Research Article

e-Commerce Online Intelligent Customer Service System Based on Fuzzy Control

Dongmei Wei [ID]

School of Computer and Software Engineering, Xihua University, Chengdu 610039, China

Correspondence should be addressed to Dongmei Wei; wdm@mail.xhu.edu.cn

In order to improve the intelligence of the e-commerce online intelligent customer service system, this paper proposes a deep rejection recognition algorithm based on the maximum interval squared hinge loss and combines the actual needs of the e-commerce online customer service system to build an intelligent customer service system with the support of the fuzzy control system. Moreover, this article chooses to build a domain ontology library for structured storage of domain knowledge needed by customer service chatbots. In addition, this article analyzes the dialogue structure based on the speech act model and combines the semantic vector model of the question sentence on the basis of the dialogue structure to understand the question sentence, which helps to improve the accuracy of the answer feedback of the Internet shopping customer service robot. Finally, this article designs experiments to verify the performance of the online customer service system constructed in this article and analyzes the experimental results through statistical methods. The experimental results show that the online intelligent customer service system constructed in this paper has certain practical effects.

1. Introduction

With the rapid development of Internet e-commerce, people's lifestyles are becoming more and more convenient, and people are accustomed to shopping online [1]. The new behavior and habits gave birth to a new position-customer service. In online shopping, customers often inquire about products, place orders, logistics, after-sales, refunds, and other information, which require customer service to answer and deal with. The customer service work is arduous, boring, and highly repetitive. In reality, manual customer service often fails to deal with customer problems in a timely manner, which leads to reduced customer satisfaction, loss of customers, and impact on sales performance. Therefore, artificial intelligence-oriented customer service robot products for e-commerce came into being. The e-commerce customer service robot frees customer service from a large number of repetitive mechanical answer questions and handles complex problems in a timely manner. Moreover, it can improve customer satisfaction while also reducing business operating costs. According to the analysis of the company's actual situation, a project was established

to build a customer service robot system oriented to the e-commerce field. Customers have priority access to the customer service robot system, which can automatically answer customer questions, recommend customer products based on the related products inquired, and recommend questions that customers may ask in advance. The e-commerce customer service robot is not like a mecha robot in a science fiction film but an intelligent question-and-answer system for the e-commerce field [2]. Moreover, it can understand and analyze the user's intention based on the analysis of the customer's intention and then call the corresponding matching rule system to deal with the problem and then answer the question better and accurately. Technically, customer service robots are mainly based on natural language analysis- and machine learning-related algorithms to automatically answer. The knowledge base of the customer service robot uses collected and organized conversations between human customer service and customer chats. This system can effectively help corporate customer service departments to provide 24-hour service for uncomplicated customer problems. Moreover, the service is stable, and users will not be aware of the difference in service

due to the quality of customer service. In the future, it can also analyze purchase behaviors based on purchase records and flexibly provide users with personalized recommendations and fashion advice pushes. In addition, it saves manual customer service from simple and boring answering questions, so that manual customer service can give priority to serving customers with complex problems, and it can manage user satisfaction and allow users to continue to consume. Finally, it can achieve the purpose of reducing the operating cost of enterprise customer service and improving user experience [3].

On the basis of the above analysis, based on fuzzy control technology, this article combines the actual needs of e-commerce customer service to construct an e-commerce online intelligent customer service system based on fuzzy control and analyzes the performance of the system.

2. Related Work

With the continuous huge amount of Internet data and the development of computer linguistics, question-answering systems have begun a new life. It has broken away from traditional information retrieval and developed into an independent research direction. Under the promotion of TREC, the development of automatic question answering technology has been greatly promoted. The question answering system of this period mainly obtained the answers to the user extraction questions from a large number of free text collections [4]. The language handled by TREC is English, so many technologies cannot be used in languages of Eastern countries [5]. Fact-based problems are the best type of problem solved so far and have achieved good results, but the research on other types of problems has been slow [6]. Corresponding to TREC is the evaluation task CLEF for European languages, which also joins the QA branch [7]. In addition to the question answering system based on free text, some people have proposed a question answering system based on network links or metadata results in recent years [8].

In recent years, with the development of online communities, a community question answering system with common question answering pairs as the knowledge base has emerged [9]. With the increase in interactive requirements in the question answering process, interactive question answering has appeared in recent years. Interactive question answering is a research field between the traditional question answering system and the dialogue system [10]. Interactive question answering inherits the characteristics of traditional question answering, and it allows users to ask questions of the system in natural language and gives answers in natural language or in a way that users can understand [11]. Different from traditional question answering technology, it allows users to conduct multiple rounds or further questions when they are not satisfied with the answer. The interactive question answering system based on traditional question answering adds interactive functions to the traditional question answering system, such as adding continuous question processing and contextual information processing. Although for question answering systems, allowing users to interact with the system through natural language for multiple consecutive rounds is a very important improvement; for dialogue systems, such continuous multiple rounds of human-computer interaction have existed for a long time [12]. Essentially, any dialogue system with the purpose of information consultation can be called an interactive question answering system [13].

Customer service staff often have to face a large number of customers consulting the same problem, which not only causes a waste of labor but also reduces work efficiency. As a result, the e-commerce assistant of information came into being; IBM's chat dialogue system NLA assistant [14] can help customers find the products they want through dialogue with customers and customer feedback. In each conversation, NLA provides incremental feedback to customers to feedback their understanding of customer needs and to show customer products that meet their needs. The B2C e-commerce question answering robot developed in the literature [15] can answer questions in the field of digital cameras and can be extended to multiple dedicated fields. The context-sensitive sales assistant developed in the literature [16] can intelligently communicate with customers and help customers complete the purchase of goods.

