International Journal for Housing Science and Its Applications
Publish August 4, 2025. Volume 46, Issue 3 Pages 2164-2179

https://doi.org/10.70517/ijhsa463172

Research on English Grammar Structure Analysis Model Based on Unsupervised Learning Algorithm and Automated Grammar Teaching Approach

Limin Zhang[1,*]
[1] School of International Studies, Luoyang Institute of Science and Technology, Luoyang, Henan, 471023, China
Corresponding authors: (e-mail: lilyzhangww@163.com).

Abstract This paper starts from the unsupervised learning algorithm and introduces it into the field of English teaching to analyze English grammatical structure in depth. The similarity of English texts is calculated using dependent syntactic analysis, and the English grammar structure analysis model (HHMM) based on unsupervised learning is constructed through the methods of plain Bayes, Hidden Markov Model, and Hierarchical Markov Model. The model of automated classification and sharing of English grammar teaching resources is constructed, and the automatic scoring module of online examination of the system is designed using graph attention network. The performance and usability of the HHMM grammar structure analysis model and the automated English grammar teaching system are evaluated respectively. The analysis indexes of this paper's HHMM grammar structure analysis model in all experimental datasets are optimal results, and its grammar analysis performance far exceeds that of other models. The overall score for the usability of the automated grammar teaching system in this paper is 4.21. The scores of the four first-level indicators are all over 4 or more. The range of scores in the secondary metrics is [4.02,4.31]. The range of scores in tertiary indicators is [3.97,4.54], and only two tertiary indicators are below 4. It can be seen that the automated grammar teaching system in this paper has obtained good evaluation results among the student population.

Index Terms unsupervised learning, HHMM model, graph attention network, grammar structure, automated teaching

I. Introduction

With the development of cross-cultural communication and economic globalization, the status of English as a language in the world remains firm. The study of grammar is an important and indispensable node for the mastery of English skills. Grammar mainly focuses on words and sentences, the lexical nature of words, their meanings, and their general application in sentences are one of the basic contents of grammar, while the constituents of sentences, changes in meaning and so on fully embody the basic contents of the whole grammar, such as structure, change and connection [1]-[3]. Grammar learning in the traditional sense is actually the learning of structures and concepts, and only by mastering these basic and relatively stable linguistic elements and practical rules can we build up a framework for English learning and form a relatively clear application model and practical language application and expression of ideas [4]-[7]. However, there are various problems in the current grammar teaching. First, most teachers fail to design teaching from the students' perspective, and still follow the traditional and monotonous teaching mode and methods, such as duck-filling teaching, neglecting the cultivation of practical operation and practical ability, students' diversified needs are difficult to be met, and students lack the joy of learning, which leads to the difficulty of enhancing the students' practical language application ability [8]-[11]. Teachers are the center and the main body, students lack the initiative and have no vitality in the classroom, and the teaching effect is half the result with twice the effort. Secondly, system theory emphasizes that the function of a well-structured system of things far exceeds the simple addition of the components, and if there is a lack of comprehensive understanding of the whole, the knowledge of the parts will inevitably appear to be one-sided and lack of connection [12]. In view of this, in the process of grammar teaching, teachers need to integrate all kinds of grammatical phenomena encountered by students, and gradually transition from the local to the whole, in order to build a complete knowledge system. Although the current textbooks follow the principle of gradual in-depth knowledge of language, to some extent, the systematic and coherent structure of grammatical knowledge should be ignored, such as the determiner clauses are scattered in a number of units, if the teacher does not systematically summarize, it is difficult for students to clarify the fundamental differences between the relational pronouns and the relational adverbs, resulting in ambiguity in the results of the learning [13]-[16]. Thirdly, in all kinds of English teaching observation class, quality class demonstration, almost difficult to find the figure of grammar class, grammar teaching has become a “neglected corner” in the reform process [17]. In view of the current actual situation of grammar teaching, analyzing grammar structure, improving grammar teaching strategies and relieving students' learning pressure have become urgent problems.

In this paper, we calculate the text similarity in English sentences through dependent syntactic analysis, and then construct an unsupervised learning-based English grammar structure analysis model (HHMM) by using plain Bayes, Hidden Markov Model, and Hierarchical Markov Model. In order to further promote the development of grammar teaching, this paper constructs an automated classification and sharing model of grammar teaching resources encompassing the campus network, the Internet, and the inter-campus network. Then, we design the automatic scoring module of online English grammar test through graph attention network, so as to complete the construction of automated grammar teaching system. The performance of the HHMM grammar structure analysis model in this paper is compared with other models. The effect of using the automated grammar teaching system proposed in this paper is explored to explore the usability of the system.

II. Unsupervised learning-based model for analyzing English grammatical structures

II. A.English Text Similarity Calculation Based on Dependency Syntactic Analysis

Sentence similarity computation can generally be categorized into three levels: syntactic, semantic and pragmatic [18]. In the field of text parsing, the purpose of calculating sentence similarity is to extract the common part as a template from multiple records with similar structure, the nature of which determines that our focus is on the syntactic level, and the similarity is affected by sentence structure and lexicality. The current universal similarity calculation formula is as follows:

$$simSeq = \frac{\sum_{i=1}^{n} equ(seq_1(i), seq_2(i))}{n} \tag{1}$$

where seq_1 and seq_2 represent the log records and log templates to be matched, respectively, $seq(i)$ represents the i th word in the sentence, and n represents the length of the sentence, and for equ is defined as follows:

$$equ(t_1, t_2) = \begin{cases} 1, & if\ \ t_1 = t_2 \\ 0, & otherwise \end{cases} \tag{2}$$

Equation (2) compares two words to see if they are equal, if they are equal, the value of equ is 1, otherwise it is 0.

This calculation method treats all words in a sentence as equally important, but in text data, words in certain positions may be variables whose changes do not significantly affect the structure of the sentence and the template extraction results. Therefore, when designing the similarity calculation model, the contribution of key information in the sentence should be fully considered, and the interference of secondary variables should be avoided by distinguishing and labeling the importance of words, so as to improve the accuracy and reliability of similarity calculation. Based on the above analysis, this section proposes a similarity calculation model for English text based on dependent syntax analysis.

Dependency syntax is a grammatical form used to describe the structure of a sentence [19], which focuses on the dependency relationship between the components of a sentence, i.e., the grammatical relationship between the center word and its modifying components. By analyzing the dependency relationship between each word and the central word, Dependency Syntax can effectively extract the main structure of the sentence, i.e., the most important components of the sentence and the relationship between them. For the convenience of description, the following two definitions are given in this section:

Definition 1 (Key Words): In a dependent syntax tree, the words contained in the root node and the nodes directly connected to the root node are the key words of the whole sentence, labeled as set K (Key Words). Definition 2 (Auxiliary Words): The rest of the sentence is labeled as Auxiliary Words, except for the words in set K. The words in set A are labeled as Auxiliary Words.

This chapter uses spaCy as a dependency syntax analyzer to perform dependency analysis on example sentences. spaCy is an open-source natural language processing library that includes functionality for dependency syntax analysis. With dependent syntax analysis, spaCy can identify and represent the grammatical relations between words in a sentence, such as subject-predicate relations and modification relations, as well as the syntactic structure of words in a sentence. This analysis helps to understand the syntactic structure of sentences, thus supporting natural language processing tasks such as lexical disambiguation, information extraction and relation extraction. The results of the analysis are shown in Figure 1.

