Hindawi
Scientific Programming
Volume 2022, Article ID 2576535, 9 pages
https://doi.org/10.1155/2022/2576535

Research Article

Design and Application of an AI-Based Text Content Moderation System

Heng Sun [1,2] **and Wan Ni** [1]

[1]*School of Journalism and Communication, Shandong University, Jinan 250100, China*
[2]*School of Foreign Languages and Literature, Shandong University, Jinan 250100, China*

Correspondence should be addressed to Wan Ni; niwan@sdu.edu.cn

Received 21 November 2021; Revised 6 January 2022; Accepted 26 January 2022; Published 21 February 2022

Academic Editor: Ahmed Farouk

Cloud computing, 5G mobile network, and other new technologies have been applied in higher education in recent years. The education video resource service system with adaptive multiterminals has received widespread attention from the field. From these videos, students can get new knowledge and use the system's built-in text comment function to communicate and interact with others. However, due to the fast increase in the number of such text comments, the traditional text content moderation methods such as the keyword method and the regular expression method can no longer meet the growing business needs. Therefore, to solve this matter, this study designed a text content moderation (TCM) system based on artificial intelligence (AI), which uses artificial intelligence and cloud-based algorithm models to analyse and recognize the text comments submitted from the web-end and app-end of the education video resource service system and completes operations such as automatic detection and manual moderation. The proposed TCM system can significantly improve the efficiency of text content moderation.

1. Introduction

With the expansion of the 5G network coverage and the development of Internet digital media technology, high-quality higher education resources based on mobile Internet spring up continuously, bringing fundamental changes to the knowledge acquisition and digestion mode of learners, and using fragmented learning time to acquire fragmented knowledge has become a common learning method in such a mobile Internet environment [1]. To better serve the teaching practice, our research team designed and developed an education video resource service system based on actual teaching requirements, containing modules such as mobile live broadcast, video on demand (VOD), and video clips. Users use these modules to learn, comment on the resources using texts, and interact and communicate with others; they can express their opinions on these videos; however, malicious posts and junk information would appear in these comments and messages from time to time, exerting negative impact on the online learning environment. Thus, how to quickly and effectively filter the illegal and foul text information in the comments is an urgent problem that needs to be resolved. Currently, the illegal text information in the reviews is generally moderated based on keywords and rules. Specifically, a suspected text is added into a database as a keyword, and the moderation system will look for entries in the database that match the keyword. If a matching entry is found, the text will be determined as containing bad information. However, the keyword-based matching ignores the context and thus releases many false alarms, that is, faces a low accuracy. To solve the problem, the rule-based keyword and regular expression moderation mode come into being. The mode can detect deliberate text confusion, as well as context-related comments. However, the moderation rules and the corresponding corpus must be updated constantly. Otherwise, it is impossible for the mode to adapt to the ever-changing illegal texts.

In terms of the design and development of AI-based TCM system, researcher Chen Jing designed a content management system for bullet screen comments of TV live

broadcast, which realized bullet screen comment moderation, edit, and management [2]. The text contents were filtered by comparing the comments and the keywords in the moderation system. But the moderation accuracy depends on whether the keywords cover most illegal texts. Wang et al. [3] proposed a text content safety recognition system based on a recognition model and used deep learning to build moderation model and algorithm; however, each time the algorithm needs to be adjusted or the sample library needs to be updated, it'll take a lot of time and energy. Liu and Huang [4] researched the filtering of illegal and harmful information in the network content and established an information classification system, which can filter information according to the characteristics of each text type; but the classification system, the filter lexicon, and the model features were established based on the researcher's own understanding, so there are a few problems with the system such as the unclear classification, and the system could not update in real time according to the network changes.

In view of the above analysis, this study designed a new TCM system that uses AI, and cloud-end algorithm models to automatically recognize, mark, moderate, and manually recheck the text content of messages and comments submitted from multiple user terminals, and the proposed system could realize intelligent detection and management of text comments.

2. Architecture Design of the AI-Based TCM System

2.1. Technical Architecture. The proposed AI-based TCM system aims at intercepting and filtering the comments, messages, and bullet screen comments that do not meet the moderation standards and rules. The system adopts an intelligent moderation + manual recheck mode to detect the text content [5]. Intelligent moderation is not a complete replacement of manual moderation, it just offers aids and reduces and the workload and intensity of manual operations, for suspected illegal content that cannot be determined by the system, the text will be submitted for manual recheck, and administrators will make judgement on the text. According to the requirement of the moderation task, the system administrator selects the algorithm and text from the existing libraries and configures the moderation by using options and switches in the management control terminal, enabling the automated TCM of the system. When the system encounters a text that cannot be identified or processed, it will push the text to the manual moderation module for manual TCM. At the same time, administrators can conduct spot checks and second moderation on the content that has been automatically moderated; in this way, the moderation accuracy could be improved greatly [6].