Compared with the difference in knowledge, service attitude, and style of manual customer service personnel, the intelligent customer service system can provide uninterrupted, consistent, and efficient customer consulting services. This can not only increase the satisfaction of customer consulting services in the case of a large number of customers and insufficient customer service staff but also greatly reduce the labor cost of the enterprise.

3. Intelligent Machine Semantic Analysis Function Based on Fuzzy Control

The feedforward neural network can be regarded as a series of nonlinear activation input and the function transformation of its linear combination. To solve this series of parameters, a loss function needs to be defined, and the training of the network is realized by minimizing this loss function. The simplest method is to minimize the sum of square error. The input $\{(x_n, t_n)_{n=1}^{N}\}$ is given, and t_n is the label corresponding to x_n.

$$y_n = f(x_n, w). \tag{1}$$

The above formula is the output of the neural network, and the minimized error function is

$$E(w) = \frac{1}{2} \sum_{n=1}^{N} \|y_n - t_n\|^2. \tag{2}$$

The following results can be obtained by seeking the partial derivative of the k-th term:

$$\frac{\partial E}{\partial y_k} = y_k - t_k. \tag{3}$$

This is the error between the output and the true value,

and the network training is performed by backpropagating this error update parameter.

If the output of the network is a probability estimate of t_n, the parameters of the neural network can be determined by the maximum likelihood framework. We set $X = \{x_1, \cdots, x_n\}$ and $t = \{t_1, \cdots, t_N\}$ and construct the likelihood function:

$$p(t|X, w) = \prod_{n=1}^{N} p(t_n|x_n, w). \quad (4)$$

It is not difficult to understand that the smallest error means the largest probability of outputting t_n. Therefore, minimizing the sum of square error is equivalent to maximizing the likelihood function. The specific two categories are as follows. The output y_n of the network is the probability estimate of $t_n = 1$, t_n obeys the Bernoulli distribution $p(t_n = 1|x_n, w)$, and the probability corresponding to $t_n = 0$ is $p(t_n = 0|x_n, w) = 1 - y_n$. Then, the probability distribution of t_n can be written as

$$p(t_n|x_n, w) = p(t_n|y_n) = y_n^{t_n}(1 - y_n)^{1-t_n}. \quad (5)$$

Taking the negative logarithm of the likelihood function, the following results can be obtained:

$$-\log (p(t|X, w)) = -\sum_{n=1}^{N} \log(p(t_n|x_n, w))$$
$$= -\sum_{n=1}^{N} \{t_n \log (y_n) + (1 - t_n) \log (1 - y_n)\}. \quad (6)$$

This produces an error function in the form of cross-entropy. Similarly, the cross-entropy form of the K classification problem is

$$E(w) = -\sum_{n=1}^{N} \sum_{k=1}^{K} t_{nk} In y_{nk}. \quad (7)$$

The outlier detection algorithms of most deep learning frameworks are hybrid models, which perform feature extraction through neural networks and input "nondeep" outlier detection algorithms. Some recent works try to introduce the loss function in outlier detection into the neural network to realize an end-to-end model.

The objective function of SVDD is used to replace the cross-entropy loss, which minimizes the volume of the hypersphere surrounding the feature vector while training the neural network. Minimizing the volume of the hypersphere can closely map the data point to the center of the sphere:

$$\min_{R,W} R^2 + \frac{1}{vn} \sum_{i=1}^{n} \max\{0, \|\phi(x_i; W) - c\|^2 - R^2\} + \frac{\lambda}{2} \sum_{l=1}^{L} \left\| W^l \right\|_F^2. \quad (8)$$

Among them, $F \subseteq \mathbb{R}^p$ is the p-dimensional feature space and $\phi(x; W): X \longrightarrow F$ represents the neural network with the L-layer weight parameter set as $W = \{W^1, \cdots, W^L\}$. R is the volume of the hypersphere, and $c \in F$ is the center of the hypersphere. The hyperparameter $v \in (0, 1)$ controls the trade-off between the volume of the hypersphere and the error point, that is, the ratio of outlier points. When $\lambda > 0$, the regularization term is adjusted. The algorithm uses stochastic gradient descent or its variants (such as Adam) for parameter optimization. However, since W and R exist in different dimensional spaces, these two parameters need to be optimized alternately during the iterative process, which is equivalent to continuously solving two optimization problems.

The application of a single-class support vector machine loss function in deep networks is expanded:

$$\min_{w,V,\rho} \frac{1}{2} \|w\|_2^2 + \frac{1}{2} \|V\|_F^2 + \frac{1}{v} \frac{1}{N} \sum_{n=1}^{N} \max(0, \rho - \langle w, g(VX_n)\rangle) - \rho. \quad (9)$$

Among them, w is the scalar output obtained from the hidden layer to the output layer, that is, the unactivated output of the entire network; V is the weight matrix; and ρ is the offset of the classification hyperplane. Compared with the loss function of OC-SVM, the key method of the above formula is to replace the dot product $\langle w, \phi(X_n)\rangle$ with the dot product $\langle w, g(VX_n)\rangle$, that is, using the output of the penultimate layer to calculate the loss function. This allows feature learning to be implemented in a neural network, and the last layer of the network is used to perfect the outlier detection function. However, after the change, the objective function becomes a nonconvex function, so the algorithm for optimizing model parameters will not lead to global optimization. The OC-SVM loss function has more parameters. Similarly, this algorithm also needs to alternate iteratively for parameter optimization.

When training a neural network, the choice of loss function is closely related to the choice of output unit. For example, how the most commonly used cross-entropy loss function represents the output determines the form of the cross-entropy function (two-class or multiclass).