In Figure 1, each node represents a word, and at the bottom is the lexical label of the word, commonly, VERB is verb, NOUN is noun, ADP is take this, and NUM is base word. The node with an entry of 0 is the root node, which is the core word of the sentence, usually the verb in the main predicate relation, in this case "Received".

The nodes connected by arrows indicate the existence of dependencies between words, and the labels on the arrows indicate the types of dependencies, for example, as they appear in Figure 1, dobj (direct object) denotes an undirected object, prep (prepositonal modifier) denotes an unprepositioned phrase modifier, pobj (object of preposition) denotes prepositional object, compound denotes compound noun modifier, which is used to describe the modification relationship between individual words in a compound noun consisting of more than one word, and mmimod denotes quantitative modifier, which is used to describe the modification relationship of a number or quantifier to a noun.

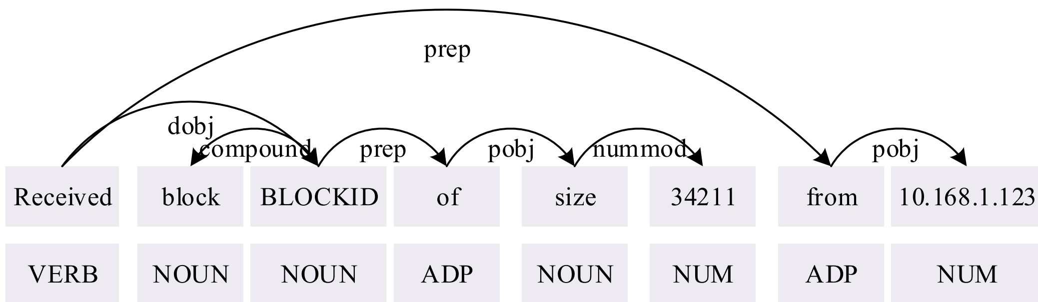

Figure 1: Dependency analysis process

The transformation of Figure 1 into a semantic dependency tree is shown in Figure 2.

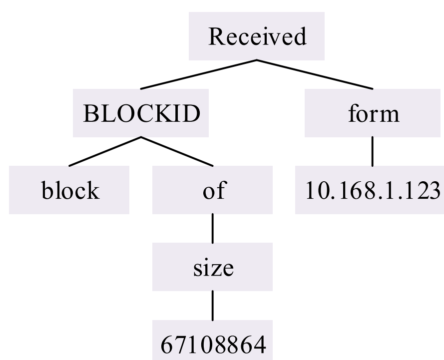

Figure 2: Semantic dependent tree

Finally, define the similarity as:

$$simSeq(seq_1, seq_2) = \frac{simk * \alpha + sima * \beta}{\|K_1\| + \|K_2\| * \alpha + \|A_1\| + \|A_2\| * \beta} \tag{3}$$

where K_i and A_i are the set of key words and the set of auxiliary words in seq_i, respectively, $\|K_1\| + \|K_2\|$ represents the total number of key words in the two sentences, $\|A_1\| + \|A_2\|$ represents the total number of auxiliary words in the two sentences, and α is the weight coefficient of key words $(0.5 \le \alpha \le 1)$, β is the weight coefficient of auxiliary words $(\beta = 1 - \alpha)$, and $simk$ and $sima$ denote the similarity of the key words and auxiliary words, respectively, which are calculated as follows:

$$simk = \sum_{i=1}^{n} eqk(seq_1(i), seq_2(i)) \tag{4}$$

$$sima = \sum_{i=1}^{n} ega(seq_1(i), seq_2(i)) \tag{5}$$

$$eqk(t_1,t_2) = \begin{cases} 2, & if\ t_1 = t_2\ and\ t_1 \in K_1\ and\ t_2 \in K_2 \\ 1, & if\ t_1 = t_2\ and\ (t_1 \in K_1\ or\ t_2 \in K_2) \\ 0, & otherwise \end{cases} \tag{6}$$

$$ega(t_1,t_2) = \begin{cases} 2, & if\ t_1 = t_2\ and\ t_1 \in A_1\ and\ t_2 \in A_2 \\ 1, & if\ t_1 = t_2\ and\ (t_1 \in A_1\ or\ t_2 \in A_2) \\ 0, & otherwise \end{cases} \tag{7}$$

where $eqk\left(t_1,t_2\right)$ denotes 2 if the two words to be tested are the same and both of them belong to the keyword, and 1 if the two words to be tested are the same but only one of them is the keyword, and 0 otherwise. $simk$ denotes the number of the keywords that have the same keyword at the corresponding position of the two sentences. Similarly, $ega\left(t_1,t_2\right)$ is used to compare whether the auxiliary words in the same position are the same or not and count the number of the same ones, and $simk$ denotes the number of auxiliary words in the corresponding positions of the two sentences that are the same.

II. B.Simple Bayesian-based analytical modeling

II. B. 1) Formal Representation of Model-Base

Let $\bar{U}^2$ be a sequence of speech acts of an English text, i.e., $\bar{U} = \left\langle u_1, \cdots; u_N \right\rangle$, where $u_i \in U$ is any speech act and U is the set of speech acts. Let also A be the set of structural tokens (speech act tokens or structural tokens) of speech acts, then the task of speech structure recognition is to establish a mapping relation $h: U \to A$ between U and A. According to Bayesian decision theory [20], the function h can be expressed as:

$$h(u) = \arg\max_{a \in A} P(a \mid u) \tag{8}$$

Since $P(a \mid u) = \dfrac{P(u \mid a)P(a)}{P(u)}$, where $P(u)$ is independent of the choice of a, the above equation can be reduced to:

$$h(u) = \arg\max_{a \in A} P(a \mid u) = \arg\max_{a \in A} P(u \mid a)P(a) \tag{9}$$

Consider $F = \left\langle f_1, \cdots, f_M \right\rangle$ is the set of features of u, where M is the number of elements in the feature set F. The characteristic function f is a schematic function on U, i.e:

$$f(u) = \begin{cases} 1 & \text{When feature f is present in } u \\ 0 & \text{Otherwise} \end{cases} \tag{10}$$

Then the characteristic representation of the function h is:

$$h(u) = \arg\max_{a \in A} P\left(\left\langle f_1(u), \cdots, f_M(u) \right\rangle \mid a\right) P(a) \tag{11}$$

In practice, it is assumed that there is no correlation between the individual features, i.e., the above equation is further simplified using a plain Bayesian model:

$$h(u) = \arg\max_{a \in A} \prod_{f_i \in F} P(f_i(u) \mid a) P(a) \tag{12}$$

II. B. 2) Feature Extraction for Model-Base

The method of feature selection can be done by Information Gain (IG), Mutual Information (MI) and so on. In our study, we choose IG as the feature selection method, which can be represented in Model-Base as:

$$G(f) = H(A) - H(A \mid f) \tag{13}$$

As:

$$H(A)=-\sum_{i=1}^{\#(A)}P(a_i)\log P(a_i) \tag{14}$$

$$\begin{aligned}H\left(A\mid f\right)=&-P(f)\sum_{i=1}^{\#(A)}P(a_i\mid f)\log P(a_i\mid f)\\&-P(\bar{f})\sum_{i=1}^{\#(A)}P\left(a_i\mid\bar{f}\right)\log P\left(a_i\mid\bar{f}\right)\end{aligned} \tag{15}$$

So:

$$\begin{aligned}G(f)=&-\sum_{i=1}^{\#(A)}P(a_i)\log P(a_i)\\&+P(f)\sum_{i=1}^{\#(A)}P\left(a_i\mid f\right)\log P\left(a_i\mid f\right)\\&+P(\bar{f})\sum_{i=1}^{\#(A)}P\left(a_i\mid\bar{f}\right)\log P\left(a_i\mid\bar{f}\right)\end{aligned} \tag{16}$$

where $\bar{f}$ denotes the case $f=0$ and $\#(A)$ denotes the number of elements in the set A.

