Construction of Quantitative Assessment System and Optimization of Personalized Planning Strategy for College Students' Competition Development Based on Transformer Algorithm

Zeseng Mao[a], Weiguo Li[b], Chengying Fang[c], Liang Chen[d]*

Guangdong University of Science and Technology, Dongguan, 523000, China

[a]maom2485835011@qq.com, [b]liweiguo87@gdust.edu.cn,
[c]3200515606@qq.com, [d]*230236879@qq.com

Abstract. The purpose of this paper is to construct a set of quantitative assessment system for competition development of college students based on Transformer algorithm. It first analyzes the importance of college students' participation in competitions and the current status of the application of Transformer algorithm in the field of education, and summarizes the research progress of related work. Subsequently, the data collection and processing methods are elaborated in detail, including data sources, preprocessing techniques and so on. In the model design and implementation section, the underlying principles of the Transformer architecture and the specific adjustments for competition assessment are introduced. Through experimental design and result analysis, the performance of the proposed system is verified and the comparative advantages with existing solutions are demonstrated. Finally, the main findings are summarized, application prospects are predicted, and future research directions are suggested.

Keywords: transformer algorithm; university student competition; quantitative evaluation system; data collection and processing; model design; experimental design; performance evaluation

1 Introduction

1.1 Background and Significance

Analysis of the importance of students' participation in competitions in higher education, Current status of the application of Transformer algorithm in the field of education, Overview of the demand for quantitative evaluation of competition results

In the current educational context, college students' participation in competitions has become an important way to cultivate their innovative ability, teamwork spirit and practical ability. In order to better assess students' performance in competitions and provide personalized planning strategies, this study proposes a quantitative assessment

© The Author(s) 2025
B. Siuta-Tokarska et al. (eds.), *Proceedings of the 2025 3rd International Academic Conference on Management Innovation and Economic Development (MIED 2025)*, Advances in Economics, Business and Management Research 348,
https://doi.org/10.2991/978-94-6463-835-6_97

system for competition development of college students based on Transformer algorithm. By analyzing the current status of the application of Transformer algorithm in the field of education, we find that its powerful natural language processing capability and deep understanding of sequence data make it an ideal choice for constructing this kind of assessment system. Meanwhile, with the increasing richness and diversity of competition activities, how to quantitatively assess students' competition results scientifically and objectively has become an urgent problem in the field of education[1]. This study aims to achieve accurate assessment of students' competition performance through the Transformer algorithm, which in turn provides strong support for students' personalized development planning[2].

1.2 Overview of Related Work

Progress of domestic and international research on students' performance in competitions, current problems and challenges, trends in the application of Transformer model to non-verbal tasks.

In constructing a quantitative assessment system for college students' competition development based on the Transformer algorithm, we first conducted an in-depth review of domestic and international research on students' competition performance. These research advances reveal the current problems and challenges, especially the shortcomings in quantitatively assessing students' competition performance. With the wide application of the Transformer model in non-verbal tasks, we see the potential of applying it to the assessment of student competitions[3].The advantages of the Transformer model in processing sequence data enable it to capture complex patterns and associations in competition data, offering the possibility of constructing a more accurate and comprehensive assessment system[4].

2 Data Collection and Processing Methods

2.1 Description of Data Sources

Collection of students' personal information, acquisition of competition history, and integration of data on other influencing factors (e.g., instructor's experience).

In the research process of constructing a quantitative assessment system for college students' competition development based on the Transformer algorithm and optimizing personalized planning strategies, the data collection and processing methods are particularly critical. First, we need to clarify the source of data, which mainly includes the collection of students' personal information, such as grade, specialty, and performance, in order to comprehensively depict the basic characteristics of students[5]. Secondly, the acquisition of competition history records is also an indispensable part, which covers the types, levels, and awards of students' participation in competitions, providing a strong basis for assessing students' competition ability. In addition, data integration of other influencing factors is also crucial, such as the experience of instructors and the school's resource allocation, all of which may have a profound impact on students' competition development. By integrating these diverse data sources, we are able to

construct a more comprehensive and accurate assessment system, thus providing students with more personalized planning strategies to optimize their programs[6].

2.2 Pretreatment Technology

Missing value handling strategies, Feature engineering: extracting useful information from raw data, Data normalization/normalization process

Preprocessing techniques are a crucial part of the data collection and processing process. For the processing of missing values, we adopt various strategies, including mean padding, median padding, and predictive padding based on machine learning algorithms, to ensure the completeness and accuracy of the data[7]. At the same time, we conducted in-depth feature engineering to carefully extract valuable information from the raw data to provide a solid foundation for subsequent model training. In addition, in order to eliminate the magnitude differences between data features, we perform data standardization and normalization, so that all data features are at the same magnitude, thus improving the model training efficiency and prediction accuracy.As shown in Table 1, the above description is verified[8].