The proposed system was developed based on cloud service. Firstly, the cloud server function in cloud service is adopted to complete the basic operation of the TCM system. Secondly, the cloud control platform of cloud service selects the algorithm and text from the existing libraries, configures the moderation, and generates the APIs that can be called by external users. Finally, the APIs are called by the program in the cloud server to detect the text contents and complete the TCM function.

Cloud service can deliver infrastructure, platform, and software resources to users via the network. Users do not have to possess professional knowledge to use the network to access cloud resources through self-service. Our project is based on Tencent Cloud, which provides leading technical products and services to government agencies, corporate organizations, and individual developers, such as cloud computing, big data, and AI. In our project, Tencent Cloud servers are employed to quickly build a system operating environment; Tencent Cloud databases are utilized to realize secure databases with multiple availability zones, preventing malfunctioning of database instances or interruption of availability zones; Tencent Cloud storage is adopted to realize distributed storage of files with high scalability, low cost, good reliability, and strong security; The TI platform of Tencent Cloud is used to achieve one-stop machine learning service platform and other functions. Cloud service providers offer dynamic and scalable resources through the network in a use-on-demand and charge-by-byte mode, and they are also responsible for security management, operation support management, service platform management, resource platform management, etc. [7].

In order to save money, manpower, and material resources, simplify the research and development process, and improve development efficiency, the proposed system was developed under the SaaS cloud service mode. Tencent Cloud service platform provides computing environment services that are directly useable, which meet the complex and diverse calculation needs of our project and satisfy the complex operating environment required by novel computing modes like AI training. The platform offers a dazzling array of computing power combinations, including CPU nodes and GPU nodes. In addition, the software platform adopts a solution different from the job scheduling systems of traditional supercomputing centers. Diverse services can be provided flexibly without sacrificing the computing performance.

On the cloud-end, developers use visual operations to complete the entire development process including data preprocessing, modeling, model training, evaluation, and model release. The project developers can work through visual operations on the cloud end, eliminating the need for laborious pure code operations like building operating environment, model construction, and data processing with command lines. When an algorithm alone cannot complete the development, the algorithm can be replaced swiftly on the cloud console, without suspending the relevant development. During the cloud visual modeling, interactive design can be carried out using the drag-type customizable task flows on the web-end. Both developers and algorithm engineers can quickly build models. The cloud visualization service supports one-stop machine learning. The developers only need to prepare the training data. All the subsequent modeling could be realized on Tencent Cloud service platform. The cloud service platform has many commonly used built-in AI algorithms and models, and developers only need to select the appropriate algorithms and models and

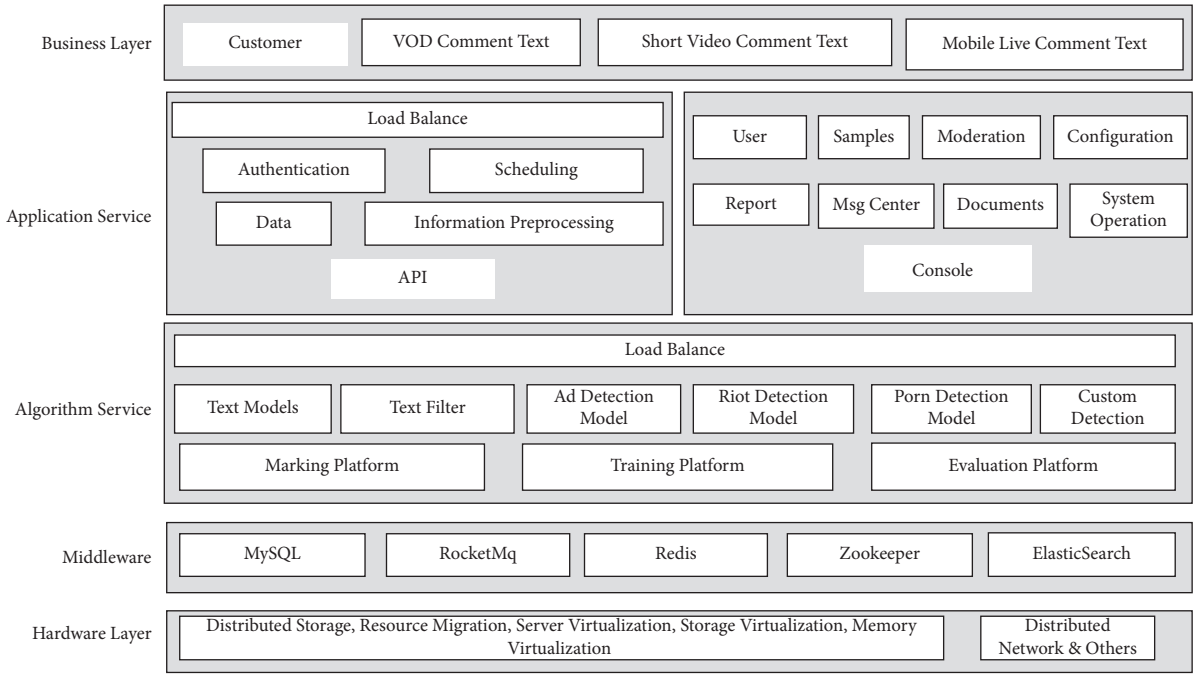

Figure 1: System architecture.

