Semantic Coherence Analysis of English Texts Based on Sentence Semantic Graphs

Nanxiao Deng[1], Yabing Wang[1], Guimin Huang[1,2*], Ya Zhou[1] and Yiqun Li[1]

[1]School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin 541004, Guangxi, China
[2]Guangxi Key Laboratory of Image and Graphic Intelligent Processing

Abstract

With the reform of China's education industry, more and more universities are using computers to conduct examinations. For the automatic correction of essays as subjective questions, existing automatic English text scoring systems suffer from insufficient extraction of coherence information and low accuracy when analysing text coherence. Therefore, this paper proposes an unsupervised semantic coherence analysis model for English texts based on sentence semantic graphs, taking Chinese students' English compositions as the research context. Guided by the semantic coherence theory, the English text is represented as a sentence semantic graph, and an improved VF2 subgraph matching algorithm is used to mine the frequently occurring subgraph patterns in the sentence semantic graph. After that, the set of frequent subgraphs is generated by filtering the subgraph patterns according to their frequencies, and the subgraph frequency of each frequent subgraph is calculated separately. Finally, the distribution characteristics of frequent subgraphs and the semantic values of subgraphs in the sentence semantic graphs are extracted to quantify the overall coherence quality of English texts. The experimental results show that the model proposed in this paper has higher accuracy and practical value compared with the current methods of coherence analysis.

Keywords: english text, semantic coherence theory, sentence semantic graph, VF2 subgraph matching algorithm, frequent subgraph

Received on 03 May 2023, accepted on 27 August 2023, published on 28 August 2023

doi: 10.4108/ eetsis.3312

*Corresponding author. Email: sendhuang@126.com

1. Introduction

In recent years, techniques related to natural language processing have become more and more widely used. As an important application in the field of natural language processing, the detection and evaluation of semantic coherence has developed rapidly [1-3]. There is an urgent need for researchers to assess the coherence quality of the large number of textual results generated by many intelligent systems, such as the results of automatic abstract generation and machine translation, because if the coherence of these texts is poor, it will create a significant barrier to reading and even lead to incomprehension of the meaning of the text. In addition, the quality of coherence is an important criterion in all English composition scoring systems, and its analysis is essential. As a result, researchers have begun to investigate and quantify the quality of text coherence with a view to its practical application, and this has led to the study of text coherence. Generally speaking, when scoring English language learners' essays, the scoring criteria should cover four aspects: lexical complexity, grammatical accuracy, syntactic complexity and discourse coherence [4], which are the prerequisites for accurate and reliable scoring results. However, existing automatic English essay scoring systems rarely address the indicator of coherence, which results in an unreasonable final score for English essays. For example, when a large number of sentences are inserted into an essay with excellent lexical complexity, grammatical accuracy and syntactic complexity, the final score will be high even if the overall coherence of the essay is not high. This increases the

likelihood that students will write deceptive essays by inserting florid sentences that are illogical and thus mislead the machine's scoring. Therefore, the analysis of coherence plays an important role in the accuracy and robustness of automatic scoring of English texts.

Currently, the analysis of discourse coherence in English texts faces challenges where inadequate extraction of coherent information results in suboptimal model training outcomes. To address this, we propose an unsupervised English text semantic coherence analysis model based on sentence semantic graphs, aimed at more effectively extracting textual coherence features. The entity graph based on entity construction and the semantic similarity graph based on semantic similarity are fused into a sentence semantic graph to relate the non-adjacent sentences in the text, so that the semantic relationship between sentences can be visualized. The subgraph matching algorithm mines the frequent subgraph patterns in the sentence semantic graph to capture specific coherence patterns in the English text and subdivides the weights of the edges in the graph to measure the specific degree of coherence between sentences. Using the above approach, we designed a semantic coherence analysis model for English texts based on the work of Guinaudeau and Strube et al. [5]. The main contributions of this paper are as follows:

(1) To address the problem of insufficient semantic information between sentences in the traditional entity graph model, under the guidance of semantic coherence theory, the Word2Vec word embedding model is used to represent the English text in the semantic space and combine the semantic similarity information between sentences with the entity information in the entity graph to construct semantic associations between sentences, thus representing the English text as a semantic graph of sentences containing rich semantic information.

(2) In order to accurately capture the coherence features in English texts, the improved VF2 subgraph matching algorithm is used to mine the frequent subgraph patterns in the sentence semantic graph, which is used to simulate the unique coherence patterns in English texts, and then to analyse the overall coherence of English texts.

(3) Based on the frequency of different subgraph patterns in the sentence semantic graph, the subgraphs are filtered to generate a set of frequent subgraphs, and the subgraph frequency of each frequent subgraph is calculated separately. Finally, the distribution characteristics of the frequent subgraphs in the sentence semantic graph and the semantic values of the subgraphs are extracted to quantitatively analyse the overall coherence quality of the English text.

(4) The remainder of this paper is organized as follows. The second part presents work related to the analysis of English text coherence. The third part details our proposed semantic coherence analysis model for English texts. In the fourth part, the experimental results are compared and analysed. The fifth part summarises the results of the work in this paper and looks forward to the next step.

2. Related Work

With the rise of a range of natural language processing applications such as automated question and answer and text generation, the use of text coherence analysis has grown exponentially [6-11]. Among the large number of textual results generated by intelligent systems, English texts are more common. Therefore, it is necessary to study the coherence of English texts.