Table 1. Classification of data preprocessing techniques

form	subcategory	Strategy/Technology	explanation
Missing value handling	deletion	Full case deletion	Delete observations containing missing values (rows)
		Available cases deleted	Remove observations containing missing values only if there is enough information to make a prediction or classification
		interpolation	Fill in missing values using mean, median, plurality, etc.
		regression interpolation	Predicting values of missing variables based on values of other variables
		Multiple interpolation	Creating multiple complete datasets by interpolating multiple times to reflect the uncertainty of missing values
feature engineering	feature selection	filtration	Selection of features based on statistical tests such as correlation, chi-square tests, etc.
		parceling	Using model performance as an evaluation criterion for feature selection, e.g., recursive feature elimination
		embedding method	Feature selection as part of the model training process, e.g. L1 regularization

	Feature Construction	polynomial characterization	Create new features, such as products or combinations of original features
		Interactive features	Create features that reflect the relationship between two or more original features
		text feature	Extract meaningful features from text data, such as bag-of-words model, TF-IDF, etc.
Data standardization/normalization	standardization	Z-score standardization	Transform the data to a distribution with mean 0 and standard deviation 1
		Min-Max standardization	Scale data to a specified range (usually 0 to 1)
		RobustScaler	Standardized using median and interquartile range (IQR), insensitive to outliers
	normalize	L1 normalization	Each eigenvalue of the data divided by the L1-parameter of that eigenvalue
		L2 normalization	Each eigenvalue of the data divided by the L2 norm of that eigenvalue
		MaxAbsScaler	Scale the data to the range [-1, 1], keeping the sign of each feature constant

3 Model Design and Implementation

3.1 Model Design and Implementation

Introduction to the basic principle, Self-attention mechanism, Encoder-Decoder structure analysis

The Transformer architecture, as an important breakthrough in the field of natural language processing in recent years, is based on the principle of encoding and decoding sequence data through the self-attention mechanism. The self-attention mechanism allows the model to process each word while referring to other words in the sequence, thus capturing global dependencies[9]. The encoder-decoder structure is the core component of Transformer. The encoder is responsible for converting the input sequence into a series of high-dimensional vectors, while the decoder generates the output sequence based on these vectors. This structure allows Transformer to excel in handling sequence-to-sequence tasks such as machine translation and text summarization. As shown in Table 2, the above description is verified.

Table 2. Analysis of the principles and application scenarios of the core components of the Transformer model

form	breakdown	elaborate on	Introduction to the Principle	Self-attention mechanism	Encoder-Decoder Structure
basic principle	model essence	Deep Neural Networks Based on Self-Attention Mechanism	Transformer is a neural network architecture based on a self-attention mechanism, which learns a representation of the input data through multiple layers of self-attention mechanisms and feed-forward neural networks.	✓	✓
	core idea	Self-attentive alternative recurrent neural networks	Unlike traditional Recurrent Neural Networks (RNNs) or Convolutional Neural Networks (CNNs), Transformer relies exclusively on the self-attention mechanism to capture dependencies in the input sequence.	✓	
self-attention	Principles of the mechanism	By calculating word-to-word relationship scores	The self-attention mechanism captures the dependencies in the sequence by calculating the relationship score of each position in the input sequence with all other positions.	✓	
	vantage	Parallel computing, long range dependency capture	The self-attention mechanism allows the model to process input sequences in parallel while being able to capture long-range dependencies, solving the problem of gradient vanishing or gradient explosion in RNNs.	✓	
	drawbacks	Computational complexity, lack of location information	The computational complexity of the self-attention mechanism is high, and the original self-attention mechanism does not take into account the positional information of the input	✓	

			sequence, which needs to be supplemented by positional encoding.		
encoders	Structure Composition	Multi-layer self-attention mechanism + feed-forward neural network	The encoder consists of multiple identical layers stacked on top of each other, each containing a self-attentive mechanism and a feed-forward neural network for learning the representation of the input data.		✓
	functionality	Convert the input sequence to a high-dimensional representation	The encoder converts the input sequence into a high-dimensional vector representation for the decoder by means of a multi-layer self-attention mechanism and a feed-forward neural network.		✓
codec	Structure Composition	Multilayer self-attention mechanism + encoder-decoder attention + feedforward neural network	The decoder is similar to the encoder, but with an additional encoder-decoder attention mechanism to capture the dependencies between the input sequence and the target sequence.		✓
	functionality	Generate target sequence	The decoder generates the complete target sequence step by step based on the encoder output and the generated target sequence.		✓
application scenario	natural language processing (NLP)	Machine translation, text summarization, dialog generation, etc.	Transformer has a wide range of applications in the field of natural language processing, such as machine translation, text summarization, dialog generation, and other tasks.	✓	✓
	computer vision	Image classification, target detection, image generation, etc.	Transformer is also gradually used in the field of computer vision, such as ViT (Vision Transformer) for image classification and other models.	✓	✓