make minor parameter adjustments and local modifications to generate moderation data models according to their actual conditions. The Tencent Cloud's text sample feature library has already accumulated a large number of samples that can cover various Internet social contact scenarios. The feature library mainly contains pornographic contents (texts on severe pornography and vulgar culture), terrorism contents (texts on severe terrorist behaviors and articles), advertising contents (texts on advertising behaviors), illegal contents (texts on contents violating laws and regulations), and abusive contents (texts on severe and light abusive words). The content in the text sample feature library is updated at a regular basis, which can effectively reduce the pressure of developers to collect data and build sample library by themselves.

2.2. System Architecture. The proposed system realizes text content processing, analysis, moderate, and other functions by calling the API (Application Programming Interface) [8]. The system was built with modules, the bottom layer uses various functional components provided by AI, and the top layer adopts self-developed business applications [9]. From bottom up, the system is composed of five modules including a hardware layer, a middleware layer, an algorithm service layer, an application service layer, and a business layer [10], Figure 1 gives a diagram of the system architecture (see Figure 1).

Users can use mobile live broadcast module, VOD module, video clips module, and other modules by logging in to the web-end or app-end of the education video resource service system, and then, they can input text content on the relevant comment page or live broadcast interaction page and submit their comments. The submitted text information performs data communication and transmission with the system through mobile Internet, wireless local area network, and wired network, and the business structure of the system is shown in Figure 2 (see Figure 2) [11]. At first, the text information is preprocessed by the front-end platform, and the security firewall installed in the front-end platform can detect network attacks in a timely manner and prevent the system from being threatened by illegal intrusions; then, the front-end preprocessing server judges the format of the text information and checks for illegal characters and input codes; after security check, the text information enters the AI-based TCM system and the system calls out API to perform text analysis, Lexical analysis, syntax analysis, semantic analysis, sentiment analysis, text classification, and other processing operations on the content. The AI computing platform that provides API has multiple functions such as text tags, model training, machine learning, and AI algorithms, and it can fully cover the various processing of the text content [12]. After security check and content check, the system will detect the text for advertisement, banned content, spams, and other features, and then, the automatic processing results will be pushed to the system administrators who need to confirm or correct the text content processing results. The administrator can adjust the accuracy of intelligence moderation according to the moderation criteria. There are a total of five types of moderation criteria: pornographic contents, terrorism contents, advertising contents, illegal contents, and abusive contents. For example, the common moderation mode could be selected as the moderation criteria for daily use. Then, moderate content moderation will be conducted on the texts that disseminate texts on obscenity, pornography, gambling, violence, murder, and terrorism. For another example, if many terrorism contents pop up in a short term, posing a threat to national