For binary classification problems, the target variable is $t_n \in \{0, 1\}$. When $t_n = 0$, it means belonging to category c_0, and when $t_n = 1$, it means belonging to category c_1. We set a neural network with one output unit and set the input to x_n and the hidden layer output to $a = w^T x_n$. Using the sigmoid function as its activation function is mainly to compress the value range of the hidden layer output to $(0, 1)$ to represent the probability. The input x_n belongs to category c_1.

$$p(c_1|x_n) = \sigma(a) = \frac{1}{1 + \exp (-a)}. \quad (10)$$

At this time, the posterior probability of category c_0 is

$$p(c_0|x) = 1 - p(c_1|x). \quad (11)$$

Similarly, we set a standard multiclass neural network, and each input is divided into K mutually exclusive categories $\{c_1, c_2, \cdots, c_K\}$. In the output layer, the form of "1-of-K" is used to represent the category: the output unit of the corresponding category is 1 and the output unit of the other categories is 0. Softmax is used as the output unit activation function to predict the probability that x_n belongs to category c_k. We set the input as x_n and the output of the hidden layer as $a_k = w_k^T x_n$. Then, the conditional probability that x_n belongs to category c_k is

$$p_k = p(y = c_k | x_n) = \text{soft max}(a_k) = \frac{\exp(a_k)}{\sum_{k=1}^{K} \exp(a_k)}. \quad (12)$$

In this way, we can convert the K scalar outputs $\{a_1, a_2, \cdots, a_K\}$ of the hidden layer into a probability distribution.

$$\sum_{k=1}^{K} p_k = 1, \quad (13)$$

$$\forall k, p_k \in (0, 1).$$

The predicted result is

$$y_n = \arg\max_k p_k = \arg\max_k a_k. \quad (14)$$

For the k-th output unit, which is also a prediction unit of category c_k, the neural network uses data with label c_k as positive samples during training and the remaining label data as negative samples. When forecasting, we introduce a probability threshold λ to realize rejection recognition. For the k-th output unit, we check whether the probability $p_k = \text{sigmoid}(a_k)$ predicted by the k-th sigmoid function is less than the threshold λ_k of category c_k. If the predicted probabilities on all output units of an input sample are less than their corresponding threshold, the input is refused. Otherwise, the input is predicted to be the class with the highest probability value:

$$y = \begin{cases} \text{Refuse}, & \max_k p_k < \lambda_k, \\ \arg\max_k p_k < \lambda_k, & \text{otherwise}. \end{cases} \quad (15)$$

In the second classification, 0.5 is a general probability threshold.

Cross-entropy is an important concept in Shannon's information theory, which is mainly used to measure the difference information between two probability distributions. The purpose of using cross-entropy loss via the network is to reduce the error between the true distribution of the training data and the predicted distribution of the model. In addition to faster training speed and better generalization ability, another advantage of using cross-entropy lies in its versatility, which eliminates the need to design a special cost function for each model. However, for specific tasks, a customized cost function is obviously more useful.

The data in the rejection scenario is an open set, so the distribution of training samples cannot provide a reference for the true distribution of the overall data. The reason is that the unseen category causes the true sample distribution to be unpredictable. Therefore, reducing the difference between the distribution of training samples and the distribution of model predictions through cross-entropy has a limited effect on the estimation of real samples. The binary classification sigmoid function can be used to replace the softmax function. For the output of the current class, only two cases of belonging and not belonging to are considered. While the current class is known during training, the cross-entropy loss function becomes

$$E(w) = -\sum_{i=1}^{n} \sum_{k=1}^{K} \{t_{ik} In y_{ik} + (1 - t_{ik}) \text{In}(1 - y_{ik})\}. \quad (16)$$

Among them, $y_{ik} = y_k(x_i)$.

The objective function of SVM is called L1-SVM with standard hinge loss. Since L1-SVM is not continuously differentiable, it is not a good choice for neural network training. Therefore, we use its squared variant L2-SVM as the loss function of the rejection neural network, that is, the maximum interval squared hinge loss:

$$\min_{w,b} \frac{1}{2} \|w\|^2 + C \cdot \sum_{i=1}^{n} \max(0, 1 - y_i(w^T x_i + b))^2. \quad (17)$$

Using sigmoid activation in the output layer means that for the k-th output unit of the neural network, the data with label c_k is used as a positive sample during training, and the rest of the label data is used as a negative sample. With the help of the maximum interval squared hinge loss, the classification hyperplane with the largest separation between the two types of data can be found. The data imbalance problem caused by the OVR strategy just makes the classification boundary deviate to the "positive sample" side and obtains a compact boundary effect similar to the OC-SVM objective function, reducing the risk of the open area surrounding the classification boundary of the positive sample. Compared with DOC, the loss function of the entire network not only avoids the probability model but also affects the data representation in the hidden layer through error back propagation and obtains the sentence vector customized according to the rejection task. Janocha et al. proved through a large number of experiments that the squared hinge loss is a better choice than cross-entropy in deep neural networks for classification. Moreover, it is the best choice among the 12 experimental loss functions (including the square sum error), because it converges faster and is more robust to noise in the training data. In this article, we will only use the maximum interval squared hinge loss of the linear kernel to learn the parameters of the lower layer of the model by backpropagating the gradient from the loss function of the last layer. For this, we need to derive the loss function according to the activation value h_i of the penultimate layer.
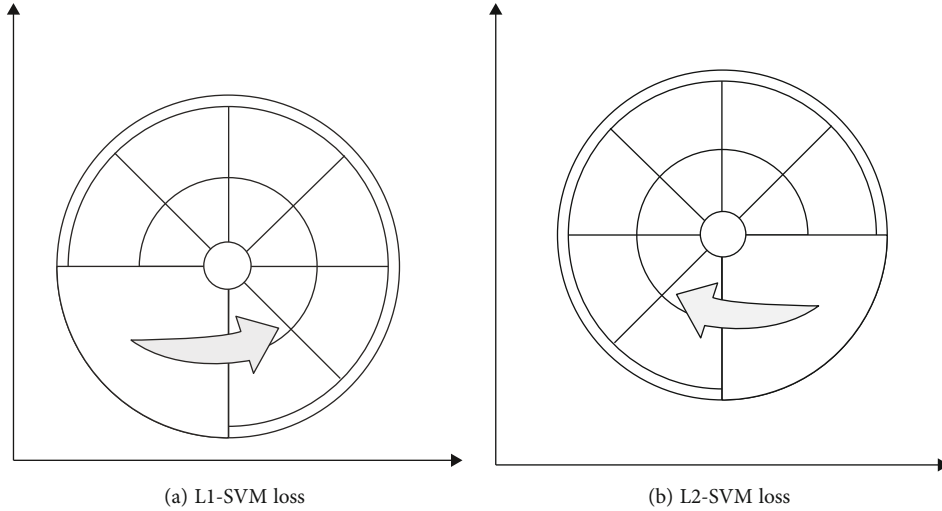
(a) L1-SVM loss