3.2 Specific Adjustments for Competition Assessment

Definition of input and output formats, basis of loss function selection, sharing of training optimization techniques

In the model design and implementation phase, we fine-tuned the quantitative assessment system for the development of university student competitions. First, we specified the input and output formats of the model. The input data include students' competition participation, awards and other relevant features, while the output is the quantitative assessment results of students' competition development level. In order to ensure the effectiveness of model training, we chose an appropriate loss function, and this choice was mainly based on the characteristics of data distribution and the importance of assessment indicators. During the training process, we used a series of optimization techniques, such as learning rate adjustment strategy, regularization method, and early stopping mechanism, to prevent the model from overfitting and improve the generalization ability. The application of these techniques not only improves the performance of the model, but also makes our evaluation system more accurate and reliable.As shown in Table 3, the above description is verified[10].

Table 3. Evaluation dimensions and technical implementation of the whole process of machine learning model development

Assessment dimensions	breakdown	Technical details	Implementation methodology	Effectiveness evaluation
Input and output format definitions	Data preprocessing	1.Raw data format parsing 2.Data cleaning and transformation 3.Feature engineering	1.Data processing using Pandas library 2.Custom data transformation functions - Feature selection using Scikit-learn	- Improve model training efficiency - Ensure data quality - Improve feature validity
	Data Post-Processing	1.Parsing of prediction results2. Formatting of results3. Storage and output of results	- Customize result parsing functions - Output results in Json or CSV format - Store results to database or cloud storage	- Facilitates analysis and presentation of results - Meets business requirements format - Ensures traceability of results
Loss function selection basis	Type of loss function	1.Mean Square Error (MSE) 2.Cross Entropy Loss (CEL) 3. Customized Loss Functions	- Select the appropriate loss function based on the problem type - Define the loss function in the deep learning framework - Adjust the loss function weights according to the business requirements	- Optimize model training objectives - Improve model prediction accuracy - Meet business-specific requirements

	regularization term	- L1 regularization - L2 regularization - Dropout etc.	- Add regularization terms during model training - Adjust regularization coefficients to balance model complexity and generalization ability	- Prevent model overfitting - Improve model generalization - Optimize model structure
Training Optimization Tips Shared	Batch resizing	- Mini-batch Gradient Descent (Mini-batch GD) - Dynamic batch resizing	- Select appropriate batch size based on memory size and model complexity - Dynamically adjust batch size during training to accelerate training	- Increase training speed - Reduce memory usage - Find the optimal balance between training stability and speed.
	Learning rate adjustment strategies	- Fixed learning rate - Learning rate decay - Adaptive learning rate algorithms (e.g. Adam)	- Initially set the appropriate learning rate - Use a learning rate decay strategy to minimize oscillations in the late stages of training - Try to use an adaptive learning rate algorithm to automatically adjust the learning rate	- Accelerate model convergence - Prevent oscillations and instability during training - Find the best way to set the learning rate
	Model saving and loading	- Save optimal model weights - Load pre-trained model for fine-tuning	- Use the model saving and loading functionality provided by the deep learning framework - Periodically save the best model weights from the training process - Load the pre-trained model for fine-tuning or continued training when needed	- Facilitates model iteration and optimization - Saves training time - Increases model application flexibility

4 Experimental Design and Analysis of Results

4.1 Experimental Setup

Selection of evaluation indicators, introduction of comparative baseline models, configuration of experimental parameters

In the experimental design and result analysis section, we first focus on the experimental setup. To ensure the comprehensiveness and accuracy of the evaluation, we carefully selected a series of evaluation metrics, which not only cover the traditional metrics of model performance, such as accuracy, recall, etc., but also include metrics customized for the specific scenarios of this study to portray the model's performance in the quantitative evaluation of competition development in a more detailed way. Meanwhile, in order to validate the superiority of the model based on the Transformer algorithm, we introduce several comparative baseline models that have demonstrated good performance in previous studies, and thus can provide a strong reference for this study.

In terms of experimental parameter configuration, we adopted a rigorous strategy. By referring to the relevant literature and exploring the pre-experiments, we determined the key parameters of each model, such as the learning rate, the batch size, the number of iterations, etc., in order to ensure the fairness of the experimental conditions and the comparability of the results. In addition, we also strictly control the experimental environment, including the uniformity of hardware equipment and the stability of software environment, so as to minimize the influence of external factors on the experimental results. As shown in Table 4, the above description is verified.