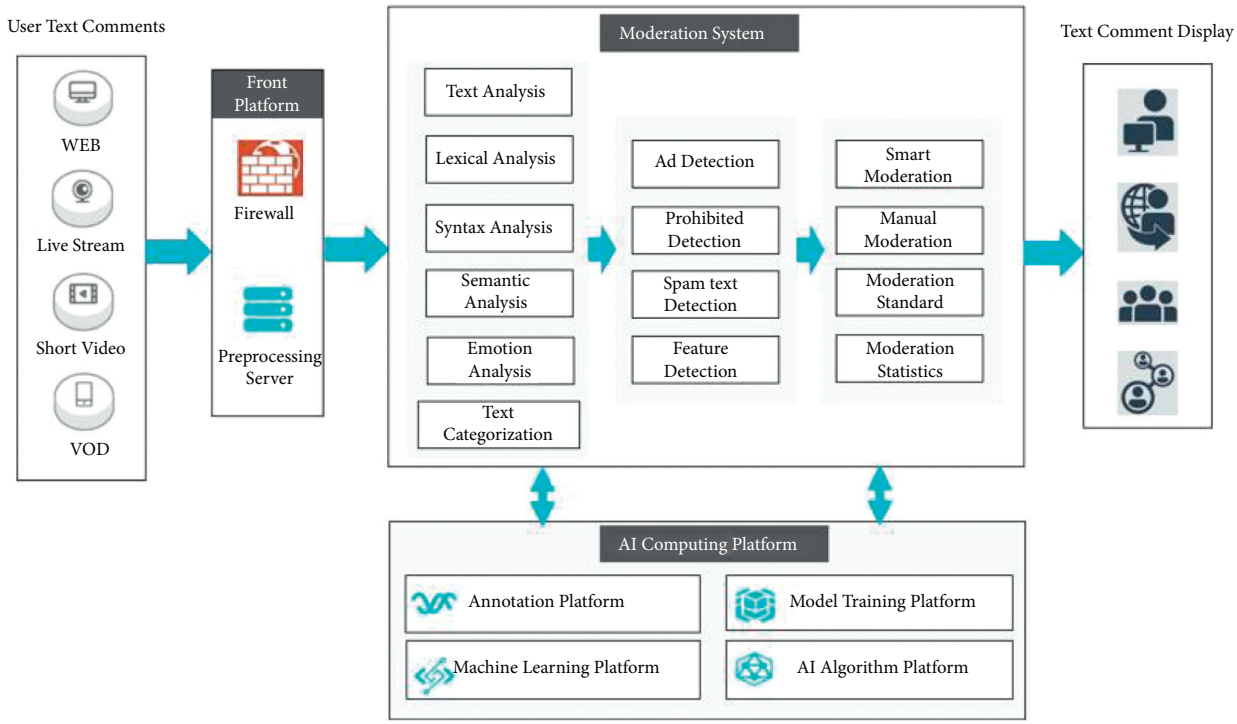

Figure 2: Business structure of the system.

security and race equality, it is necessary to increase the moderation criteria to the strong mode and step up the moderation strength on context, deformed words, and text contents. In this way, the moderation accuracy could be kept within a reasonable range. After that, the system returns the administrators' final moderation results to the AI computing platform and updates text tags and model training, so that the computing platform has the abilities of continuous learning and iterative optimization, later on, when the system comes across the same or similar text content, it can process it according to the moderation method of the administrators. At last, after system moderation, text comments that meet the publishing criteria will be pushed to the web-end and app-end, and displayed in front of all users [13].

2.3. System Operation Process. In the beginning, a user needs to enter the right name and password to log in to the system, and after identity verification is passed, the user can use the web-end or app-end to publish text comments on the mobile live broadcast programs or video clips. The system first checks the format of the input text, and if error has been detected, the system will not process the text anymore, but will prompt that the input format is incorrect, and the user needs to re-enter the text; after format check is passed, the text content will be pushed to the AI-based TCM system, which will call the API to moderate the text content. If the content check is passed, an approval message will be sent to the system administrator, who can conduct random content check and show the moderated text comments on the web-end and app-end [14]. If the text content contains illegal content such as prohibited words, spams, or advertisement and fails to pass the moderation, it will be classified as unapproved type, and then, the AI-based TCM system can clarify the violation type of the text content, prompt alert to the user that the text he/she entered contains illegal content, and point out the violation type, and the relevant information will be recorded in the system log; for the type of the violation that cannot be determined by the system, the text will be pushed to the system administrator for manual moderation.

After manually moderated by the system administrator, text comments that meet the moderation criteria will be published on the web-end and the app-end; at the same time, the system will mark the text content and return it to the database of the AI-based TCM system, so that the text data model can perform update and self-learning; for text content that fails to pass the manual moderation of system administrator, after prompting to the user that the text comment could not be published due to the existence of illegal content, the system will mark the text content, so that the AI-based TCM system can automatically analyse, judge, and moderate similar text content in the future [15]. The operation process of the AI-based TCM system is shown in Figure 3 (see Figure 3).

3. Implementation of the AI-Based TCM System

3.1. System Implementation. According to the settings of different functional modules, the education video resource service system puts a limit on the number of characters in text comments. In the mobile live broadcast module, since the real-time bullet screen comments move fast, comments with many words will be refreshed very quickly; therefore, 50