II. C.HMM-based analytical modeling

II. C. 1) Temporal correlation of structures

The use of mutual information as an analytical tool for temporal correlation is there:

$$MI_d(T1,T2)=\sum P_d(T1,T2)\log\frac{P_d(T1,T2)}{P(T1)P(T2)} \tag{17}$$

where $T1$ and $T2$ are both structural tokens (speech act tokens/structures), $P_d(T1,T2)$ denotes the probability of co-occurrence of $T1$ and $T2$ at time intervals of d and $P(T1)$ and $P(T2)$ denote the probability of occurrence of $T1$, $T2$, respectively.

II. C. 2) Hidden Markov Models (HMM)

A formal description of the Hidden Markov Model HMM [21]: $M=\left(S,O,A,B,\pi\right)$.

(1) S denotes the states (i.e., outputs) in the model, and its number of states is N. For some practical applications, each state of the model is associated with some physical meaning that can be transferred from one state to another. All independent states are defined as $S=\left\{S_1S_2\cdots S_N\right\}$, and q_t is used to denote the state at moment t.

(2) O denotes the observations for each state, and the number of possible observations corresponding to each state is M. The observations correspond to the actual outputs of the model system, and we denote these observations as $W=\left\{w_1w_2\cdots w_M\right\}$.

(3) State transfer probability matrix $A=\{a_{ij}\}$, where $a_{ij}=P\left(q_{t+1}=S_j\mid q_t=S_i\right)$, $1\le i$, $j\le N$. a_{ij} denotes the probability of moving from state i to state j, a_{ij} satisfies: $a_{ij}\ge 0,\forall i,j$. and $\sum_j a_{ij}=1,\forall i$.

(4) Output the observation probability distribution matrix $B=\{b_j(k)\}$, where $b_j(k)$ denotes the probability of w_k occurring at moment t in state S_j, i.e., $b_j(k)=P\left(\text{Observed at moment } t\ w_k\mid q_t=S_j\right)$, $1\le j\le N$, $1\le k\le M$. $b_j(k)$ satisfies: $b_j(k)\ge 0,\forall j,k$. and $\sum_k b_j(k)=1,\forall j$.

(5) The initial state distribution vector $\pi=\{\pi_i\}$, where $\pi_i=P(q_1=S_i)$, and $1\le i\le N$, i.e., the probability of being in the state S_i at the moment $t=1$. Satisfies: $\sum_i \pi_i=1$.

For convenience, a $\lambda=\left(A,B,\pi\right)$ can be used to represent the HMM model parameters.

II. C. 3) Formal Representation of Model-I

Let $\bar{U}$ be the sequence of speech acts corresponding to an English text, i.e., $\bar{U}=\left\langle u_1,\cdots;u_N\right\rangle$, where $u_i\in U$ is any speech act, U is the set of speech acts, N is the length of the sequence of speech acts. Length of the sequence of speech acts. Let $\bar{A}$ be the sequence of shallow tokens corresponding to $\bar{U}$, i.e., $\bar{A}=\left\langle a_1,\cdots,a_N\right\rangle$ where $a_i\in A$ is the $u_i\in U$ of the speech act category, and A is the set of categories of grammatical structure markers. Then the speech act recognition task is to establish a mapping relation $h:\bar{U}\to\bar{A}$ between $\bar{U}$ and $\bar{A}$ such that the function h can be expressed as:

$$h(\bar{U})=\arg\max_{\bar{A}} P(\bar{A}\mid\bar{U}) \tag{18}$$

Since $P(\bar{A}\mid\bar{U})=\frac{P(\bar{U}\mid\bar{A})P(\bar{A})}{P(\bar{U})}$, where $P(\bar{U})$ and $\bar{A}$ are chosen independently, so the above equation can be reduced to:

$$h(\bar{U})=\arg\max_{\bar{A}} P(\bar{A}\mid\bar{U})=\arg\max_{\bar{A}} P(\bar{U}\mid\bar{A})P(\bar{A}) \tag{19}$$

For the sake of completeness of presentation, we give below the expansion of the above equation under the order condition:

$$h(\bar{U})=\arg\max_{\bar{A}}\prod_i P(u_i\mid a_i)P(a_i) \tag{20}$$

When $P(u_i\mid a_i)$ is expanded in the feature space according to the independence that is $P(u_i\mid a_i)=\prod_{f_l\in F}P(f_l(u_i)\mid a_i)$, so (20) is expanded as $h(\bar{U})=\arg\max_{\bar{A}}\prod_i\prod_{f_l\in F}P(f_l(u_i)\mid a_i)P(a_i)$.

II. D. Analytical model based on HHMM

II. D. 1) Hierarchical Hidden Markov Models (HHMM)

A formal description of the Hierarchical Hidden Markov Model HHMM [22]: $M=\langle\Sigma,O,A,B,\Pi,D\rangle$, where: Σ is the finite set of states. O is the finite set of observations. Σ^* denotes the set of all possible strings on Σ. An observation sequence is a finite string on Σ^*, denoted as: $\bar{O}=o_1o_2\cdots o_T$. A is the state transfer matrix. B is the state-to-observation probability matrix. Π is the initial state distribution. D is the depth of M. A state of HHMM is represented as $q_i^d\ (d\in\{1,\cdots,D\})$, where i is the state index and d is the hierarchical index. The hierarchical index of the root is 1 and the index of the emitting state is D. Internal states need not have the same number of substates, so $|q_i^d|$ is used to denote the number of substates of the internal state q_i^d. In cases where the contextual semantics are clear, the state index can be ignored and q^d is used to denote a state on level d. In addition to the model structure, an HHMM is represented by a vector of state transfer probabilities between internal states and output distributions of emitted states. Each internal state q_i^d ($d\in\{1,\cdots,D\}$) has a state transfer probability matrix $A^{q^d}=\left(a_{ij}^{q^d}\right)$ where $a_{ij}^{q^d}=P\left(q_j^{d+1}\mid q_i^{d+1}\right)$ is the horizontal transfer probability from the i th sub-state to the j th sub-state of state q^d. Similarly, $\Pi^{q^d}=\left\{\pi^{q^d}(q_i^{d+1})\right\}=\left\{P(q_i^{d+1}\mid q^d)\right\}$ is the initial distribution vector of the q^d substates that denotes the probability that state q^d initially activates state q_i^{d+1}. If q_i^{d+1} is also an internal state, then $\pi^d\left(q_i^{d+1}\right)$ can be interpreted as a vertical transfer probability: the probability of moving from the parent state q^d into the child state q_i^{d+1}. Each emitting state q^D is uniquely defined by its output probability vector $B^{q^D}=\left\{b^{q^D}(k)\right\}$, where $b^{q^D}(k)=P\left(\sigma_k\mid q^D\right)$ is the probability that the emitting state q^D outputs the symbol $\sigma_k\in\Sigma$. The complete parameter set is denoted as follows:

$$\lambda=\left\{\lambda^{q^d}\right\}_{d\in\{1,\cdots,D\}}=\left\{\left\{A^{q^d}\right\}_{d\in\{1,\cdots,D-1\}},\left\{\Pi^{q^d}\right\}_{d\in\{1,\cdots,D-1\}},\left\{B^{q^D}\right\}\right\} \tag{21}$$