(b) L2-SVM loss

FIGURE 1: Classification effect of two types of hinge loss.

The input of the above formula is replaced with h_i.

$$L(w) = \frac{1}{2} w^T w + C \cdot \sum_{i=1}^{n} \max(0, 1 - w^T h_i y_i). \qquad (18)$$

For the loss function of L1-SVM,

$$\frac{\partial L(w)}{\partial h_i} = -C \cdot w^T y_i \cdot I\{w^T h_i y_i < 1\}. \qquad (19)$$

Among them, $I(\cdot)$ is an indicator function. If $\cdot$ is true, the function value is 1; otherwise, the function value is 0. The partial derivative of L2-SVM for the activation value of the hidden layer is

$$\frac{\partial L(w)}{\partial h_i} = -2C \cdot y_i \cdot w \left(\max(0, 1 - w^T h_i y_i)\right). \qquad (20)$$

We found that L2-SVM is slightly better than L1-SVM in most cases, as shown in Figure 1. In the feature space, L2-SVM produces a more clear classification boundary than L1-SVM, so we will use L2-SVM in the experimental part.

We use the adaptive gradient descent (AdaGrad) algorithm and its variants (such as Adam) to optimize the parameter W of the neural network, and the training is carried out until it converges to the local minimum. At the same time, we use AdaGrad to solve the model parameters, so that the deep OVR-SVM network can be well extended to large datasets, because its computational complexity increases linearly in the number of training batches, and each batch can be processed in parallel (for example, each batch can be processed on multiple GPUs).

Compared with OC-SVM and SVDD loss function, OVR-SVM is more convenient in parameter solving. The reason is that in addition to the parameter W, OC-SVM has an additional parameter ρ, and SVDD has the parameter radius r of the hypersphere. Both of these algorithms need to alternately optimize two parameters in the iterative process,

while our algorithm only needs to pay attention to one parameter C. When OVR-SVM is used as a decision machine, each decision function $f(x) = (w_k^T x) = (a_k)$ has only two results, +1 and −1; when $K > 2$, the prediction results of multiple support vector machines will conflict. Therefore, we use the calculated probability obtained by the sigmoid function to make predictions:

$$y = \arg \max_k p_k = \arg \max_k (\text{sigmoid}(a_k))4. \qquad (21)$$

For the binary classification problem, 0.5 is a natural choice as the probability threshold for classification, but this threshold does not consider the potential open-set space risk from unseen classes. Therefore, we can reduce the risk of open space by increasing the threshold.

In order to get a better threshold λ, we introduce a statistical model to select the outlier threshold. Normal distribution is one of the most important distributions in probability theory and mathematical statistics. It is Gauss's first use to describe the distribution of errors when studying error theory, so it is also called the Gaussian distribution. The central limit theorem shows that if a random variable is the result of the superposition of a large number of small and independent random factors, then this variable can generally be considered to obey a normal distribution. Therefore, many random variables can be described by normal distribution or approximate description, such as measurement error, product weight, and annual rainfall, which can be described by normal distribution.

If the density function of the random variable X is

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \quad -\infty < x < \infty, \qquad (22)$$

then it is said that X obeys a normal distribution, denoted as $X \sim N(\mu, \sigma^2)$, where $-\infty < \mu < \infty, \sigma > 0$. $p(x)$ is a bell-shaped curve, left and right are symmetrical with respect to μ, the
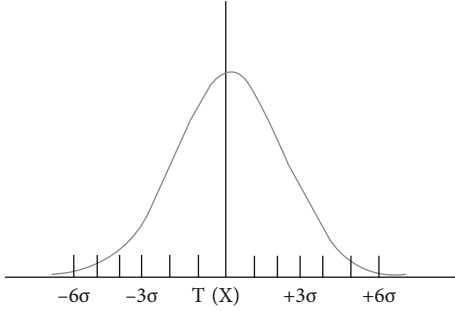
Figure 2: The 3σ rule of the normal distribution.

probability of taking a value near $x = \mu$ is the largest, and the probability of taking a value on both sides is the smallest. $\mu \pm \sigma$ is the inflection point of the curve.

The 3σ rule (three-sigma rule of thumb) means that "almost all" values are within the range of plus or minus three standard deviations of the mean, as shown in Figure 2. Specifically, if the random variable is $X \sim N(\mu, \sigma^2)$, then

$$p(|X - \mu| < k\sigma) = \begin{cases} 0.6826 & k = 1, \\ 0.9545 & k = 2, \\ 0.9973 & k = 3. \end{cases} \quad (23)$$

It can be seen from the above formula that although the value range of the normal variable is $(-\infty, +\infty)$, 99.73% of its value falls within $(\mu - 3\sigma, \mu + 3\sigma)$. This property is called the "3σ rule" of the normal distribution. This is very useful in practical work. For example, industrial production control charts and some product quality indexes are all formulated according to the "3σ rule." Therefore, we can set the rejection threshold according to the "3σ rule."

