpression: uses, mechanisms, and interpretation," *Genetics*, vol. 190, pp. 841–854, 2012.

[42] A. Gertych, Z. Swiderska-Chadaj, Z. Ma et al., "Convolutional neural networks can accurately distinguish four histologic growth patterns of lung adenocarcinoma in digital slides," *Scientific Reports*, vol. 9, article 1483, 2019.

[43] (MSKCC), "M. S. K. C. C," http://www.mskcc.org/.

[44] N. Indurkhya and F. J. Damerau, *Handbook of Natural Language Processing*, CRC Press, 2010.

[45] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*, O'Reilly Media, Inc., 2009.

[46] C. Manning and H. Schutze, *Foundations of Statistical Natural Language Processing*, MIT press, 1999.

[47] A. Kao and S. R. Poteet, *Natural Language Processing and Text Mining*, Springer Science & Business Media, 2007.

[48] A. Moschitti and R. Basili, *European Conference on Information Retrieval*, Springer, 2007.

[49] J. Hartmann, J. Huppertz, C. Schamp, and M. Heitmann, "Comparing Automated Text Classification Methods," *International Journal of Research in Marketing*, vol. 36, no. 1, pp. 20–38, 2019.

[50] X. Deng, Y. Li, J. Weng, and J. Zhang, "Feature selection for text classification: a review," *Multimedia Tools and Applications*, vol. 78, no. 3, pp. 3797–3816, 2019.

[51] H. Amazal, M. Ramdani, and M. Kissi, *International Conference on Smart Applications and Data Analysis*, Springer, 2020.

[52] W. Chen, X. Liu, D. Guo, and M. Lu, *International Conference on Data Mining and Big Data*, Springer, 2018.

[53] O. Einea, A. Elnagar, and R. Al Debsi, "Sanad: single-label Arabic news articles dataset for automatic text categorization," *Data in Brief*, vol. 25, article 104076, 2019.

[54] T. Joachims, *European Conference on Machine Learning*, Springer, 2005.

[55] J. Chen, H. Huang, S. Tian, and Y. Qu, "Feature selection for text classification with naïve Bayes," *Expert Systems with Applications*, vol. 36, no. 3, pp. 5432–5435, 2009.

[56] N. Suguna and K. Thanushkodi, "An improved k-nearest neighbor classification using genetic algorithm," *International Journal of Computer Science Issues*, vol. 7, pp. 18–21, 2010.

[57] T. Pranckevičius and V. Marcinkevičius, "Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification," *Baltic Journal of Modern Computing*, vol. 5, p. 221, 2017.

[58] S. Bloehdorn and A. Hotho, *International Workshop on Knowledge Discovery on the Web*, Springer, 2004.

[59] X. S. Zhang, D. Chen, Y. Zhu et al., "A multi-view ensemble classification model for clinically actionable genetic mutations," 2018, https://arxiv.org/abs/1806.09737.

[60] E. Al Daoud, "Comparison between XGBoost, LightGBM and CatBoost using a home credit dataset," *International Journal of Computer and Information Engineering*, vol. 13, pp. 6–10, 2019.

[61] B. A. McKinney, D. M. Reif, M. D. Ritchie, and J. H. Moore, "Machine learning for detecting gene-gene interactions," *Applied Bioinformatics*, vol. 5, pp. 77–88, 2006.

[62] T. J. Cleophas, A. H. Zwinderman, and H. I. Cleophas-Allers, *Machine learning in medicine*, Springer, 2013.

[63] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "Statistical and machine learning forecasting methods: concerns and ways forward," *PLoS One*, vol. 13, article e0194889, 2018.

[64] J. Schmidhuber, "Deep learning in neural networks: an overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[65] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–21, 2020.

[66] X. Wu, S. Lv, L. Zang, J. Han, and S. Hu, *International Conference on Computational Science*, pp. 84–95, Springer.

[67] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune BERT for text classification?," in *China National Conference on Chinese Computational Linguistics*, Springer, 2019.

[68] G. Giannakopoulos, P. Mavridi, G. Paliouras, G. Papadakis, and K. Tserpes, "Article 13 (Association for Computing Machinery)," in *Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics*, Craiova, Romania, 2012.