The generation of a string can be described as such a top-to-bottom depth-first (DF) process $\overline{Step}$ as shown in Figure 3. Starting from the root state, a sub-state of the root state is then randomly selected according to Π^{q^1}. Similarly, for each entered internal state q, a sub-state of q is randomly selected according to the initial probability vector Π^q of q. The operation proceeds recursively until a firing state q^D is reached, at which point a unique symbol is generated based on a state output probability vector B^{q^D}. Control then returns to the activated state q^D. When a recursive string generation process is complete, the internal state that started the recursive process selects the next state in the same layer based on the state transfer matrix for that layer, and the newly selected state starts a new recursive string generation process. Each layer has a terminal state, denoted as q_{end}^{d}, which terminates the random state activation process. When a terminal state is reached, control returns to the parent state of the entire layer. When the recursive activation control returns to the root state, the generation of the observation sequence is complete.

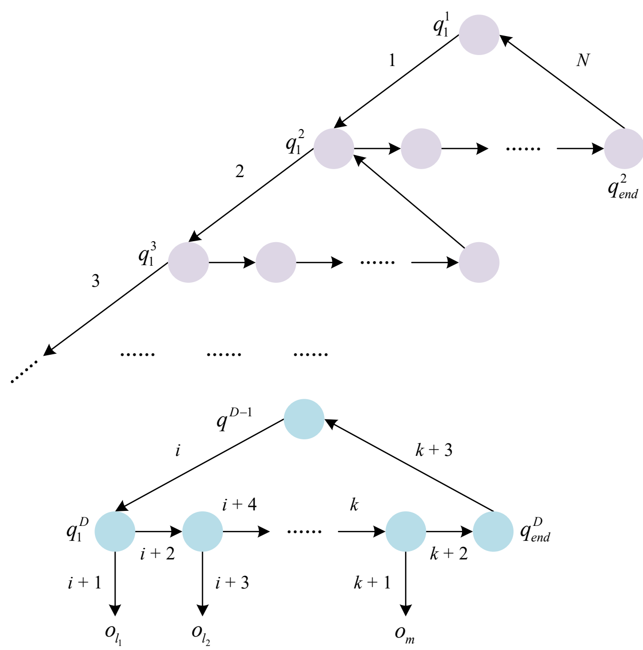

Figure 3: HHMM string generation process

Description:

(1) Generation is performed in a top-to-bottom depth-first order, and the generation probability of each step is $P\left(Step_i\right)$, which corresponds to the corresponding values of A, B, and Π, where i denotes the ordinal number of the generation step.

(2) The launching state q^D launching the visible character o is also considered as a generation step.

Then the generation probability of $\overline{Step}$ is:

$$P\left(\overline{Step}\right)=\prod_{i} P\left(Step_i\right) \tag{22}$$

Since $\overline{Step}$ is the HHMM unfolded in depth-first order, also notated as $\overline{Step_{DF}}$, it can be shown that $P\left(\overline{Step}\right)=P\left(\overline{Step_{DF}}\right)$. Since the difference between $\overline{Step_{BF}}$ (width-first unfolding) and $\overline{Step}$ is only in the order in which $Step_i$ is arranged, it follows:

$$P\left(\overline{Step_{BF}}\right)=P\left(\overline{Step_{DF}}\right) \tag{23}$$

The main differences between HHMM and HMM are:

① The representation of a state in Σ is: $q_i^d\left(d\in\{1,\cdots,D\}\right)$ where i is the number of the state in the current layer of HMM states. d is the hierarchical depth of the state in M, and all the states form a tree structure with a depth of $D-1$, where the depth of the root is 1 and the depth of the deepest leaf is D, and the states with $d<D$ we call internal states.

② Each internal state $q^d\left(d\in\{1,\cdots,D-1\}\right)$ exists a sub-state, the number of sub-states is recorded as $\left|q^d\right|$, and all sub-states constitute a Hidden Markov Chain, the state outputs of the $d+1$ layer can be regarded as a sequence of states of the HMM of that layer, and the state transfer matrix of that layer of HMM is. The state transfer matrix of the layer HMM is:

$$A(q^d)=(a_{ij}(q^d)) \tag{24}$$

where $a_{ij}(q^d)=P\left(q_j^{d+1} \mid q_i^{d+1},q^d\right)$.

Meanwhile, the initial distribution of each state is:

$$\prod(q^d)=\pi^d(q_i^{d+1})=P(q_i^{d+1} \mid q^d) \tag{25}$$

Its physical meaning can be understood as the probability of activation of some internal state q^d of the d th layer HMM to the $d+1$ th layer HMM.

③ In each layer of the HMM, only the D th layer has a truly observable terminator, i.e., the state-to-observation output probability:

$$B(q^D)=\left(b_k(q^D)\right) \tag{26}$$

where $b_k(q^D)=P\left(o_k \mid q^D\right)$, and o_k is an observation that belongs to a finite set of terminators.

Therefore, the set of parameters of HHMM can be expressed as:

$$\begin{aligned}\lambda&=\left\{(\lambda\left(q^d\right)\right\}_{d\in\{1,\ldots,l)}\\&=\left\{\left\{A\left(q^d\right)\right\}_{d\in\{1,\ldots,D-1\}},\left\{\prod\left(q^d\right)\right\}_{d\in\{1,\ldots,D-1\}},\left\{B\left(q^D\right)\right\}\right\}\end{aligned} \tag{27}$$

II. D. 2) Formal representation of Model-II

Let the English text correspond to a sequence of speech acts $\overline{U}=\langle u_1,\cdots;u_N\rangle$, where $u_i\in U$ is any speech act, U is the set of speech acts, and N is the length of the speech act sequence. Let $T_i^l\left(1\le l\le 3,1\le i\le N_l\right)$ be an internal node of HHMM, where N_l denotes the maximum index of the internal node at the l th level:

$$\begin{aligned}&\underset{T_{mama}}{\arg\max}\,P\left(\overline{T_{HHMM}\mid\overline{u_1\cdots u_{N_3}}}\right)\\&=\underset{T_{mama}}{\arg\max}\frac{P\left(\overline{u_1\cdots u_{N_3}}\overline{T_1^3\cdots T_{N_3}^3},\overline{T_1^2\cdots T_{N_2}^2}T_1^1\right)}{P\left(\overline{u_1\cdots u_{N_3}}\right)}\\&=\underset{T_{maaa}}{\arg\max}\,P\left(\overline{u_1\cdots u_{N_3}},\overline{T_1^3\cdots T_{N_3}^3}\overline{T_1^2\cdots T_{N_2}^2}T_1^1\right)\end{aligned} \tag{28}$$

where T_{HHMM} denotes the internal state of the HHMM.