For each output unit, the expected probability value of the positive sample is 1, but the probability value generated by the sigmoid transformation is always less than 1, which cannot satisfy the symmetry of the Gaussian distribution. Therefore, we assume that the predicted probability of each type of training data obeys a half-Gaussian distribution of mean $\mu = 1$ and then artificially construct the other half of the data. For the probability $p(c_k|x_i)$ of the point x_i, we construct a mirror point to make its probability $2 - p(c_k|x_i)$ satisfy the mean value of 1. The reason is that this mirror point is not a probability. The probability of each type of positive sample and the corresponding mirror point are used to fit a Gaussian distribution to the output of each type to obtain the standard deviation σ_k. According to need, the threshold of rejection probability is

$$\lambda_k = \max(0.5, 1 - \alpha\sigma_k), \quad \alpha = \{1, 2, 3\}, \quad (24)$$

because each class is fitted with the Gaussian distribution and each class has its own threshold.
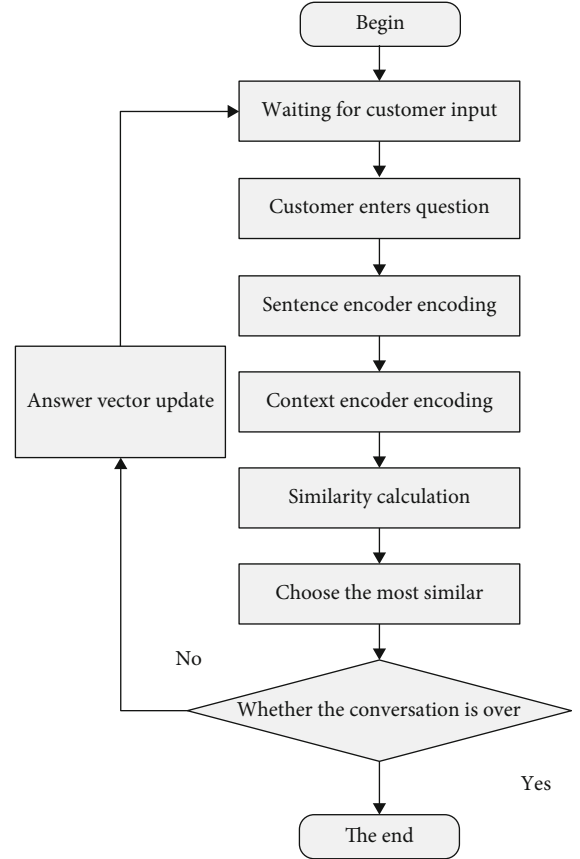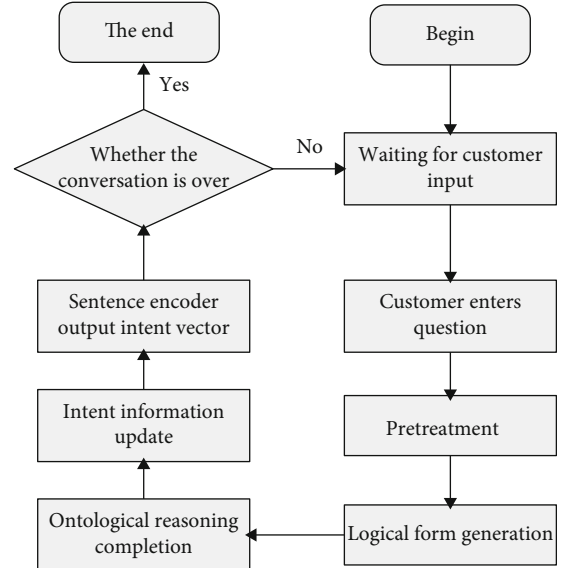

Figure 3: Flow chart of the model dialogue.