Research on English text coherence analysis falls into two main categories. The first is based on Latent Semantic Analysis (LSA), which analyses the coherence of a text by transforming it into a vector space model, dimensioning it down and calculating the cosine between two words or sentences. Liping Pu et al. [12] applied LSA to the computational tools Coh-Metrix and TAACO to analyse English compositions written by non-native English learners. SR Vrana et al. [13] used LSA to assess the coherence of traumatic narrative texts. However, due to the drawbacks of LSA such as poor interpretability and the inability to handle the phenomenon of word polysemy, many researchers have turned their research towards solid grid models. The second type of research is based on the entity lattice model. It measures the coherence of a text by extracting the common entities of adjacent sentences and counting the frequency of grammatical role transitions of these common entities in different sentences. However, the entity lattice model has some limitations. Firstly, it is a supervised model, which is affected by data sparsity and domain dependency, and secondly, it can only analyse the coherence between adjacent sentences, which is local coherence. Therefore, researchers have made many extensions to it. Luyao Teng et al. [14] proposed that graph-based methods are widely applied when studying the structure and relationships among research data, demonstrating the effectiveness of using graphs for data feature extraction. Jiao Yin et al. [15] investigated knowledge transfer between subgraphs extracted from the same knowledge graph, leveraging the graph structure to extract data relationships and applying them in software vulnerability detection. Rangjun Li [16] suggested that considering both intrinsic attribute information and inter-sample structural information simultaneously enhances the feature recognition capability of the model. Guinaudeau and Strube et al. [5] extended the entity grid into an entity graph model to represent a text in a graph, thus analysing the coherence of the text in its entirety. Takenobu Tokunaga et al. [17] constructed semantic similarity graphs by means of word embedding, so that the degree of semantic relevance of different sentences could be distinguished. Guimin Huang et al. [18] propose a new discourse coherence quality analysis model (sentence semantic graph) by merging entity graph and semantic similarity graph of the text. In addition, knowledge graphs, as an extension of graph data structures, have also found widespread applications in extracting relationships among data [19]. The semantic coherence information in text, as one form of data feature, can be a source of inspiration. M. Gao et al. [20] proposed that the multi-relational semantics in knowledge graphs can further enhance the model's

interpretability, but an excessive reliance on the feature data from knowledge graphs may limit expressive capacity. In recent years, as deep learning has made significant gains in the field of artificial intelligence, researchers have experimented with deep learning to analyse the coherence of text, and some progress has been made. M Hung et al. [21] used deep learning methods to construct a model for English text coherence analysis, which performed well in English text coherence diagnosis. Vasamsetti Srinivas et al. [22] used deep neural networks to build pre-trained language models to learn text features, and also made research progress. Ping Li [23] proposed a novel approach for English relation extraction using a residual network-based temporal feature extraction framework, which effectively captures English relationships.

In summary, researchers have proposed research methods for the study of text coherence quality in a number of ways, with one of the more popular ideas being a series of improvements and extensions based on the solid grid model. Based on Guinaudeau and Strube et al. [5], this paper designs a semantic coherence analysis model for English text by fusing entity-based construction of entity graphs with semantic similarity graphs based on semantic similarity into a sentence semantic graph and capturing the unique coherence patterns in English text by mining frequent subgraph patterns in the sentence semantic graph with a subgraph matching algorithm.

3. Methodology

The semantic coherence analysis model for English text in this thesis consists of five parts: pre-processing of text, entity graph construction, generation of sentence semantic graphs, mining of frequent subgraphs and coherence quality analysis. The overall processing flow of the model in this paper is depicted in Figure 1. Detailed descriptions of each component will be provided in this chapter.

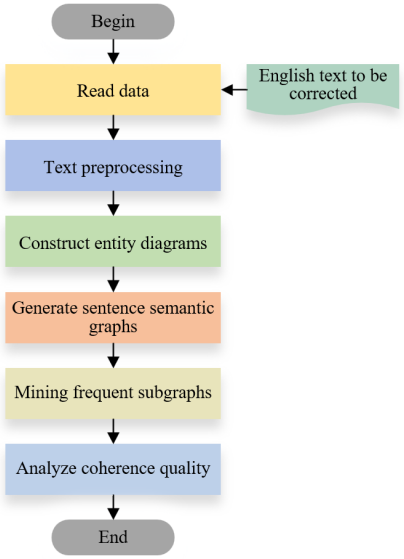


Figure 1. Display of the results of the lexical annotator tagging

3.1. Pre-Processing Of Texts

Text Slice Processing

In this paper, regular expressions are used to achieve a cut-and-score processing of the English text to be approved based on the characteristics between paragraphs, sentences and words. Two consecutive occurrences of the newline marker are cut into paragraphs; in terms of subsentences, two cases are distinguished, as a full stop can indicate the end of a sentence in addition to an abbreviation character and a surname; the presence of a space is cut into a word.

Part-Of-Speech Tagging Processing

In this paper, the lexical annotator we use is a lexical annotator based on recurrent dependency neural networks [24]. Unlike traditional maximum entropy or decision tree lexical annotators that use one-way inference of word sequences to annotate words with lexical properties, it uses a two-way symmetric inference method, and its results are more accurate. Figure 2 shows the result of the lexical annotation of the sentence "It is our responsibility that to keep a harmonious and peaceful campus." The labels above the words are the lexical properties of the words, e.g., NN for nouns and VB for verbs.

Figure 2. Display of the results of the lexical annotator tagging

Dependency Syntax Analysis

This paper uses a neural network-based dependent syntactic parser that enables fast parsing of sentences. We use the parsing algorithm to analyze the syntactic structure of the sentence. The input sentence is represented by the word sequence $W=\{w_1,w_2,\ldots,w_n\}$ of the sentence, where w_i represents the i^{th} word in the sentence to be parsed. In addition, the lexical sequence of the sentence is also inputted correspondingly, represented by $X=\{x_1,x_2,\ldots x_n\}$. Combining the above input information, we use a representation of the dependency in the form of a triple (h,r,s), which is usually expressed as:

$$f=\{(h,r,s):0\le h\le \mathrm{n},1\le r\le n,s\in S\} \qquad (1)$$

In equation (1), h in the triplet (h,r,s) represents core words, r represents modifiers, s represents the type of dependency between core words and modifiers, and S represents the set of dependency types. Figure 3 shows the dependency parsing results of the sentence "We should try to learn science at the university instead of playing computer", displaying the meanings of the abbreviated dependency relationships as presented in Table 1.