Expanding $P\left(\overline{u_1\cdots u_{N_3}},\overline{T_1^3\cdots T_{N_3}^3},\overline{T_1^2\cdots T_{N_2}^2}T_1^1\right)$ by a first order markov expansion in the direction of the depth of the HHMM then yields:

$$\begin{aligned}&P\left(\overline{u_1\cdots u_{N_3}},\overline{T_1^3\cdots T_{N_3}^3},\overline{T_1^2\cdots T_{N_2}^2}T_1^1\right)\\&=P\left(\overline{T_1^1}\right)P\left(\overline{T_1^2\cdots T_{N_2}^2}\mid T_1^1\right)P\left(\overline{T_1^3\cdots T_{N_3}^3}\mid\overline{T_1^2\cdots T_{N_2}^2}\right)P\left(\overline{u_1\cdots u_{N_1}}\mid\overline{T_1^3\cdots T_{N_3}^3}\right)\end{aligned} \tag{29}$$

Assume that T_i^l as well as u_i are generated only with respect to their parent nodes:

$$P\left(\overline{T_1^3\cdots T_{N_3}^3}\mid\overline{T_1^2\cdots T_{N_2}^2}\right)=\prod_i P\left(\overline{T_l^3\cdots T_m^3}\mid T_i^2\right) \tag{30}$$

$$P\left(\overline{u_1\cdots u_{N_3}}\mid\overline{T_1^3\cdots T_{N_3}^3}\right)=\prod_i P\left(u_i\mid T_i^3\right) \tag{31}$$

So:

$$P\left(\overline{u_1\cdots u_{N_3}},\overline{T_1^3\cdots T_{N_3}^3},\overline{T_1^2\cdots T_{N_2}^2 T_1^1}\right) = P\left(T_1^1\right)P\left(\overline{T_1^2\cdots T_{N_2}^2} \mid T_1^1\right)\prod_i P\left(\overline{T_l^3\cdots T_m^3} \mid T_i^2\right)\prod_j P\left(u_i \mid T_j^3\right) \tag{32}$$

III. Design of an automated grammar teaching system

III. A. A model for automated categorization and sharing of grammar teaching resources

The English grammar digital teaching resources classification and sharing model can be divided into three main categories, namely, campus network model, Internet model, and inter-school model.

(1) Campus network model

Based on its own disciplinary advantages, the school integrates digital teaching resources, integrates them in a consultative manner, stores the resources in a unified way, manages them in a categorized way, and builds a perfect resource management platform, which grants the corresponding rights according to the needs, and the users are able to access the required resources. The campus network model framework is shown in Figure 4.

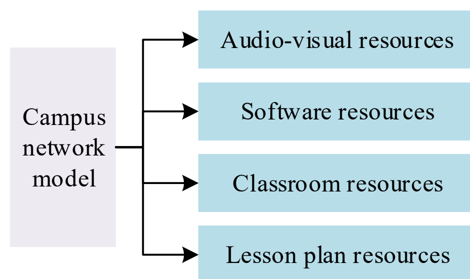

Figure 4: Campus network model framework

(2) Internet Model

The Internet is used to standardize the sharing of digital teaching resources of English grammar to avoid blindness and achieve sharing benefits. The Internet model framework is shown in Figure 5.

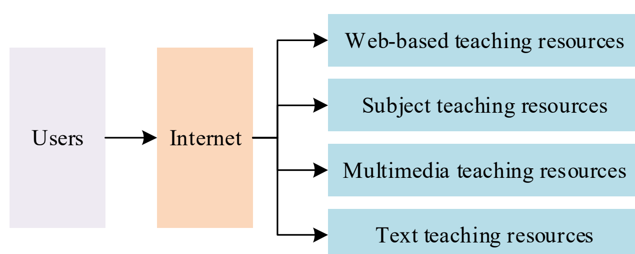

Figure 5: Internet model framework

(3) Intercampus model

The intercampus model belongs to the ultimate model, based on a unified mirror site with remote access to the database in order to share resources. The framework of the intercollegiate model is shown in Figure 6.

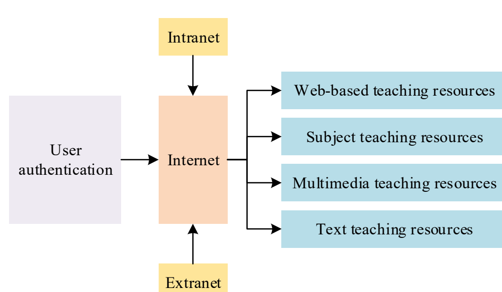

Figure 6: Intercollegiate model framework

III. B. Automatic Grading Module for Online Exams

III. B. 1) Graph Attention Network Fundamentals

Graph Attention Network (GAT) is a new neural network architecture based on graph data structure [23]. It is obtained by introducing the attention mechanism based on graph convolutional neural network. The network uses the attention mechanism to weight the nodes in the graph to achieve accurate capturing of relationships between nodes and extraction of key information of the graph structure.

Before the graph convolutional neural network carries out feature extraction, it needs to linearly transform the feature matrix of the current neighboring nodes, and after that, it carries out mean pooling computation, and aggregates the results to obtain the feature vector of the current node. Different from graph convolutional neural network, graph attention network only needs to obtain the information of neighboring nodes to extract the feature vector. The graph attention network learns the importance of any neighboring entities x_i, x_j in the input data through the attention weight matrix, and determines whether there is a correlation between nodes based on the adjacency matrix. The specific expression of the adjacency matrix is:

$$A = a_{ij} + I_N \tag{33}$$

The updating mechanism of the graph attention neural network can be represented as:

$$h_i^{l+1} = \sigma\left(\sum_{j\in N_i} a_{ij} h_i^l W^l\right) \tag{34}$$

where h_i^{l+1}, h_i^l denote the vector representation of the $l+1$ and l th layer i node, respectively. N_i denotes the set of neighboring nodes of i nodes. a_{ij} is the matrix of attention correlation coefficients between i nodes and j nodes. W^l denotes the parameter matrix of the l th layer. σ denotes the nonlinear activation function.

After obtaining the attention coefficients, the neighbor nodes of the current node i can be weighted and summed, which is calculated as:

$$h_i' = \sigma\left(\sum\nolimits_{j\in N_i} a_{ij} \overrightarrow{Wh_j}\right) \tag{35}$$

$$\vec{h}_i' = \Big\|_{k=1}^{K} \sigma\left(\sum\nolimits_{j\in N_i} a_{ij}^k \vec{W}^k h_j\right) \tag{36}$$

where W denotes the shared weight matrix and k is the number of weight matrices.

III. B. 2) Encoding feature vectors

Abstract Syntax Tree (AST) is a tree-like data structure that allows the derivation of grammatical structures and statements of online test samples, and the nodes can be categorized into the types of variable names, operation symbols, and loop statements. In the automatic scoring model, the nodes of the abstract syntax tree are responsible for the accuracy of the key information, and the key nodes are generated through lexical analysis, syntactic analysis and semantic analysis, and finally converted into the corresponding syntax tree form.