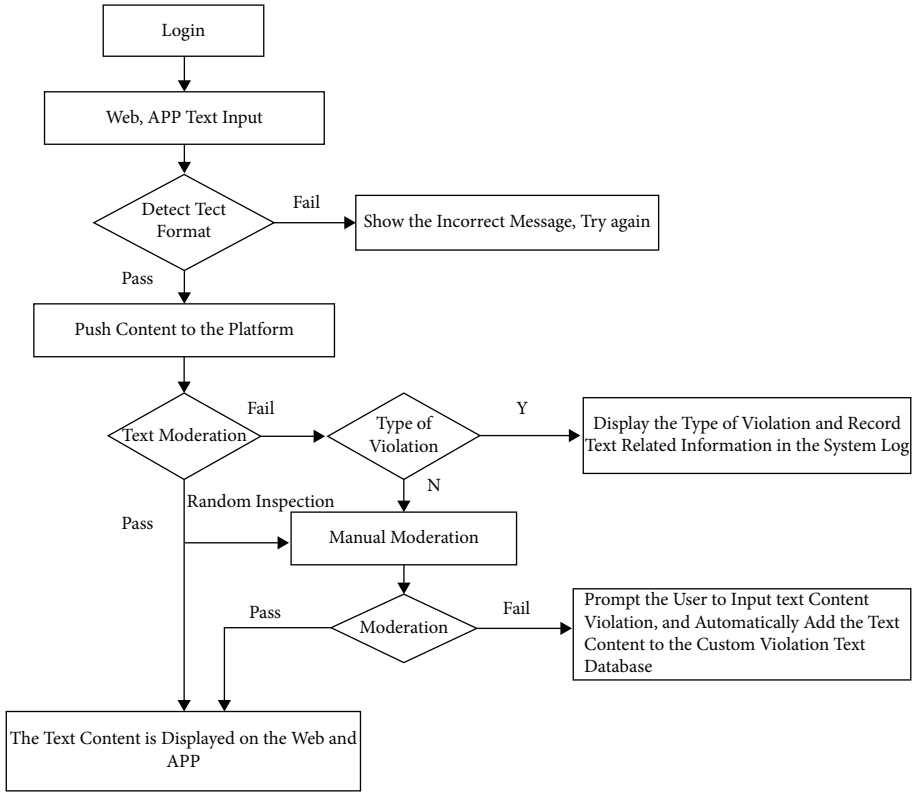

Figure 3: Operation process of the AI-based TCM system.

characters have been set as the limit of each bullet screen comment to facilitate other users' fast understanding of the comment content. In the VOD module, 260 characters have been set as the limit of comments on the videos published by users. In the video clips module, this number has been set to 90 characters. Therefore, the AI-based TCM system was developed based on these character limits of the education video resource service system.

By calling SDK (Software Development Kit), the system encapsulates the moderation API and completes interface docking by passing parameters, and then, it can directly call the interface function provided by the SDK to use the intelligent moderation function [16]. System developers do not have to care about issues such as protocol, encryption, or decryption. The system implementation process is as follows:

(a) The cloud-end management console opens the AI service function and completes works such as user information authentication, system access key authentication, and user ID authentication [17].

(b) The cloud-end management console performs visual modeling. There are many built-in AI algorithms on the cloud platform, such as machine learning and deep learning. In terms of scenarios, it could realize natural language processing and structured data modeling [18]. The cloud-end management console can establish independent models and rules according to different requirements. In this study, the FastText algorithm provided by the cloud server was selected to construct the deep-learning text classification model. The FastText algorithm contains 3 parts: model architecture, hierarchical Softmax, and N-gram features. (1) The model architecture: the FastText model enters a word sequence and outputs the probability of the word sequence belonging to different classes. The words and phrases in the sequence form an eigenvector, which is mapped to the intermediate layer through linear transformation and then mapped to the labels. FastText uses a nonlinear activation function for label prediction. (2) The hierarchical Softmax: some text classification tasks involve many classes, making it complex to compute a linear classifier. To shorten the runtime, the FastText model adopts hierarchical Softmax technique. Based on the Huffman code, the hierarchical Softmax encodes the labels and minimizes the number of model prediction targets. (3) N-gram features: the FastText model adds N-gram to the input word sequence, aiming to prevent the loss of the word order. Specifically, the N-gram is treated as a word and represented by an embedding vector. During the calculation of the hidden layer, the embedding vector of each N-gram is incorporated into the summation and averaging operations. The hash bucket is employed to hash all N-grams into a number of buckets. All the N-grams in the same bucket share the same embedding vector. The process of visual modeling can be divided into

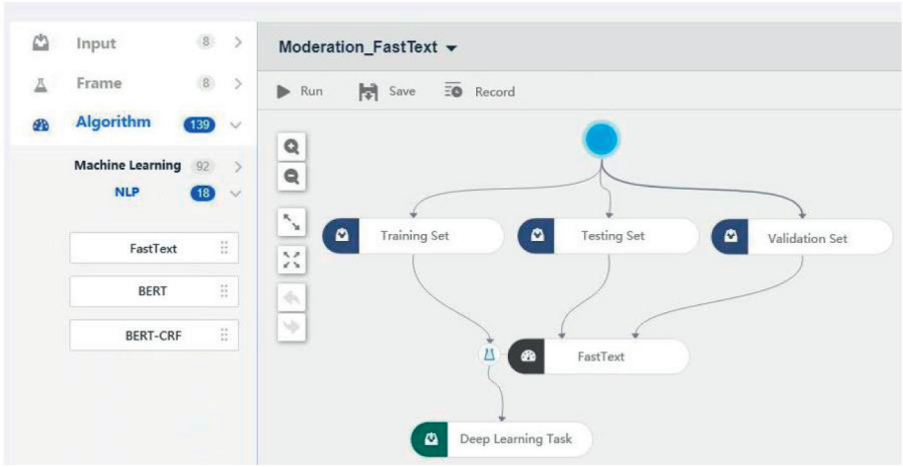

Figure 4: Model construction under the cloud management interface.

several steps of project creation, data preparation, model building, and model evaluation.

(i) New project creation: create a new project named "Moderation" in the cloud. In the new project, multiple workflows could be created to perform various processing on the text content according to requirements.