Table 1. Dependency Table

Dependancy	Meaning
nsubj	nominal subject
aux	auxiliary
mark	marker
xcomp	open clausal complement
dobj	direct object
case	case marking
det	determiner
nmod	nominal modifier
acl	clausal modifier of noun (adnominal clause)
mwe	multi-word expression

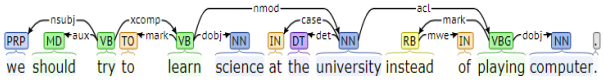


Figure 3. Display of dependency syntax analysis results

3.2. Construction Of Entity Graph

The construction of entity diagrams is divided into four steps: identification of entity words, co-reference disambiguation of entity words, grammatical role annotation of entity words and construction of entity diagrams. Each step is described in detail as follows.

Recognition Of Entity Words

Entity words generally act as noun or pronoun attributes in English texts, so we extract nouns and pronouns based on the results of lexical annotation in the preprocessing module. However, in some English texts, numbers and symbols that are not useful for text coherence analysis are also labelled as nouns, which are noisy factors. Therefore, after extracting all the entity words, the model will also filter the entity word set in order to reduce the noise effect.

Co-reference Resolution Of Entity Words

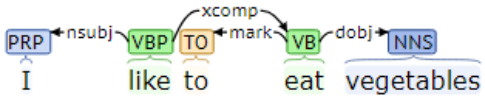
The phenomenon of co-reference between entity words can pose a major obstacle to textual analysis. Therefore, it is necessary to disambiguate the co-reference phenomenon in the text. The boundaries of the noun phrases in the text are first determined by traversing the syntactic tree.After that, we adopt the co-referencing disambiguation method based on the hierarchical filtering model proposed by Rahunathan and Lee et al. [25-26] to disambiguate the entity word referencing phenomenon. Table 2 shows an example of co-reference disambiguation for a short text.

Table 2. Example of coreference resolution

Essay example	"[Jerry] $_{e1}$ is [my favorite football player] $_{e2}$. When [he] $_{e3}$ was [7 years old] $_{e4}$, [he] $_{e5}$ tried to learn [foo tball] $_{e6}$."
Coreference chain	{[Jerry] $_{e1;}$ [my favorite football player] $_{e2}$. [he] $_{e3;}$ [he] $_{e5}$}

Grammatical Role Labeling of Entity Words

In this model, we classify the grammatical roles of entity words into three: subject (S), predicate (O) and presence (X). As shown in Figure 4, the dependency annotation for the sentence "I like to eat vegetables" is as follows, displaying the meanings of the abbreviated dependency relationships as presented in Table 1.



nsubj(like -2,I-1),xcomp(like-2,eat-4),mark(eat-4,to-3),dobj(eat-4,vegetables-5)

Figure 4. Simple sentence dependent syntax parsing results

Constructing Entity Graph

After these two steps, information on the co-reference of the entity words in the English text and the grammatical role of the entity words in the sentence can be obtained, and by combining these features, the entity graph model of this module can be constructed. We have made some improvements on the solid graph model of Guinaudeau and Strube et al. [5]. On the one hand, we build our entity graph model by using only P_{Acc} projection for bipartite graphs with unimodal projection. On the other hand, in order to facilitate the quantitative analysis for subsequent processing, we set different weights for the grammatical roles of the entity words, specifically, φ_1 for subject (S), φ_2 for object (O) and φ_3 for presence (X) and $1 \succ \varphi_1 \succ \varphi_2 \succ \varphi_3 \succ 0$ .In addition we filter out some edges that are below a set threshold to reduce the interference caused to the model. The formula for calculating the weights of sentence edges is as follows:

$$W_{\text{entity}}(S_i, S_k) = \frac{\sum_{e \in E_{ik}} w(e, s_i) \cdot w(e, s_k)}{|k - i|} \qquad (2)$$

E_{ik} in Equation (2) is the set of entities shared by sentence s_i and sentence s_k , and $w(e, s_i)$ and $w(e, s_k)$ represent the grammatical role weights of the shared entity e in sentences s_i and s_k . After projecting and calculating the weights of the edges, we can represent an English text as an entity graph. Figure 6 is the entity graph representation of the test text in Figure 5.

As children, we should understand parents well. We should know most behaviors of parents are good for us. So, we need communicate with our parents more actively. And as parents, we need listen to voices of children more patiently. Sometimes, children's idea is pretty good. It's useful to us helping children grow up that we understand the children's point of view. It is the best way to solve adolescent problems establishing a good relationship between parents and children. So, children understand their parents more, and parents give their children more freedom.

Figure 5. Example of test English text

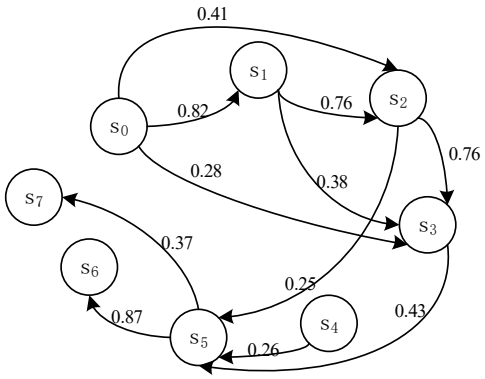


Figure 6. Entity diagram of test text

3.3. Generating Sentence Semantic Graphs

The entity diagram model is incomplete in analysing the coherence of English texts by focusing only on the transfer patterns of entity words in sentences, i.e., by analysing the coherence of English texts in terms of lexical recurrences. This is because in real English texts, lexical reduplication does not constitute a large proportion of the text, and there are also sentences that do not share a common entity but are also semantically coherent. An example is the two sentences in Figure 7.