Table 4. Comparison of performance evaluation indicators of machine learning models with baseline models

serial number	Evaluation indicators	Detailed description	Comparison of baseline models	Experimental Parameter Configuration
1	Accuracy	Proportion of correctly categorized samples to total samples	Logistic Regression	Learning rate = 0.01, number of iterations = 1000, regularization strength = 0.001
2	Precision	Proportion of samples predicted to be positive and actually positive as a percentage of samples predicted to be positive	Support Vector Machine	Kernel function = RBF, C=1.0, gamma=0.001
3	Recall	Proportion of samples predicted to be positive and actually positive to those actually positive	Random Forest	Number of trees = 100, Maximum depth = 10, Minimum sample split = 2
4	F1 Score	A reconciled average of precision and recall for comprehensive evaluation of model performance	Gradient Boosting	Learning rate = 0.1, number of iterations = 100, maximum depth = 3, subsample proportion = 0.5
5	ROC Curve	Curves plotted with the false positive rate on the horizontal axis and the true rate on the vertical axis are used to assess	AdaBoost	Learning rate=1.0, number of iterations=50, base classifier=DecisionTreeClassifier

		model performance		
6	AUC Score	Area under the ROC curve for quantitative assessment of model performance	Naive Bayes	Using Gaussian Park Bayes
7	Training Time	Time required for model training	K-Nearest Neighbors	Number of neighbors = 5, Distance measure = Euclidean distance
8	Prediction Time	Time required for the model to make predictions on new data	Decision Tree	Maximum Depth = 5, Minimum Sample Split = 2
9	Model Complexity	Number of parameters or structural complexity of the model	Linear Regression	Use L2 regularization, regularization strength = 0.01
10	Overfitting Degree	The extent to which the model's performance on the training set differs from the performance on the test set	Neural Network (multilayer perceptual machine)	Number of hidden layers = 2, Number of neurons per layer = [64, 32], Activation function = ReLU, Optimizer = Adam

5 Conclusion

Demonstration of key metrics such as accuracy and recall, robustness testing in different scenarios, and comparative advantages with existing solutions.

In the section of experimental design and result analysis, we focused on the performance evaluation of the quantitative assessment system for college students' competition development based on Transformer algorithm. The performance of the model in different scenarios was comprehensively evaluated through the demonstration of key indicators such as accuracy rate and recall rate. The robustness test further verifies the stability and reliability of the model in various practical application environments. Compared with existing solutions, the evaluation system proposed in this study demonstrates significant advantages in terms of data processing efficiency, prediction accuracy, and practicality of personalized planning strategies. These experimental results fully demonstrate the effectiveness and application value of the assessment system based on the Transformer algorithm in the quantitative assessment of college students' competition development and the optimization of personalized planning strategies.

References

1. YU Jiuyang, HU Tianhao, DAI Yaonan, et al. Anchorless frame Transformer algorithm for remote sensing target detection[J]. Electronics Letters,2023,51(11):3238-3247. DOI:10.12263/DZXB.20220612.

2. Zhicheng Zhang. Research and Implementation of Lightweight Visual Transformer Algorithm [D]. Guangdong:Guangdong University of Technology,2024.
3. Qian Wanghan. Research and application of image restoration technology based on Transformer algorithm [D]. Jiangsu:Yangzhou University,2024.
4. YIN Yuchen, LIU Yuhang, MA Wangqian, et al. Integrated energy demand forecasting for parks based on Transformer algorithm[J]. Integrated Intelligent Energy,2023,45(10):61-69. DOI:10.3969/j.issn.2097-0706.2023.10.008.
5. SONG Dongsheng,CHEN Liang. Improved Transformer Algorithm for Semantic Segmentation of Remote Sensing Images[C]// Proceedings of the 17th National Conference on Signal and Intelligent Information Processing and Applications. 2023:387-392,525.
6. Chang Jiaqi, Huang Hongwei, Zhang Dongming, et al. Prediction of normal wear of shield hob based on clustering and Transformer algorithm in upper-soft and lower-hard strata[J]. Construction Technology (in Chinese and English),2024,53(1):107-115. DOI:10.7672/sgjs2024010107.
7. WEN Wei, FU Zhiyuan, ZHANG Yanhui. Research on hybrid option pricing model based on FFT and Transformer algorithm[J]. Journal of Hebei University of Science and Technology,2024,45(5):562-572. DOI:10.7535/hbkd.2024yx05011.
8. Lian Zhirun,Zhang Jiawei,Yang Baolin. Construction of Chinese medicine prescription recommendation model based on rule generation and Transformer algorithm[J]. China Journal of Basic Chinese Medicine,2024,30(3):437-442.
9. Sun Yuanyuan. Prediction of people flow in geographic areas based on transformer algorithm[J]. Digital User,2023(51):59-60.
10. Dandan Huang,Yuchen Zhang,Guangqiu Chen,et al. Lightweight single target tracking algorithm based on Transformer[J]. Journal of Huazhong University of Science and Technology (Natural Science Edition),2025,53(3):41-47,158. DOI:10.13245/j.hust.250458.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.