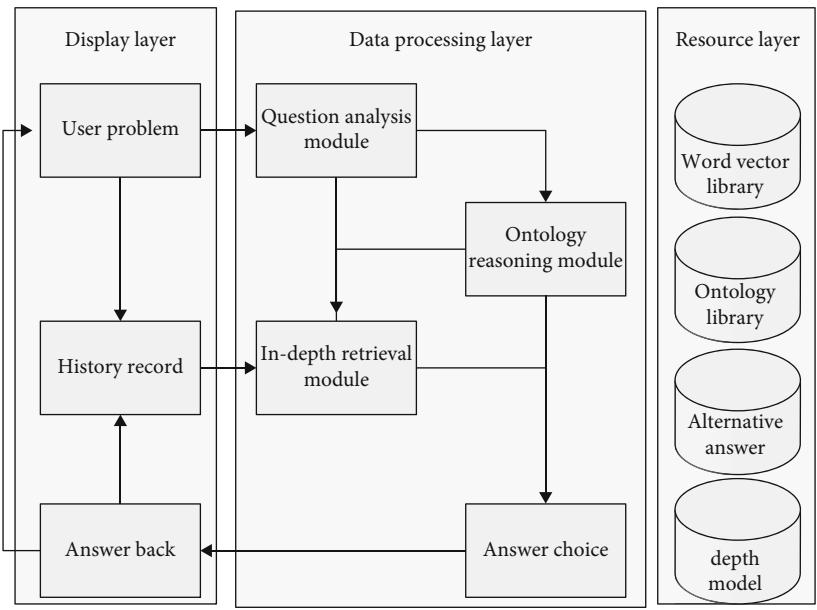Figure 4: Flow chart of the constructing intention vector.
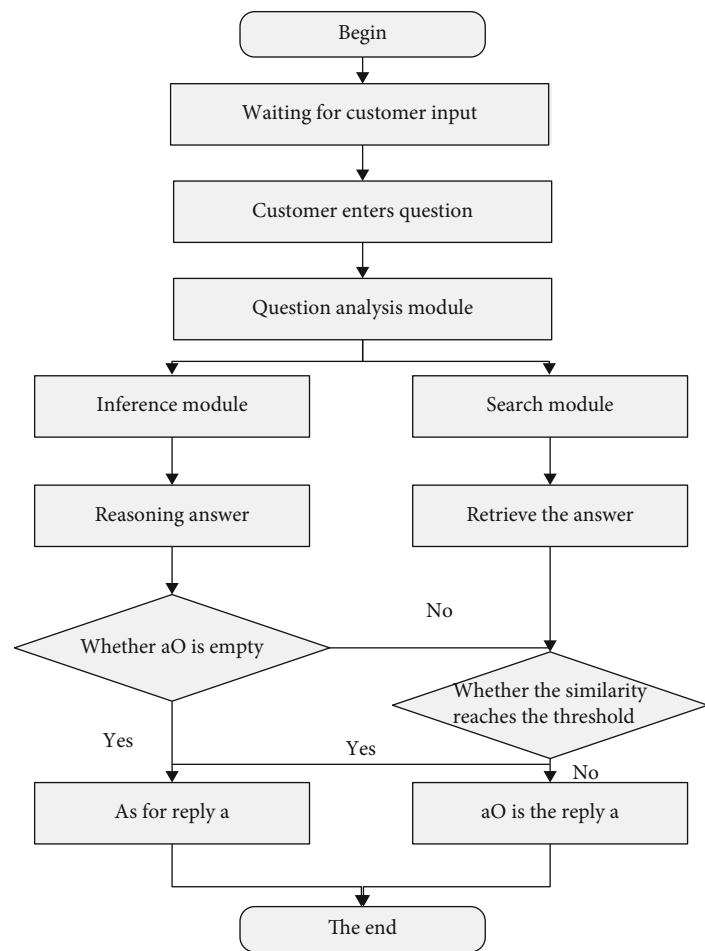
FIGURE 5: The overall architecture of the system.
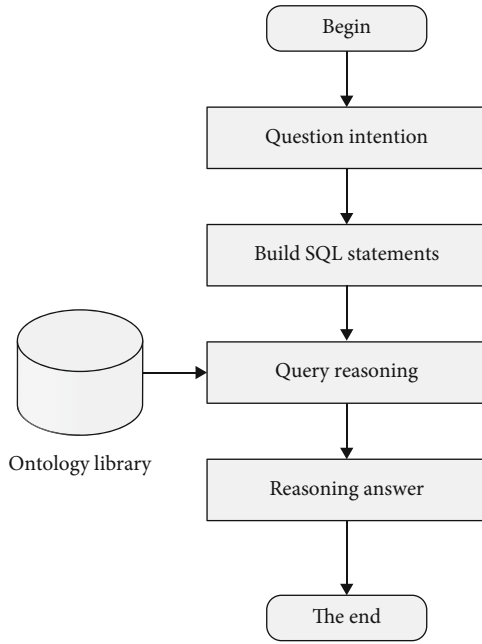

FIGURE 6: Flow chart of the dialogue.

FIGURE 7: Flow chart of the deep retrieval module.

4. Construction of the e-Commerce Online Intelligent Customer Service System Based on Fuzzy Control

This paper chooses a retrieval model to construct a customer service chat robot, preprocesses the original manual customer service history, and deals with several problems in the original corpus. The question answering library constructed in this article has the characteristics of alternating customer speeches and manual customer service speeches. Moreover, the question answering pairs in the question answering library are in the form of customer questions and customer service answers. In the subsequent model construction, this article will select the question answering library as the training corpus and extract all the customer service answers from the question answering library.

This paper uses the structure of the contextual deep retrieval model of the hierarchical network, and the process of processing customer service dialogue is shown in Figure 3. In the following content of this article, the sentence encoder and the context encoder will be constructed separately.

The construction process of the intention vector in the dialogue is shown in Figure 4.

In the customer service chat robot system, firstly, the user's consultation intention expressed by the ontology is obtained through the semantic understanding of the customer's question. After that, the system applies the inquiry intent to the ontology library query and reasoning to generate corresponding answers and obtains the answers from the query and reasoning and the customer service answers retrieved from the human customer service history dialogue records. Finally, the system selects the answer back to the user based on the similarity of the retrieved answers. The customer service chat robot is a system that can respond to

TABLE 1: Statistical table of data retrieval performance evaluation of the intelligent customer service system.

No.	Data retrieval performance	No.	Data retrieval performance	No.	Data retrieval performance
1	96.6	28	93.3	55	96.7
2	94.4	29	96.0	56	94.1
3	94.1	30	93.4	57	93.9
4	97.9	31	96.5	58	96.3
5	95.8	32	96.2	59	93.7
6	97.6	33	97.7	60	94.3
7	95.2	34	94.1	61	96.9
8	93.5	35	97.9	62	96.7
9	96.8	36	93.2	63	97.5
10	97.5	37	97.3	64	97.5
11	93.3	38	97.3	65	94.2
12	97.4	39	94.2	66	95.6
13	96.3	40	96.6	67	96.9
14	93.6	41	95.4	68	97.0
15	95.3	42	98.0	69	94.0
16	93.3	43	94.1	70	97.0
17	97.7	44	96.2	71	94.0
18	95.7	45	95.9	72	95.3
19	95.3	46	93.8	73	93.1
20	93.8	47	95.6	74	94.8
21	94.3	48	95.0	75	93.2
22	93.4	49	95.2	76	96.6
23	97.2	50	94.8	77	96.3
24	97.5	51	97.7	78	95.7
25	96.2	52	94.7	79	97.2
26	95.2	53	94.0	80	93.7
27	94.4	54	97.4	81	94.9

the questions raised by the customer based on the knowledge learned in advance. The overall architecture diagram of the customer service chat robot system is shown in Figure 5.