S1:The husband has finished his meal.
S2:The wife is going to wash the dishes.

Figure 7. Example of a coherent sentence

In Figure 7, the meaning of the sentence is "The husband has finished eating and the wife is getting ready to wash the dishes". It is clear from this that the sentences S1 and S2 are semantically coherent, but there is no common entity or referential phenomenon between them, and the entity diagram model does not accurately analyse them coherently. Therefore, in order to solve such problems, this model is improved on the basis of the entity graph, by distributing the sentences in the English text and calculating the semantic similarity between sentences to incorporate more semantic information between sentences into the entity graph model, thus constructing a sentence semantic graph model that contains rich semantic information of sentences.

Distributed Representation of Sentences

The sequence of words of a sentence S_i is represented as $S_i=\{w_1,w_2,\cdots w_m\}$. The vector of words takes the form $Vec(w_i)=\{v_1,v_2,v_3,\cdots v_n\}$, then the vector of sentences S_i is represented as follows:

$$Vec(S_i)=\sum_{m}^{j=1}(\lambda_1 Vec(w_s)+\lambda_2 Vec(w_o)+\lambda_3 Vec(w_x)) \quad (3)$$

In equation (3), λ_1, λ_2 and λ_3 are the weights of the subject component, object component and presence component of the sentence respectively, m is the number of words in the sentence, and w_s, w_o and w_x are the subject, object and words present in the sentence that do not act as the main component respectively.

Semantic Similarity Calculation

In the vector semantic space, the cosine value between sentence vectors is then a good reflection of the relationship between them. Therefore, the cosine similarity algorithm is used to calculate the semantic similarity between sentences, which is represented as follows.

The vector representations of the sentences S_i and S_j are respectively

$$S_i=\left[v_{1,i},v_{2,i},v_{3,i},\cdots,v_{n,i}\right] \quad (4)$$

$$S_j=\left[v_{1,j},v_{2,j},v_{3,j},\cdots,v_{n,j}\right] \quad (5)$$

Then the semantic similarity between two sentences is represented as follows:

$$Similarity(S_i,S_j)=\frac{\sum_{n}^{h=1}v_{h,i}\cdot v_{h,j}}{\sqrt{\sum_{n}^{h=1}v^2{}_{h,i}}\cdot\sqrt{\sum_{n}^{h=1}v^2{}_{h,j}}} \quad (6)$$

If the value of semantic similarity is equal to 1, the two sentences are basically the same; if the value of semantic similarity is greater than 0 and less than 1, the two sentences are semantically similar, and the degree of similarity depends on the absolute value of cosine similarity.

Generate Sentence Semantic Graphs

First, a threshold β is set, and then, when improving the entity graph, the set of semantic similarity values between each sentence and other sentences is traversed. If a semantic similarity value is greater than the initial threshold, the two sentences are considered to be coherent. If an edge has been established between these two sentences in the entity graph, the weight of this edge is updated again; if no edge has been established between the two sentences in the entity graph, a new edge is created. The weights of the edges are calculated from the weight formula as follows：

$$Weight(S_i, S_j) = \eta_1 W_{Entity}(S_i, S_j) + \eta_2 W_{Similarity}(S_i, S_j) \tag{7}$$

In the Equation (7) , η_1 and η_2 are the weight coefficients, $W_{Entity}(S_i, S_j)$ and $W_{Similarity}(S_i, S_j)$, are respectively the weight between two sentences in the entity graph and the semantic similarity between the two sentences.

3.4. Mining Frequent Subgraphs

Unlike Guinaudeau and Strube et al. [5] who used the feature of average out-degree to measure the overall coherence of English texts, in this model, we use frequent subgraphs, a more complex graph feature, to capture the uniqueness of English texts. The coherence pattern of, which uses different frequent subgraph frequencies to analyze the coherence of English texts.

Improved VF2 Subgraph Matching Algorithm

The VF2 algorithm works by ending the search as soon as a subgraph state is found that fully satisfies the conditions and returning the result. Therefore, it is not possible to search for all subgraph structures in the target graph that are isomorphic to the query graph. In order to improve the accuracy of the model, we have made some improvements to the original algorithm procedure, i.e., when a subgraph state satisfying the condition is matched, the result is saved and then traversed through the other candidate node pairs so that all subgraphs in the target graph that are isomorphic to the query graph can be searched.

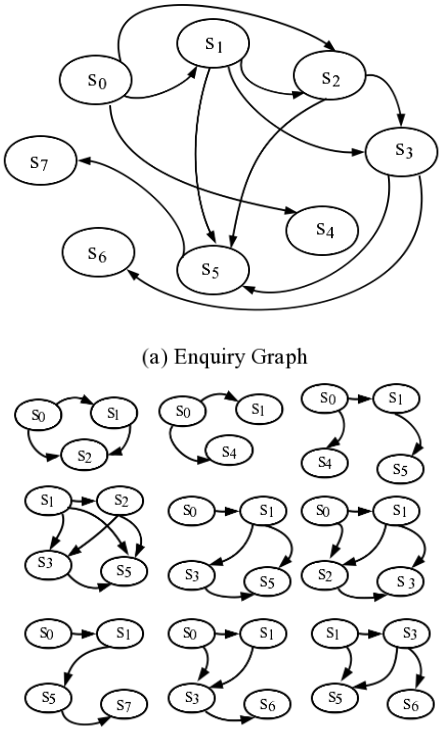
It should be noted that the same query graph may have two or more matches on a subgraph structure in the target graph, which can have a significant noise impact on the subsequent analysis of the results. Therefore, we need to further filter the matching results (bolded part in the pseudo-code) in order to exclude the influence of isomorphic subgraphs [27] and ensure that the same query graph can only have at most one matching result on a subgraph structure in the target graph.