(ii) Data preparation: in this study, a Weibo sentiment analysis dataset containing locally collected and tagged data was adopted. Firstly, deduplication is performed on the repeated samples in the dataset. The samples with a high missing rate and the same value are deleted. Secondly, the data are regularized through case transition and spacing removal. Finally, the missing values are padded, and the abnormal values are processed. After the preprocessing, the dataset contained about 360,000 Weibo posts with sentiment tags, about 200,000 of which were happy, and posts with other sentiments of angry, disgust, and low were about 50,000 each. The dataset was divided into three parts: a training set, a validation set, and a test set, and all of them were uploaded to the AI platform. The training set was used to train the model, the validation set was used to adjust the model parameters, and the test set was used to evaluate the overall performance of the model. The model construction under the cloud management interface is shown in Figure 4 (see Figure 4).

(iii) Use the FastText algorithm to build the text classification model. Configure the FastText model using the cloud console, and complete the setting of algorithm parameters. The paths of training data input, verification data input, and model directory are generated automatically according to the connecting lines. The dimensions of word vector are set to 300, the batch size to 32, the training epochs to 5, and the learning rate to 0.001. Whether to use the pretrained word vector was configured as False and the resource type as on-demand.

(iv) Model performance evaluation: the platform has built-in visual components for model evaluation. In this study, the deep-learning classification task was adopted for evaluation. The prediction effects of this model are evaluated by the deep-learning classification tasks of Tencent Cloud TI platform, and the evaluation indices include Accuracy, Precision, Recall, and F1-score. On the left navigation bar of the console, "Output," "Model Evaluation," and "Depth Learning Classification Task Evaluation" are selected in turn and dragged into the canvas. The output pin is connected to "Deep Learning Classification Task Evaluation." In addition, the tagged sequence number of the evaluation parameter was set to 2, the prediction sequence number was set to 1, and other parameters took the default values, and then, the prediction effect of the established text classification model was evaluated. (c) Set up customized white list and black list vocabulary libraries, and add whitelist keywords and blacklist keywords. (d) Write codes to perform a series of operations such as obtaining system access key authentication, obtaining intelligent moderation SDK software package, environment configuration, and SDK project import, and using SDK for development; then call the moderation model API that had been built on the cloud platform. (e) Sample codes of the calling are as follows:

(i) Configure the user access key authentication in the java file, and generate the corresponding client-end connection object.

```
AisAccess
service = ServiceAccessBuilder.builder()
. ak("######")//User access key
. region("#####")//Set the region where the intelligent moderation function is provided
. connectionTimeout(5000)//Timeout limit for connecting target url
. connectionRequestTimeout(1000)//Timeout limit for connection pool to get available connection
```

. socketTimeout(20000)//Timeout limit for obtaining serve response data
.build();

(ii) Enter the text to be reviewed, configure parameters and check.

```
String uri = "/moderation_FastText/text";//The configured text content moderation model module
JSONObject json = new JSONObject();//Create a new JSON object
json.put("categories," new String[] {"porn," "politics," "flood," "ad,"});//Check content type: pornographic, political, spam, or advertisement
JSONObject text = new JSONObject();
text.put("text," "xxxxxxxx");//Text content input position
text.put("type," "content");
JSONArray items = new JSONArray();
items.add(text);
json.put("items," items);
StringEntity stringEntity = new StringEntity(json.toJSONString(), "utf-8");
```

HttpResponse response = service.post(uri, stringEntity);//Pass in the uri parameter corresponding to the text content moderation service and other parameters required by the text content moderation service, the JSON object method is mainly adopted for parameter passing, and the POST method is used to call the service.

ResponseProcessUtils.processResponseStatus(response);//Verify the returned calling status; if it displays correct codes, then the calling is successful; otherwise, it is failed.

3.2. Function Realization of the AI-Based TCM System. The main function of the proposed system is to analyse and detect the text content submitted from multiple terminals to the education video resource service system [19]. The system could realize following functions:

(a) Intelligent moderation: using massive tag data and deep-learning algorithm, the system can perform multidimensional real-time detection on the text information in the comment area or interaction area input from web-end or app-end, and judge whether the text content has violated any regulation or not.

(b) Manual moderation: for text content types that cannot be determined by the intelligent moderation module, the system can submit them for manual moderation. The system administrator uses the system to recheck the published content that has been moderated by the system. After the manual moderation is completed, the system automatically records the processing operations of the system administrator and adds the text content this time into the sample library as a new sample.

(c) Automatic update of text sample library: the system can continuously and quickly upgrade the text sample library; the system administrator can merge the local sample library with the cloud sample library and identify new types of illegal content in a timely manner.

(d) System management console: the system can perform content safety management, user management, system configuration, and other operations; it also can conduct content safety detection configuration, such as define the detection intensity level and manage the vocabulary database.

(e) Algorithm model configuration: if the detection effect of the algorithm model does not meet requirements, the system administrator can adjust the parameters of the original algorithm model or use a different algorithm model for testing [20]. The following models of the model can be adjusted: (1) separator—the separator for separating sentences and labels; (2) word vector dimensions—the dimensions of the word vector in the network; (3) batch size—the size of the batch of training samples; (3) training epochs—the number of trainings for the training data; (4) learning rate—the rate of learning (learning_rate); (5) whether to use the pretrained word vector—if it is set to True, the path of the word vector file can be filled in; the file format is the same as the official format of glove word vector.