The abstract syntax tree is mainly divided into placeholder nodes, semantic nodes and syntax nodes. Feature vector encoding is mostly realized by one-hot encoding. However, when there are more nodes in the abstract syntax tree, the one-hot coding method will increase the coding time and cost, and is not time-effective. Therefore, a combination of word embedding and positional coding is proposed to convert the abstract syntax tree to vector representation.

(1) The principle of word embedding is the type representation of words, which is an important representation in the field of natural language processing. Word embedding is obtained by optimizing one-hot coding. Since the node sequence of the abstract syntax tree is a string sequence, the feature vector extraction cannot be performed directly using traditional feature extraction methods. Therefore, the node sequences need to be converted into numerical vectors, and the continuous bag-of-words (CBOW) model is used as the word vector generation algorithm. The principle of this model is to predict the next word based on the word context environment.

The CBOW model is mainly composed of input layer, projection layer and output layer. Its workflow is as follows: Firstly, the context word of the word is input in the input layer. Then use the projection of the target word's context word to multiply the weight matrix, the output hidden layer vector, the vector sum and calculate the average value, thus obtaining the projection layer output vector. Finally use softmax normalization to process the vector multiplication weight matrix of the projection layer output, which is the word embedding matrix.

(2) Positional encoding. In order to ensure the effect of vector transformation of abstract grammar tree nodes, it is proposed to add position information based on word embedding to accurately express the position of abstract grammar tree nodes in the expression of exam samples through sine and cosine functions. The two functions can control the distance between each node within 1. The specific expression is:

$$PE_{(pos,2i)} = \sin\left(pos / 1000^{2i/d}\right) \tag{37}$$

$$PE_{(pos,2i+1)} = \cos\left(pos / 1000^{2i/d}\right) \tag{38}$$

where pos represents the location information. i represents the dimension.

After summing the word embedding results and the positional coding information, the vector representation of the abstract syntax tree can be obtained, thus the abstract syntax tree information can be transformed into a vector representation that can be recognized by the automatic scoring model, which provides effective input information for the subsequent automatic scoring of the model.

III. B. 3) Extracting feature vectors

Based on the vector representation of the abstract grammar tree information, it is proposed to input it into the graph attention network for feature vector extraction. The process of feature extraction based on graph attention network is:

(1) Firstly, the graph attention mechanism model is used to learn the linear mapping weight matrix $W \in R^{Fea' \times Fea}$. Then the attention coefficient between nodes i and j can be expressed as:

$$e_{ij} = \alpha\left(\overrightarrow{Wh_i}, \overrightarrow{Wh_j}\right) \tag{39}$$

where $\vec{W}$ denotes the projection matrix. h_i and h_j denote two nodes respectively, and e_{ij} is the degree of importance between the two nodes, and the nodes and the neighboring nodes are subjected to a weighted summation operation to obtain the vector specific representation:

$$e_{ij} = Leaky\,\mathrm{Re}\,LU\left(\vec{a}^T\left[\overrightarrow{Wh_i} \| \overrightarrow{Wh_j}\right]\right) \tag{40}$$

where $Leaky\,\mathrm{Re}\,LU$ denotes the activation function. $\|$ is responsible for splicing the hidden layer vectors of neighboring entities x_i, x_j. $\vec{a}^T$ denotes the feedforward neural network.

(2) Aggregate neighbor node information and normalize the attention of all neighbor nodes of the target node:

$$a_{ij} = soft\max{}_j(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \in N_i} \exp(e_{ik})} \tag{41}$$

where a_{ij} denotes the aggregation coefficients, and the $soft\max$ function is responsible for regularization normalization of node i 's neighbor node j.

(3) Vector weights are computed using the attention mechanism and parameterized and nonlinearized by the weight vector $\overline{a} \in R^{2F}$ and the $Leaky\,\mathrm{Re}\,LU$ with a negative slope of 0.2:

$$a_{ij} = \frac{\exp\left(Leaky\,\mathrm{Re}\,LU\left(\vec{a}^T\left[\overrightarrow{Wh_i} \| \overrightarrow{Wh_j}\right]\right)\right)}{\sum k \in N_i \exp\left(Leaky\,\mathrm{Re}\,LU\left(\vec{a}^T\left[\overrightarrow{Wh_i} \| \overrightarrow{Wh_j}\right]\right)\right)} \tag{42}$$

(4) Once the normalized attention coefficients are obtained, a linear combination of the target node features can be computed and used as the output features for each node:

$$\overrightarrow{h_t'} = \sigma\left(\sum_{j \in N_i} \alpha_{ij} \overrightarrow{Wh_j}\right) \tag{43}$$

III. B. 4) Similarity calculation

The feature vectors extracted based on graph attention network can be input to the graph embedding model for similarity calculation. Since the input object is an online exam sample, the problem of poor classification may occur when using traditional classifiers for language classification of exam samples, while the twin neural network (SN) can effectively solve the binary classification problem. Therefore, twin neural networks are used to extract features for similarity calculation for graph attention networks.

The twin neural network is obtained by constructing two artificial neural networks with two inputs. After inputting two neural networks in each of the two inputs, they are mapped to a new space to obtain a representation of the high-dimensional space of the two samples. Finally, the loss function is calculated and the similarity of the two samples is compared, thus obtaining the similarity between the student test sample and the template sample.

In order to ensure that the similarity calculation results have accuracy, it is proposed to adopt Contrastive Loss as the loss function of the twin neural network, through which the problem of poor classification of data processing is solved. The specific expression is:

$$Loss = (1-Y')\frac{1}{2}(E_w) + (Y')\frac{1}{2}\{\max(0, m-E_w)\}^2 \tag{44}$$

where E_w denotes the Euclidean distance between the twin neural network outputs. m denotes the maximum threshold value, and Y' can be calculated based on the above equation. When Y' takes the value of 0, it means that the two input vectors are similar, and if Y' takes the value of 1, it means that the two vectors are not similar. The shared weight value W of the network is updated by gradient backpropagation.

IV. Language structure and pedagogical analysis

IV. A. Structural analysis model assessment

In order to verify the effectiveness of the model, about 20 million English sentences with high rating quality were screened on the PARANMT-50M dataset and divided into two parts: the training set and the test set. For PKU Paraphrase Bank data, a total of about half a million sentence pairs are included, of which 90% of the data are randomly selected as the training set in this paper, and the remaining 10% are used as the test set to complete the training and evaluation of the Chinese rephrasing model.

In addition, in order to verify the performance of the model in different contexts and improve the generalization ability of the model, this paper selects scene-specific data to train and evaluate the model. The movie review dataset IMDB dataset and the English news dataset are selected in different English scenarios to train the model in different language scenarios to adapt to a variety of different linguistic expressions. The IMDB dataset is mainly used for the task of sentiment categorization, which is obtained from the Internet Movie Data Bank IMDB. The original dataset contains 50,000 movie reviews with severe bifurcation, and the data are equally divided into a training set of 25,000 and a training set of 25,000 and a training set of 25,000, respectively. The training set of 50,000 and the test set of 25,000, both the training set and the test set contain 50% positive reviews and 50% negative reviews. Since the model proposed in this paper does not have the function of sentiment recognition and is only designed to make the model applicable to the scenario of movie reviews, 40,000 pieces of data are screened directly from 50,000 pieces of data as the training set. The English news dataset is obtained from various news consulting platforms, which contains about 40,000 news data in various categories such as history, text, military, education, entertainment, etc. In this paper, we use sentence-by-sentence segmentation to obtain 26w news phrases as training. The statistical information of the dataset is shown in Table 1.