Generating Frequent Sub-Graphics

In this paper, an efficient VF2 subgraph matching algorithm is used to obtain a set of frequent subgraphs that reflect the coherent patterns of English texts that occur frequently in the training set by using a large number of well-connected English texts for training.

First, the text data in the training set are to be generated into corresponding sentence semantic graph data, and the sentence semantic graph data are to be converted into a graph format suitable for the search of the VF2 subgraph matching algorithm, which is used as the target graph set of the algorithm. Meanwhile, to avoid data sparsity, we only consider three-node and four-node subgraphs and use them as the query graph set for input. During the search process, the set of query graphs is first traversed and the total number of times a particular query graph appears as an isomorphic subgraph in all target graphs is counted, and the total number of times is then compared with the frequency factor. If the total number of occurrences is greater than or equal to the frequency factor, the query graph is considered to be frequent and the label of the query graph and its corresponding number are stored; if the total number of occurrences is less than the frequency factor, the query graph is considered to be infrequent. If the total number of times is less than the frequency factor, it is considered to be infrequent. This cycle is continued until the end of the query diagram traversal, and then the corresponding subdiagram pattern is found according to the saved subdiagram number to generate the frequent subdiagram set.

An example of frequent subgraph mining for an English text is shown in Figure 8. Figure (a) shows the target graph converted from the sentence semantic graph of the English text, and Figure (b) shows the partial frequent subgraph results obtained by subgraph matching on this target graph.



(a) Enquiry Graph

(b) Target Graph

Figure 8. Example of frequent subgraph mining

3.5. Coherent Quality Analysis

Coherent Feature Extraction

This model treats the following graph features as coherent features for extraction.

Frequent subgraph frequency: The ratio of the frequency of each k-node frequent subgraph in the training set to the sum of the frequencies of all k-node frequent subgraphs is called the frequent subgraph frequency of the frequent subgraph, which is calculated as follows:

$$P(\mathrm{sg}_i) = \frac{N(\mathrm{sg}_i)}{\sum_{i=1}^{n} N(\mathrm{sg}_i)} \tag{8}$$

In Equation (8), $N(\mathrm{sg}_i)$ denotes the number of occurrences of k-node frequent subgraph sg_i in the text of the training set, i is the ordinal number of the current k-node frequent subgraph in the set of all k-node frequent subgraphs, $i = 1, 2, \cdots \mathrm{n}$; n is the total number of k-node frequent subgraphs.

Figure signature $\Phi_{\mathrm{k}}(G)$: The vector generated by the combination of the frequencies of all k-node frequent subgraphs occurring in a sentence semantic graph is called the graph signature of this sentence semantic graph, which reflects the distribution of different k-node frequent subgraphs in the sentence semantic graph, (k=3, 4). The graph signature is represented as follows：

$$\Phi_{\mathrm{k}}(G) = (\phi(sg_1, G), \phi(sg_2, G), ..., \phi(sg_m, G)) \tag{9}$$

$$\phi(\mathrm{sg}_i, G) = count(\mathrm{sg}_i, G) \tag{10}$$

In Formula (9), $\phi(sg_1, G)$ represents the frequency of occurrence of frequent subgraph sg_1 in sentence semantic graph G.

Subgraph semantic value:The sum of the weight values of all edges in a frequent subgraph is defined as the subgraph semantic value of a frequent subgraph, as follows:

$$SemanticValue(sg_i) = \sum_{e \in E} Weight(e, sg_i) \tag{11}$$

In Formula (11), $Weight(e, sg_i)$ represents the weight value of sentence edge e in frequent subgraph sg_i, and E represents the set of frequent subgraph sg_i edges.

Consistent Quality Analysis

Coherence quality analysis is the most important part of all coherence analysis models, and each of them has a different approach. Most of the traditional sentence diagram models use the feature of average output to measure the coherence quality of English texts, but such an approach does not accurately capture the coherence information of English texts, and therefore its experimental results are not satisfactory. We realise that a text with good coherence follows a specific logical and coherent pattern between words or sentences within its discourse. Based on this, we used the frequent subgraph approach to capture the coherence patterns in the text. In a sentence semantic graph, the coherence information of a text is reflected as the difference in the distribution of connection patterns between sentence nodes and the weight values of sentence edges, so this paper analyses the coherence quality of English texts by capturing the frequency of these frequent subgraph patterns and the semantic values of the subgraphs. The specific analysis process is as follows:

(i) Using a large number of coherent English texts as a training set for training, the frequency of occurrence of all three-node and four-node subgraphs in the training set was counted;

(ii) setting the frequency coefficients, filtering out the frequent subgraph patterns in the training set, and calculating the probability of occurrence of each frequent subgraph pattern to generate a frequent subgraph model, and using it as a frequent subgraph distribution feature of English texts with good coherence quality;

(iii) Extracting graph signature and subgraph semantic value information from the sentence semantic graph representation of the English text to be analysed;

(iv) Combined with the frequent subgraph-related calculation method of Leo Born et al. [28], the distribution features of frequent subgraphs in the sentence semantic graph are used to design an algorithm to analyse the coherence quality of English texts. Its calculation formula is as follows:

$$CoherenceScore(G) = \frac{\sum_{j=1}^{n} \lambda_j \sum_{i=1}^{m} P(\mathrm{sg}_i) \times \phi(sg_i, G) \times SemanticValue(sg_i)}{SentenceNum(G)} \tag{12}$$

In equation (12), m is the total number of k-node frequent subgraphs in the English text to be analysed, n is the number of values of k, and in this paper the value of n is 2. $P(sg_i)$ is the frequency of the i-th k-node frequent subgraph in the English text to be analysed, $\phi(sg_i, G)$ is the number of occurrences of the frequent subgraph sg_i in graph G, $SemanticValue(sg_i)$ is the subgraph semantic value of the frequent subgraph sg_i , and $SentenceNum(G)$ is the number of nodes in the sentence semantic graph G. Finally, $CoherenceScore(G)$ normalised the coherence quality score of the English text to a value between 0 and 1. The closer the value is to 1, the better the coherence quality of the English text.