Table 1: Moderation accuracy of the proposed system in the test.

	Normal text	Advertising	Spam
Sample size	60	30	30
AI confirm	60	29	29
AI suspicion	0	1	1

4. Test of the AI-Based TCM System

To test the accuracy of the proposed system, our research team had compiled and collected 120 text posts from the Internet, each text contained 20–150 characters, and the text samples included 60 normal comments, 30 advertisements, and 30 spams. Taking the web-end VOD comments as an example, text samples were input to the proposed system for testing. The proposed system completed the moderation within 1 millisecond and returned the results. According to final statistics, 118 texts had been directly identified by the system, 2 texts were identified as suspected illegal texts by the system and submitted for manual moderation; the moderation accuracy was 98.3%, a very high value. The moderation accuracy of the proposed system in the test (see Table 1).

Using the back-end content moderation list, the system administrator browses and moderates all comments that had passed the detection. The system records the new manual moderation operations, and returns the moderation results and text content to the back-end, and then, it performs sample analysis, tagging, and training, modifies the cloud-end moderation model, expands the feature sample library, and personalizes the moderation strategies, keeping the moderation criteria being in an optimal state. Using the

Content Moderation List							
Content	User	Moderation	Type	Admin	Dimension	Confidence	Time
!!@@##$$^^^^^********	xs2005	Fail	AI	Auto	Invalid Text	0.998	2020-12-10 10:50:22
This party is really great	yjs1955	Pass	AI	Auto			2020-12-10 10:52:01

Figure 5: The back-end content moderation list.

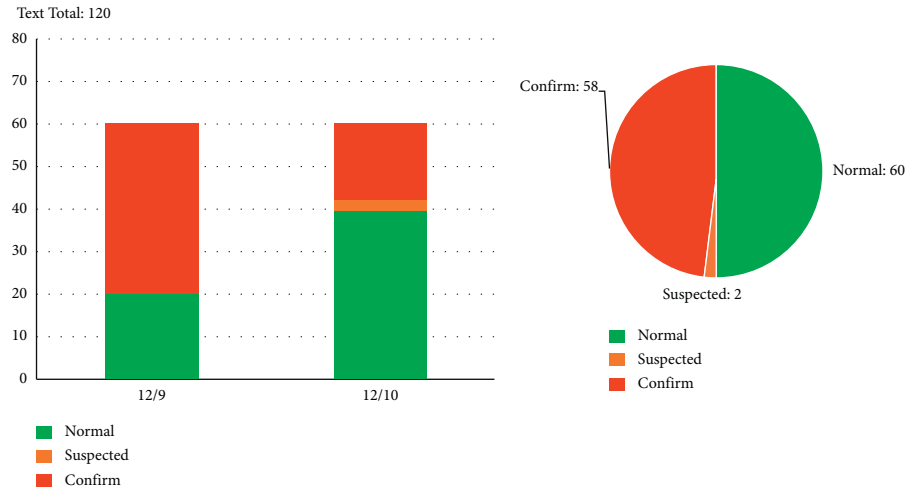

Figure 6: Statistics of text content moderation.

query function, the system administrator can count and analyse the numbers of normal texts, suspected illegal texts, and confirmed illegal texts in the comments, and plot the statistical results in the form of histogram or pie chart. Figure 5 shows the back-end content moderation list (see Figure 5); Figure 6 shows the statistics of the text content moderation (see Figure 6).

5. Conclusion

This article proposed an AI-based TCM system, which was developed on an AI cloud service platform, and the system could automatically and intelligently analyse and detect the text content input by users from the web-end and app-end by calling built-in algorithm models on the cloud-end. With the help of the proposed system, the workload of manual moderation had been greatly reduced, and fast and efficient text moderation had been realized. The computation workload of cloud-end text content moderation can be controlled flexibly according to the use scenarios, and if the computation workload is heavy, the concurrent moderation capacity will be expanded; if the computation workload is small, idle resources will be quickly released. In this way, the system can ensure moderation efficiency under frequent computation workload changes and effectively reduce the moderation costs.

As higher education is heading towards a digitized and networked development direction, the proposed system can effectively detect and manage the ever-increasing real-time interactive text comments, and it could play a positive role in creating a healthy, safe, and civilized network environment, and ensuring the safety and stability of the campus and the society.

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 that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the Research and Development Planning Project of Shandong Province, China (year: 2019; Grant no. 2019GGX105019); Education and Teaching Reform Research Project of Shandong University (Year: 2020; Grant no. 2020Y288); and Construction and Management Research for Laboratory Major Project of Shandong University (year: 2020; Grant no. SY20201101).

References

[1] S. Q. Guo, J. J. Huang, and Q. F. Yuan, "A review of the development of mobile learning applications abroad," *E-Education Research*, vol. 5, pp. 105–109, 2011.