Table 1: Data set statistics

	PARANMT-50M		PKU Paraphrase Bank	
	Training set	Test set	Training set	Test set
Vocabulary number	30654	20089	22932	19624
Sample number	32645895	20357	568459	16042
Mean sentence number	16.2	13.5	38.5	30.6
Maximum sentence number	42	42	63	60
Minimum sentence number	3	3	3	3

In this paper, the unsupervised grammar analysis models such as SIVAE, VGVAE, and SynPG are selected to be compared and analyzed with the HHMM model in this paper, and the comparison of the results of the four unsupervised models under the review is shown in Table 2.

The four models present different advantages in the dataset under the review. Under the ParaNMT+MSCOCO+IMDB dataset, the HHMM model has the highest evaluation in generating analysis metrics that conform to the syntactic templates, which is 3.4 percentage points higher than the second best model (SIVAE), and this paper's model also achieves the optimal results in analysis metrics that do not conform to the syntactic templates, which is 16 percentage points higher than the SIVAE model. Meanwhile, comparing with the model SynPG, this paper shows an overall improvement in the effectiveness of analysis generation. For the PKU BANK+English news dataset dataset, the model of this paper also achieves an overall improvement of 5.1 percentage points over the SynPG model. For the ParaNMT+MSCOCO+IMDB dataset and the PKU BANK+English news dataset dataset, the HHMM model performs optimally under both datasets.

Table 2: Comparison of results of the unsupervised model under evaluation

Dataset	ParaNMT+MSCOCO+IMDB			
Model	2	1	0	2+1
SIVAE	52.4	23.4	24.7	73.4
VGVAE	48.7	19.7	34.1	69.2
SynPG	16.3	33.6	25.9	77.4
HHMM	55.8	39.4	36.5	84.6
Dataset	PKU BANK+English news dataset			
Model	2	1	0	2+1
SIVAE	37.2	27.9	37.7	64.3
VGVAE	35.8	20.4	48.5	55.7
SynPG	43.1	26.3	36.1	67.8
HHMM	46.6	29.2	50.1	72.9

Table 3: Automated English grammar teaching system availability evaluation index system

Primary index	Secondary index	Tertiary index
Teaching information availability (A)	Interactive interface information (A1)	Layout style (A11)
		Organizational form (A12)
		Navigation pane (A13)
	Content information (A2)	Knowledge structure (A21)
		Fine degree (A22)
		Performance form (A23)
	Technical assurance information (A3)	Privacy limit (A31)
		Error protection (A32)
		Operational response (A33)
Teaching interaction availability (B)	Individualized interaction (B1)	Resource design (B11)
		Resource management (B12)
		Resource sharing (B13)
	Social interaction (B2)	Discuss dialogue (B21)
		Collaborative task (B22)
		Homework evaluation (B23)
Teaching support availability (C)	Automated analysis (C1)	Accompanying collection (C11)
		Automatic processing (C12)
		Visual feedback (C13)
	Intelligent evaluation (C2)	Dynamic evaluation (C21)
		Fine evaluation (C22)
		Interpretability (C23)
	Humanized auxiliary (C3)	Teaching auxiliary (C31)
		Personality guidance (C32)
		Smart recommendation (C33)
Teaching improvement availability (D)	Knowledge & skill improvement (D1)	Professional knowledge (D11)
		Teaching skill (D12)
	Occupational identity improvement (D2)	Use willingness (D21)
		Value realization (D22)

IV. B. Results of the evaluation of the usability of automated instruction

IV. B. 1) Establishment of an evaluation indicator system

The analytic hierarchy process was used to establish a judgment matrix based on the usability evaluation dimensions of the automated English grammar teaching system that have been determined in this study. In the Yaahp software, there are three types of levels: 1) High level: also known as the goal level or decision-making goal, which contains only one element, and we take the "usability evaluation index system of the automated English grammar teaching system" as the control goal of this layer. 2) Middle layer: This layer is also called the program layer or criterion layer, which is the element that affects the decision-making of the upper layer, and we take "teaching information availability", "teaching interaction availability", "teaching support availability" and "teaching improvement availability" as the control criteria of this layer. 3) The lowest level: This layer is also called the index layer, which contains all the choice and decision-making schemes to achieve the goal, and is the index element that affects the upper level criteria, and we take "typesetting style" and "organizational form" as the index elements of this level, and finally form the hierarchical structure model of the usability evaluation index system of the automatic English grammar teaching system, as shown in Table 3.

Table 4: Automated English grammar teaching system availability evaluation index weight

Primary index	Weight	Secondary index	Weight	Tertiary index	Weight
A	0.258	A1	0.332	A11	0.326
				A12	0.345
				A13	0.329
		A2	0.347	A21	0.332
				A22	0.314
				A23	0.354
		A3	0.321	A31	0.362
				A32	0.315
				A33	0.323
B	0.259	B1	0.482	B11	0.306
				B12	0.338
				B13	0.356
		B2	0.518	B21	0.372
				B22	0.311
				B23	0.317
C	0.271	C1	0.336	C11	0.345
				C12	0.337
				C13	0.318
		C2	0.351	C21	0.302
				C22	0.365
				C23	0.333
		C3	0.313	C31	0.352
				C32	0.346
				C33	0.302
D	0.212	D1	0.513	D11	0.523
				D12	0.477
		D2	0.487	D21	0.489
				D22	0.511

In order to determine the weight values of each usability evaluation dimension, we need to statistically summarize the questionnaire data of the expert panel, quantitatively assign values to the two-by-two importance comparison opinions given by the experts, and construct a judgment matrix. The index weights of the usability evaluation index system of the automated English grammar teaching system are shown in Table 4.

IV. B. 2) Analysis of evaluation results

The automated English grammar teaching system designed in this paper was put into use and 500 questionnaires were distributed to the students using it after a period of time, and finally 487 valid questionnaires were obtained, with an effective rate of 97.4%. A 5-level Likert scale was used to measure the scores, and 1, 2, 3, 4, and 5 represented "very good", "good", "fair", "poor", and "poor", respectively. After processing the questionnaire data, the scores of the indicators of the usability of the automated English grammar teaching system are shown in Table 5.

The overall rating of the audience for the usability of the automated English grammar teaching system proposed in this paper is 4.21 points. In the first-level indicators, the scores of "availability of teaching information", "availability of teaching interaction", "availability of teaching support" and "availability of teaching improvement" were 4.26, 4.26, 4.16 and 4.17, respectively. The score range of the second-level indicators was [4.02, 4.31], with the highest score being "knowledge and skill improvement" and the lowest being "occupational identity improvement". The score range of the third-level indicators is 3.97~4.54, among which, the third-level indicator with the highest score is "privacy permission", and the lowest score is "job evaluation". Overall, the automated English grammar teaching system designed in this paper has achieved good results and has strong usability.