4. Experiment

4.1. Data Set and Model Evaluation Criteria

The data set consisted of the ICNALE [29], COLEN [30], CELC [31] and TECCL [32] corpora. One thousand articles were selected from the ICNALE corpus under the same essay topic as the test set for incoherent sentence extraction;

100 unlabelled plain-text English texts from the COLEN corpus were selected as the test set for sentence ranking experiments on this model. Two hundred essays scoring between 11 and 14 and 200 essays scoring between 6 and 9 (out of 15) from the CELC corpus were used for the filtering work on the frequent subgraph set. The 9,000 essays from the TECCL and the 6,000 essays from the CELC corpus were used for training to generate our frequent subset, and an additional 500 essays from the TECCL were used as a test set for comparison experiments with teacher ratings.

In this paper, evaluation metrics widely used in the field of natural language processing analysis: accuracy, precision, recall, F1 value and Pearson correlation coefficient.In marking English composition for coherence, a scale for the quality of English composition for coherence has been developed with reference to the marking criteria for CET-4 and CET-6 English composition and the actual manual marking situation. There are four levels of coherence, ranging from 20-25, 13-20, 7-13 and 0-7.

4.2. Analysis of Experimental Data

Experimentation and Analysis of Filtering Subgraphs

In this paper, 200 English essays with a coherence score of 11-14 (total score of 15) were taken from the CLEC corpus as the test sample with good coherence, which is denoted by Sample-1; and 200 English essays with a score of 6-9 were taken as the test sample with poor coherence, which is denoted by Sample-2. Figure 9 shows some of the frequent subgraph patterns, and Table 3 shows their performance in the test set, where the first two columns indicate the frequency of subgraph pattern *Sgi* in the two types of test sets, and the last two columns indicate the average of the number of occurrences of subgraph pattern *Sgi* in the two types of test sets.

Table 3. Number of Frequent Subgraphs in Different Test Texts

Subgraph mode	Sample-1	Sample-2	Avg-1	Avg-2
*Sg*1	148	87	0.74	0.44
*Sg*2	210	80	1.05	0.40
*Sg*3	122	74	0.61	0.37
*Sg*4	128	68	0.64	0.34
*Sg*5	119	48	0.60	0.24
*Sg*6	142	91	0.71	0.46
*Sg*7	131	87	0.66	0.44
*Sg*8	178	102	0.89	0.51
*Sg*9	189	126	0.95	0.63
***Sg*10**	**224**	**201**	**1.12**	**1.01**
*Sg*11	86	30	0.43	0.15
***Sg*12**	**300**	**278**	**1.50**	**1.39**
***Sg*13**	**380**	**351**	**1.90**	**1.76**
*Sg*14	165	114	0.83	0.57
*Sg*15	138	96	0.69	0.48

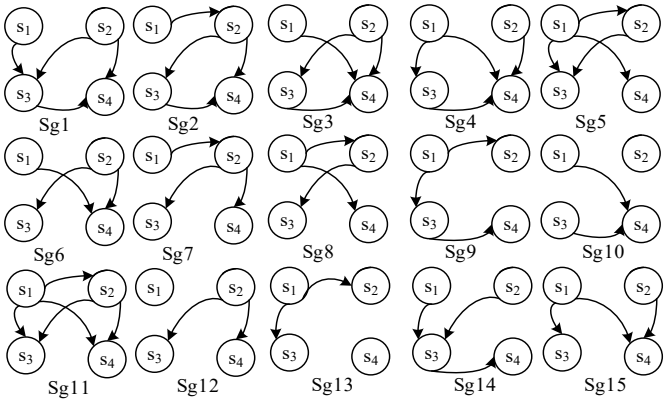


Figure 9. Partial frequent subgraph pattern

The subgraph patterns sg10, sg12 and sg13, which occur in good and poorly coherent test texts, are not very different on average and do not capture the coherence patterns of the text very well, so we filter out these subgraphs and keep those that perform well.

Experimentation and Analysis of Extracting Incoherent Sentences

In order to verify the accuracy of the semantic coherence model in extracting incoherent sentences, we conducted experiments on the recognition of incoherent sentences. A random sample of 1000 articles from the Asian Corpus of Learners (ICNALE) was used as the test set, and four incoherent English sentences were randomly inserted into each article as manual annotation of incoherent sentences. The incoherent sentences were extracted with an extraction threshold θ. We use the test set to experiment with different incoherent sentence extraction thresholds in order to find the most suitable extraction threshold. The results of the experiments are shown in Table 4.

Table 4. Experimental Results of Incoherent Sentence Extraction Under Different Thresholds

Threshold θ	Accuracy	Recall	F1
0.26	82.21%	80.50%	81.35%
0.28	83.34%	81.48%	82.40%
0.30	84.36%	82.31%	83.32%
0.32	86.54%	85.75%	86.14%
0.34	**88.43%**	**86.67%**	**87.54%**
0.36	86.23%	87.45%	86.83%
0.38	84.25%	88.21%	86.18%
0.40	81.36%	88.60%	84.83%
0.42	80.46%	90.15%	85.02%

From the data in the Table 4, it can be concluded that when the threshold value θ for incoherent sentence extraction is set to 0.34, the F1 value for incoherent sentence extraction

in this model is optimally 87.54%, so the threshold value for incoherent sentence extraction in this model is set to 0.34.
In order to verify the performance of this model in extracting incoherent sentences under different numbers of essays, the threshold of incoherent sentence extraction was set to 0.34. The results of the incoherent sentence extraction experiments under different numbers of essays are shown in Table 5.