In the customer service chat robot system, the intelligent reply function is the core of the entire system. The system accepts the question raised by the user and returns an appropriate response based on the question. The processing process of the system after the customer raises a question is shown in Figure 6. In the process of answering a user, the system uses ontology library query and reasoning and question answering database retrieval to obtain answers and determines the returned answer by judging whether the similarity of the answers obtained by the retrieval exceeds a threshold.

Figure 7 shows the process of calling the ontology library query and reasoning module. The process of calling the ontology library for query and reasoning is relatively simple. After the SPARQL statement is constructed by the user's intention, Jena API can be directly called to query the ontology library to obtain the query and reasoning answer aO. Generally, if the semantics of the question is correctly
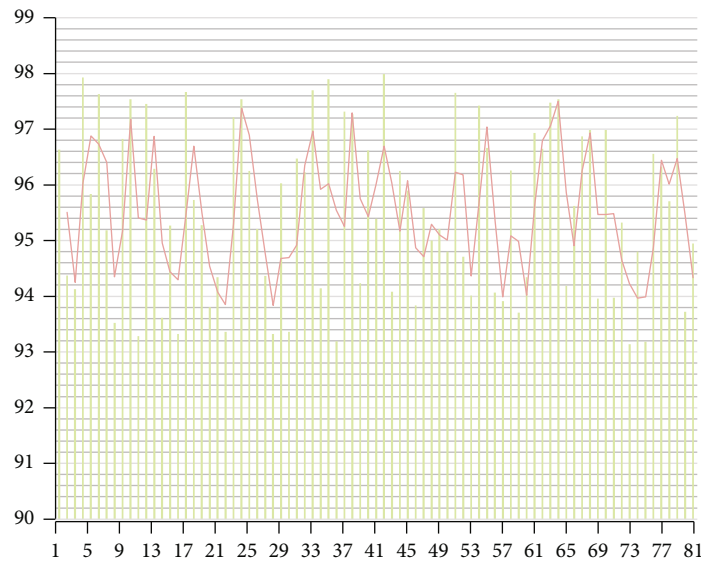
Figure 8: Statistical diagram of data retrieval performance evaluation of the intelligent customer service system.

Table 2: Statistical table of product recommendation performance evaluation of the intelligent customer service system.

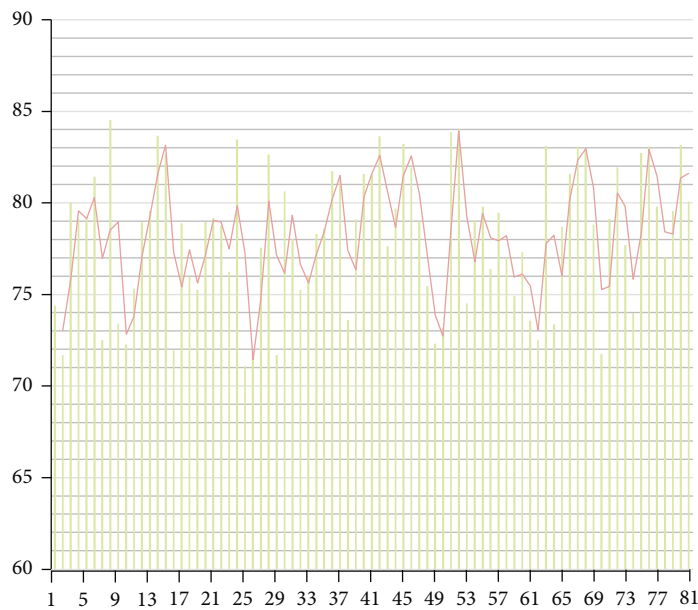| No.	Product recommendation performance	No.	Product recommendation performance	No.	Product recommendation performance
1	74.4	28	82.6	55	79.8
2	71.7	29	71.7	56	76.4
3	80.0	30	80.6	57	79.5
4	79.1	31	78.0	58	77.0
5	79.1	32	75.2	59	74.9
6	81.4	33	76.0	60	77.3
7	72.5	34	78.3	61	73.6
8	84.5	35	78.6	62	72.5
9	73.4	36	81.7	63	83.1
10	72.3	37	81.2	64	73.3
11	75.3	38	73.6	65	78.7
12	78.9	39	79.1	66	81.6
13	79.6	40	81.6	67	83.1
14	83.7	41	81.5	68	82.8
15	82.7	42	83.6	69	78.8
16	71.9	43	77.6	70	71.7
17	78.9	44	79.7	71	79.1
18	76.0	45	83.2	72	81.9
19	75.2	46	81.9	73	77.7
20	79.0	47	79.0	74	74.0
21	79.2	48	75.5	75	82.7
22	78.8	49	72.3	76	83.1
23	76.2	50	73.2	77	79.8
24	83.5	51	83.9	78	77.0
25	71.1	52	84.0	79	79.6
26	71.8	53	74.5	80	83.2
27	77.5	54	79.1	81	80.1

FIGURE 9: Statistical diagram of product recommendation performance evaluation of the intelligent customer service system.

understood, the answer can accurately return the target answer of the user's inquiry. However, when the customer input is not standard or the input content is relatively empty, such as greetings, the ontology library query and reasoning module will return empty.

5. Performance Analysis of the e-Commerce Online Intelligent Customer Service System Based on Fuzzy Control

The e-commerce online intelligent customer service system based on fuzzy control constructed in this paper can answer customers' questions in e-commerce. Therefore, when constructing the e-commerce online intelligent customer service system, this article first answers common questions through the keyword search database, which has certain requirements for the data retrieval of the online customer service system. Secondly, the intelligent customer service system of this article can also recommend the products that users need. Therefore, in the experimental research, this article firstly tests the data retrieval function of the online intelligent customer service system. This article analyzes 81 groups of simulated conversations, and the results are shown in Table 1 and Figure 8.