[2] J. Chen, "Design and realization of the content management system of TV," *Broadcasting and Television Technology*, vol. 4, pp. 4–10, 2015.

[3] S. Wang, Z. Wang, and H. Ren, "Research on fusion model based on deep learning for text content security enhancement," *Telecommunications Science*, vol. 36, no. 5, pp. 25–30, 2020.

[4] M. Liu and G. Huang, "Research on harmful text filtering model based on semantic analysis," *Journal of Chinese Information Processing*, vol. 32, no. 2, pp. 126–131, 2017.

[5] S. Myers West, "Censored, suspended, shadowbanned: user interpretations of content moderation on social media platforms," *New Media & Society*, vol. 20, no. 11, pp. 4366–4383, 2018.

[6] A. Heldt and S. Dreyer, "Competent third parties and content moderation on platforms: potentials of independent decision-making bodies from A governance structure perspective," *Journal of Information Policy*, vol. 11, no. s1, pp. 266–300, 2021.

[7] Q. Chen, X. Hu, and H. Yuan, "A large-scaled platform of education resources based on cloud services," *Modern Educational Technology*, vol. 21, no. 12, pp. 112–115, 2011.

[8] E. A. Ahmed and H. Ali Ahmed, "A proposed model for education system using cloud computing," in *Proceedings of the 2018 3rd International Conference on Emerging Trends in Engineering, Sciences and Technology*, December 2018.

[9] J. Seering, "Reconsidering community self-moderation: the role of research in supporting community-based models for online content moderation," *Proceedings of the ACM on Human-Computer Interaction*, CSCW2, vol. 4, , p. 107, 2020.

[10] B. Cameron, E. Crawley, and D. Selva, *Systems Architecture. Strategy and Product Development for Complex Systems*, Pearson Education, 2016, https://www.google.com/search?rlz=1C1GCEB_enIN990IN990&q=London&stick=H4sIAAAAAAAAAOPgE-LUz9U3MC8wrSpW4gAxTbIKcrS0spOt9POL0hPzMqsSSzLz81A4VhmpiSmFpYlFJalFxYtY2Xzy81Ly83awMgIA61Psi08AAAA&sa=X&ved=2ahUKEwjr976Lyd71AhXZ8XMBHbFNCGwQmxMoAXoECDgQAw.

[11] W. Zhu, H. Gong, R. Bansal et al., "Self-supervised euphemism detection and identification for content moderation," in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 229–246, Francisco, CA, USA, May 2021.

[12] G. Uddin, F. Khomh, and C. K. Roy, "Mining API usage scenarios from stack overflow," *Information and Software Technology*, vol. 122, Article ID 106277, 2020.

[13] P. Coutinho and J. Rui, "Moderation techniques for user-generated content in place-based communication," in *Proceedings of the 2017 12th Iberian Conference on Information Systems and Technologies*, June 2017.

[14] A. Veglis, "Moderation techniques for social media content," *Social Computing and Social Media*, in *Proceedings of the International Conference on Social Computing & Social Media*, pp. 137–148, Heraklion, Crete, Greece, June 2014.

[15] M. J. Kavis, *Architecting the Cloud: Design Decisions for Cloud Computing Service Models*, Wiley, 2021, https://www.google.com/search?rlz=1C1GCEB_enIN990IN990&q=Hoboken&stick=H4sIAAAAAAAAAOPgE-LUz9U3MMotLypT4gAxK_LMk7S0spOt9POL0hPzMqsSSzLz81A4VhmpiSmFpYlFJalFxYtY2T3yk_KzU_N2sDICAC-vG35QAAAA&sa=X&ved=2ahUKEwiZpKvvzN71AhVt7HMBHY68D08QmxMoAXoECCYQAw.

[16] D. M. Riehle, M. Niemann, J. Brunk, D. Assenmacher, H. Trautmann, and J. Becker, "Building an integrated comment moderation system - towards a semi-automatic moderation tool," *Lecture Notes in Computer Science*, in *Proceedings of the International Conference on Human-Computer Interaction*, pp. 71–86, Heraklion, Crete, Greece, June 2020.

[17] J. A. Panattoni, R. McAlpine, K. B. Bobade, M. J. Wilson, and C. Willy, "Escalation of machine-learning inputs for content moderation," *U.S. Patent Application*, vol. 331, 2018.

[18] J. Pavlopoulos, P. Malakasiotis, and I. Androutsopoulos, "Deeper attention to abusive user content moderation," in *Proceedings of the EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing*, pp. 1125–1135, Copenhagen, Denmark, September 2017.

[19] V. Maddumala and R. Arunkumar, "Big data-driven feature extraction and clustering based on statistical methods," *Traitement du Signal*, vol. 37, no. 3, pp. 387–394, 2020.

[20] S. Jhaver, S. Ghoshal, A. Bruckman, and E. Gilbert, "Online harassment and content moderation: the case of blocklists," *ACM Transactions on Computer-Human Interaction*, vol. 25, no. 2, Article ID 3185593, 2018.