Table 5. Experimental results of incoherent sentence extraction with different numbers of English compositions

Number of English essays	Accuracy	Recall	F1
20	83.65%	88.20%	85.86%
40	85.31%	87.80%	86.53%
60	84.64%	87.82%	86.20%
80	86.01%	87.55%	86.77%
100	86.77%	87.16%	86.96%
200	87.04%	86.95%	86.99%
400	87.23%	86.87%	87.05%
600	87.71%	86.80%	87.25%
800	88.12%	86.71%	87.41%
1000	88.43%	86.67%	87.54%

From the experimental results in Tables 4, we can see that the performance of the model in this paper in extracting incoherent sentences is relatively stable for different numbers of English compositions as the test set. The accuracy rate for extracting incoherent sentences also increases with the number of compositions, and the recall rate decreases. Therefore, the sentence semantic graph model constructed in this paper performs relatively well in the experiments.

Experiment and Analysis of Sentence Ranking

The sentence ordering task is one of the most commonly used methods for testing coherence analysis models. We randomly selected 100 articles from the COLEN corpus as the test set, disordered the sentences in each article to generate 10 disordered articles, and then used the original article and one disordered article as a set of samples, for a total of 1000 test sets. The semantic similarity graph model of Takenobu Tokunaga et al. [14], the entity graph model of Guinaudeau and Strube et al. [5] and the model of this paper were compared in terms of accuracy on this test set. The experimental results are shown in Table 6.

Table 6. Sentence sorting experiment results of different models

Model		Accuracy (%)
Entity graph model	P_{Acc}	75.80
Semantic Similarity Graph Model	PAV	80.80
	SSV	78.30
	MSV	79.50
This paper model		**84.50**

From the experimental results in Table 5, it can be seen that this model combines the advantages of the entity graph model and the semantic similarity graph model, with an accuracy of 84.5%, indicating that this model is more capable of distinguishing the degree of coherence of different texts and has better performance in text recognition.

Comparison Of Model and Teacher Scoring

To test the effectiveness of the model in practical use, we compared the scoring of essays by the model in this paper with the results of manual teacher scoring. We selected 500 student essays from the TECCL corpus as the test set. At the same time, five English teachers were invited to rate the coherence quality of the English compositions in the test set. The results of the experiment are shown in Figure 10.

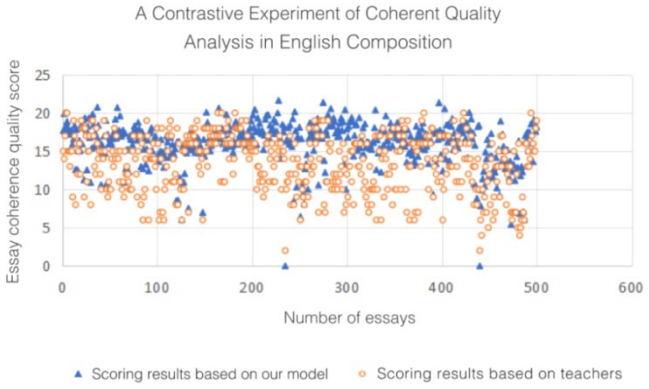


Figure 10. Comparison chart of English composition coherence quality scores

We compared the model ratings of each English composition with the teacher's manual ratings, and the absolute difference in ratings was calculated and recorded. Afterwards, the absolute difference between the ratings of the 500 English essays was averaged, and the mean difference between the model's English essay correction results and the teacher's manual correction results was obtained as 3.2202. This result also indicates that the difference between the model's correction results, and the manual correction results of this paper is not significant. Finally, we calculated the Pearson correlation coefficient between the results of this model and the results of manual correction, and the result was 0.6025, which is a strong correlation in the correlation. In conclusion, the model of this paper is reliable in terms of English composition correction.

5. Conclusion

This paper identifies the direction of exploring the semantic coherence quality analysis of English text in this paper by further analysing and studying the existing mainstream techniques of text coherence. After comparing various

coherence techniques, we finally decide to improve on the entity graph model and incorporate the semantic information between sentences to build the English text coherence quality analysis model based on the sentence semantic graph in this paper.

Although the model in this paper has achieved some good results, it still leaves much to be desired. In terms of representing the text, the sentence semantic graph is still limited in its ability to represent all the semantic information and internal associations of the whole text. If we can analyse the semantic relations and connections between sentences from more perspectives, such as rhetorical structure and contextual information, and use this information to represent the text, it will greatly improve the accuracy of the subsequent analysis of text coherence. In terms of coherence quality analysis, this paper mainly focuses on the frequency of subgraphs with three or four nodes, which is not comprehensive enough. If the frequency of large subgraphs can be analysed and the problem of sparse data in large subgraphs can be solved, then the performance of the model will be further improved. In addition, frequent subgraphs are only one feature of a graph, and the performance of the model would also become better if more appropriate graph features could be used to analyse the graph model.

Acknowledgments.

This work is supported by the National Natural Science Foundation of China (No. 62066009) and the Project of Guilin Key Research and Development (No. 2020010308)

References

[1] Liu S, Zeng S, Li S. Evaluating text coherence at sentence and paragraph levels[J]// Proceedings of the 12th Conference on Language Resources and Evaluation, 2020, 1695-1703.

[2] Chen R, Wang J, Yu L C, et al. Learning to Memorize Entailment and Discourse Relations for Persona-Consistent Dialogues[J]// Proceedings of the AAAI conference on artificial intelligence, 2023.