From the test results, it can be seen that the system constructed in this paper has a good performance in data retrieval. Next, the recommendation function of the e-commerce online intelligent customer service system is tested and analyzed. The results of the test are shown in Table 2 and Figure 9.

Through experimental analysis, it can be known that the e-commerce online intelligent customer service system based on fuzzy control constructed in this article can basically meet the actual needs, so the system constructed in this article can be used in subsequent practice to expand services.

6. Conclusion

With the rapid development of Internet commerce, more and more merchants provide online shopping services, which also bring a lot of online customer service work. At the same time, it has become the choice of more and more merchants to replace manual customer service with customer service chat robots. How to improve the work quality of customer service chatbots has become an issue of ardent concern for merchants. After analyzing the characteristics of customer service domain knowledge, this paper chooses to construct a domain ontology library for structured storage of domain knowledge required by customer service chat robots. Moreover, this article combines the ontology to construct an e-commerce online intelligent customer service system based on the fuzzy control model. In addition, this article starts with sentence vector encoding and proposes an outlier detection algorithm based on the maximum interval squared hinge loss for rejection recognition. Compared with the current baseline method, the system proposed in this paper has achieved performance improvement on the experimental dataset. It can be seen from the experimental results that the intelligent customer service system constructed in this paper has a certain effect.

Data Availability

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

Conflicts of Interest

The author declares no competing interests.

Acknowledgments

References

[1] A. A. al-Musawi, A. A. H. Alwanas, S. Q. Salih, Z. H. Ali, M. T. Tran, and Z. M. Yaseen, "Shear strength of SFRCB without stirrups simulation: implementation of hybrid artificial intelligence model," *Engineering with Computers*, vol. 36, no. 1, pp. 1–11, 2020.

[2] H. Lu, Y. Li, M. Chen, H. Kim, and S. Serikawa, "Brain intelligence: go beyond artificial intelligence," *Mobile Networks and Applications*, vol. 23, no. 2, pp. 368–375, 2018.

[3] R. Ferrari, C. Mancini-Terracciano, C. Voena et al., "MR-based artificial intelligence model to assess response to therapy in locally advanced rectal cancer," *European Journal of Radiology*, vol. 118, no. 4, pp. 1–9, 2019.

[4] X. Sun, J. Young, J. H. Liu, and D. Newman, "Prediction of pork loin quality using online computer vision system and artificial intelligence model," *Meat Science*, vol. 140, no. 7, pp. 72–77, 2018.

[5] V. Nourani, A. Hosseini Baghanam, J. Adamowski, and O. Kisi, "Applications of hybrid wavelet-artificial intelligence models in hydrology: a review," *Journal of Hydrology*, vol. 514, no. 9, pp. 358–377, 2014.

[6] X. N. Bui, H. Nguyen, Y. Choi, T. Nguyen-Thoi, J. Zhou, and J. Dou, "Prediction of slope failure in open-pit mines using a novel hybrid artificial intelligence model based on decision tree and evolution algorithm," *Scientific Reports*, vol. 10, no. 1, pp. 1–17, 2020.

[7] Z. M. Yaseen, A. el-shafie, O. Jaafar, H. A. Afan, and K. N. Sayl, "Artificial intelligence based models for stream-flow forecasting: 2000-2015," *Journal of Hydrology*, vol. 530, no. 7, pp. 829–844, 2015.

[8] J. E. Laird, C. Lebiere, and P. S. Rosenbloom, "A standard model of the mind: toward a common computational framework across artificial intelligence, cognitive science, neuroscience, and robotics," *AI Magazine*, vol. 38, no. 4, pp. 13–26, 2017.

[9] K. Chapi, V. P. Singh, A. Shirzadi et al., "A novel hybrid artificial intelligence approach for flood susceptibility assessment," *Environmental Modelling & Software*, vol. 95, no. 1, pp. 229–245, 2017.

[10] M. R. Hashemi, M. L. Spaulding, A. Shaw, H. Farhadi, and M. Lewis, "An efficient artificial intelligence model for prediction of tropical storm surge," *Natural Hazards*, vol. 82, no. 1, pp. 471–491, 2016.

[11] T. Sustrova, "A suitable artificial intelligence model for inventory level optimization," *Trends Economics and Management*, vol. 10, no. 25, pp. 48–55, 2016.

[12] B. T. Pham, M. D. Nguyen, D. Van Dao et al., "Development of artificial intelligence models for the prediction of compression coefficient of soil: an application of Monte Carlo sensitivity analysis," *Science of the Total Environment*, vol. 679, no. 8, pp. 172–184, 2019.

[13] A. Enshaei, C. N. Robson, and R. J. Edmondson, "Artificial intelligence systems as prognostic and predictive tools in ovarian cancer," *Annals of Surgical Oncology*, vol. 22, no. 12, pp. 3970–3975, 2015.

[14] J. S. Chou and D. K. Bui, "Modeling heating and cooling loads by artificial intelligence for energy-efficient building design," *Energy and Buildings*, vol. 82, no. 6, pp. 437–446, 2014.

[15] Z. Ghahramani, "Probabilistic machine learning and artificial intelligence," *Nature*, vol. 521, no. 7553, pp. 452–459, 2015.

[16] A. Zappone, M. di Renzo, M. Debbah, T. T. Lam, and X. Qian, "Model-aided wireless artificial intelligence: embedding expert knowledge in deep neural networks for wireless system optimization," *IEEE Vehicular Technology Magazine*, vol. 14, no. 3, pp. 60–69, 2019.