Table 5: Automated English grammar teaching system availability evaluation result

Primary index	Score	Secondary index	Score	Tertiary index	Score
A	4.26	A1	4.25	A11	4.23
				A12	4.39
				A13	4.12
		A2	4.27	A21	4.26
				A22	4.34
				A23	4.22
		A3	4.26	A31	4.54
				A32	3.98
				A33	4.23
B	4.26	B1	4.24	B11	4.29
				B12	4.04
				B13	4.38
		B2	4.28	B21	4.38
				B22	4.48
				B23	3.97
C	4.16	C1	4.30	C11	4.24
				C12	4.51
				C13	4.13
		C2	4.02	C21	4.00
				C22	4.01
				C23	4.04
		C3	4.17	C31	4.14
				C32	4.07
				C33	4.33
D	4.17	D1	4.31	D11	4.37
				D12	4.24
		D2	4.02	D21	4.06
				D22	3.98

V. Conclusion

The article utilizes unsupervised learning algorithms to construct an English grammar structure analysis model and proposes an automated English grammar teaching method. After evaluating the performance of the grammar structure analysis model, the usability of the automated grammar teaching system is investigated.

On ParaNMT+MSCOCO+IMDB dataset, the analysis indexes of this paper's HHMM grammar structure analysis model in terms of conforming and non-conforming to the grammar template are 55.8 and 84.6, respectively.On PKU BANK+English news dataset dataset, the analysis indexes of HHMM in terms of conforming and non-conforming to the grammar template are 46.6, 72.9, which are better than the other analysis models, and achieve the best performance of syntactic analysis.

The overall rating for the usability of the automated grammar teaching system in this paper is 4.21 points. The four first-level indicators of "availability of teaching information", "availability of teaching interaction", "availability of teaching support" and "availability of teaching improvement" scored 4.26, 4.26, 4.16 and 4.17 points respectively (all above 4 points), which shows that the automatic grammar teaching system has good effects in these four aspects. Among the second-level indicators, the highest score was "knowledge and skill improvement" (4.31), and the lowest score was "occupational identity improvement" (4.02). Among the three indicators, the highest score was "privacy permission" (4.54), the lowest score was "job evaluation" (3.97), and only two three indicators were below 4 points. The students were satisfied with the application of the grammar teaching system.

Funding

This research was supported by the Study on low-carbon economic scheduling strategy of multi-energy power system considering the quality constraints of big data interaction.

About the Author

Limin Zhang was born in Shangqiu, Henan, China, in 1984. She received her Master’s degree in linguistics from Xihua University in 2011, currently specializing in cognitive linguistics and foreign language teaching. She works as a teacher in Luoyang Institute of Science and Technology.

References

[1] Andriani, A., Yuniar, V. D., & Abdullah, F. (2021). Teaching English grammar in an Indonesian junior high school. Al-Ishlah: Jurnal Pendidikan, 13(2), 1046-1056.

[2] Saengboon, S., Kosin, P., & Toomaneejinda, A. (2022). The roles of grammar in English language teaching: Local viewpoints. Pasaa, 63(1), 179-204.

[3] Behlol, M. G., & Nasab, D. (2016). Content analysis of English textbook taught at grade 9: Focus on grammatical structures and language skills. Journal of Research in Social Sciences, 4(2), 122.

[4] Supakorn, P., Feng, M., & Limmun, W. (2018). Strategies for Better Learning of English Grammar: Chinese vs. Thais. English Language Teaching, 11(3), 24-39.

[5] Hampp, P. L., Kumayas, T. A., & Lengkoan, F. (2021). Synthesizing grammar and structure problems faced by Indonesian TOEFL participants. Jurnal Pendidikan Bahasa Inggris undiksha, 9(1), 64-68.

[6] Khasawneh, M. A. S. (2021). The effectiveness of using multimedia in the developing the concepts of the English language grammar concepts for people with learning difficulties. Science and Education, 2(6), 373-384.

[7] Jamshaid, S. (2024). Pakistani English: An Analysis of Grammatical-Syntactical Structures of Pakistani English Novels. Journal of Development and Social Sciences, 5(2), 335-352.

[8] Turopova, F. (2024). Strategies of teaching english grammar and its functions. STRATEGIES, 2(6).

[9] O’Keeffe, A., & Mark, G. (2017). The English Grammar Profile of learner competence: Methodology and key findings. International Journal of Corpus Linguistics, 22(4), 457-489.

[10] Kumayas, T., & Lengkoan, F. (2023). The challenges of teaching grammar at the university level: Learning from the experience of English lecturer. Journal of English Culture, Language, Literature and Education, 11(1), 98-105.

[11] Anglada, L., & Banegas, D. L. (2020). Reflections on English grammar instruction in EFL/ESL educational settings. Content knowledge in English language teacher education: International experiences, 49-64.

[12] Lai, C. H., & Lin, S. H. (2017). Systems theory. The international encyclopedia of organizational communication, 41, 1-18.

[13] Tiarina, Y., Syarif, H., Jufrizal, J., & Rozimela, Y. (2019). Students’ need on basic English grammar teaching material based on interactive multimedia: an innovative design. COUNS-EDU: The International Journal of Counseling and Education, 4(1), 29-37.

[14] Andika, A., Tahrun, T., & Firdaus, M. (2023). Developing English grammar instructional materials oriented to constructivism theory. Journal of Languages and Language Teaching, 11(2), 286-296.

[15] Vlčková, M. (2021). A Comparative Analysis of Grammatical Structures across English Language Textbooks at A2 Level. In INTED2021 Proceedings (pp. 10267-10276). IATED.

[16] Sukasame, N., Kantho, S., & Narrot, P. (2014). A Study of errors in learning English grammatical structures on tenses of Matthayomsuksa 4 students of the demonstration school, KhonKaen University. Procedia-Social and Behavioral Sciences, 116, 1934-1939.

[17] Waseel, A. K. (2020). Exploring Problems in Teaching and Learning Grammar in EFL Context: A Review Paper. International Journal of Language Research and Human Sciences, 3(7), 21-27.

[18] Nilesh B. Korade,Mahendra B. Salunke,Amol A. Bhosle,Prashant B. Kumbharkar,Gayatri G. Asalkar & Rutuja G. Khedkar. (2024). Strengthening Sentence Similarity Identification Through OpenAI Embeddings and Deep Learning. International Journal of Advanced Computer Science and Applications (IJACSA),15(4).

[19] Jinzhe Li. (2025). Hyperbolic Graph Convolutional Network Relation Extraction Model Combining Dependency Syntax and Contrastive Learning. International Journal of Computational Intelligence Systems,18(1),18-18.

[20] Long Yang,Xiaojie Yin,Zhenhao Li,Zhiyu Ding,Yue Zou,Ziwei Li... & Weigang Hu. (2025). Adaptive radiotherapy triggering for nasopharyngeal cancer based on Bayesian decision model. Physics in medicine and biology,70(7),075015-075015.

[21] Jianjie Guo,Lin Guo,Wenchao Xu & Haibin Zhang. (2025). Hidden Markov model with Pitman-Yor priors for probabilistic topic model. Communications in Statistics - Theory and Methods,54(9),2791-2805.

[22] Sacchi Giada & Swallow Ben. (2021). Toward Efficient Bayesian Approaches to Inference in Hierarchical Hidden Markov Models for Inferring Animal Behavior. Frontiers in Ecology and Evolution,9.

[23] Tianyi Xu,Yan Wang,Wenbin Zhang,Yue Zhao,Jian Yu,Mei Yu... & Mankun Zhao. (2025). Multi-perspective semantic decoupling and enhancement in graph attention network for knowledge graph completion. Applied Intelligence,55(7),629-629.