[3] Mesgar M, B ü cker S, Gurevych I. Dialogue coherence assessment without explicit dialogue act labels[J]// Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, 1439-1450.

[4] Hsiao M, Hung M. Construction of an Artificial Intelligence Writing Model for English Based on Fusion Neural Network Model[J]. Computational Intelligence and Neuroscience, 2022: 2022.

[5] Guinaudeau C, Strube M. Graph-based local coherence modeling[C]// Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2013, 93-103.

[6] Goyal T, Li J J, Durrett G. Snac: Coherence error detection for narrative summarization[J]. arXiv preprint arXiv:2205.09641, 2022.

[7] Ghazarian S, Wen N, Galstyan A, et al. DEAM: Dialogue Coherence Evaluation using AMR-based Semantic Manipulations[J]. arXiv preprint arXiv:2203.09711, 2022.

[8] Papalampidi P, Cao K, Kocisky T. Towards coherent and consistent use of entities in narrative generation[C]// International Conference on Machine Learning. PMLR, 2022, 17278-17294.

[9] Zhao Q, Niu J, Liu X, et al. Generation of Coherent Multi-Sentence Texts with a Coherence Mechanism[J]. Computer Speech & Language, 2023, 78: 101457.

[10] Akula A R, Zhu S C. Discourse Analysis for Evaluating Coherence in Video Paragraph Captions[J]. arXiv e-prints, 2022.

[11] Zhao W, Strube M, Eger S. Discoscore: Evaluating text generation with bert and discourse coherence[J]. arXiv preprint arXiv:2201.11176, 2022.

[12] Pu L, Heng R, Xu B. Language Development for English-Medium Instruction: A Longitudinal Perspective on the Use of Cohesive Devices by Chinese English Majors in Argumentative Writing[J]. Sustainability, 2022, 15(1): 17.

[13] Vrana S R, Bono R S, Konig A, et al. Assessing the coherence of narratives of traumatic events with latent semantic analysis[J]. Psychological Trauma: Theory, Research, Practice, and Policy, 2019, 11(5): 521.

[14] Teng L, Feng Z, Fang X, et al. Unsupervised feature selection with adaptive residual preserving[J]. Neurocomputing, 2019, 367: 259-272.

[15] Yin J, Tang M J, Cao J, et al. Knowledge-Driven Cybersecurity Intelligence: Software Vulnerability Coexploitation Behavior Discovery[J]. IEEE transactions on industrial informatics, 2022, 19(4): 5593-5601.

[16] Li R. A novel image clustering method based on coupled convolutional and graph convolutional network[J]. EAI Endorsed Transactions on Scalable Information Systems, 2022, 9(36): e1-e1.

[17] Gotama P J W , Tokunaga T . Evaluating text coherence based on semantic similarity graph[C]. 2017 Workshop on Graph Based Methods in Natural Language Processing, Annual Meeting of Association for Computational Linguistics (ACL), 2017.

[18] Huang G, Tang H, Wang J, et al. A Coherence Analysis Model for English Essay Based on Sentence Semantic Graph[C]// Journal of Physics: Conference Series. IOP Publishing, 2020, 1693(1): 012077.

[19] You M, Yin J, Wang H, et al. A knowledge graph empowered online learning framework for access control decision-making[J]. World Wide Web, 2023, 26(2): 827-848.

[20] Gao M, Du K J, Zhu P Y, et al. A Robust Two-Part Modeling Strategy for Knowledge Graph Enhanced Recommender Systems[C]//2023 15th International Conference on Advanced Computational Intelligence (ICACI). IEEE, 2023: 1-7.

[21] Hung M, Hsiao M. Application of Adaptive Neural Network Algorithm Model in English Text Analysis[J]. Computational Intelligence and Neuroscience, 2022, 2022.

[22] Srinivas V, Santhirani C. Optimization-based support vector neural network for speaker recognition[J]. The Computer Journal, 2020, 63(1): 151-167.

[23] Teng L, Feng Z, Fang X, et al. Unsupervised feature selection with adaptive residual preserving[J]. Neurocomputing, 2019, 367: 259-272.

[24] Toutanova K, Klein D, Manning C D, et al. Feature-rich part-of-speech tagging with a cyclic dependency network[C]// Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, 2003, 252-259.

[25] Raghunathan K, Lee H, Rangarajan S, et al. A multi-pass sieve for coreference resolution[C]// Proceedings of the 2010 conference on empirical methods in natural language processing, 2010, 492-501.

[26] Lee H, Peirsman Y, Chang A, et al. Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task[C]// Proceedings of the fifteenth conference on

computational natural language learning: Shared task, 2011, 28-34.
[27] Oberoi K S, Del Mondo G, Gaüzère B, et al. Detecting dynamic patterns in dynamic graphs using subgraph isomorphism[J]. Pattern Analysis and Applications, 2023: 1-17.
[28] Born L, Mesgar M, Strube M. Using a Graph-based Coherence Model in Document-Level Machine Translation[C]// Proceedings of the Third Workshop on Discourse in Machine Translation. 2017.
[29] Ishikawa S. The ICNALE Guide: An Introduction to a Learner Corpus Study on Asian Learners' L2 English[M]. Taylor & Francis, 2023.
[30] Hung M, Hsiao M. Application of Adaptive Neural Network Algorithm Model in English Text Analysis[J]. Computational Intelligence and Neuroscience, 2022, 2022.
[31] Huang G, Tan M, Sun Z, et al. RST-based Discourse Coherence Quality Analysis Model for Students' English Essays[C]//MATEC Web of Conferences. EDP Sciences, 2018, 232: 02020.
[32] He Z. Cohesion in academic writing: A comparison of essays in English written by L1 and L2 university students[J]. Theory and Practice in Language Studies, 2020, 10(7): 761-770.