International Journal of Knowledge and Language Processing
Volume 13, Number 1, Dec **2022**
AKLP International ©2010 ISSN 2191-2734
pp. 20–37

Research on Word Embedding Model for Automatic Generation of Novelty Retrieval Expression

Tingting Wang
IMINZU University of China
No. 27, Zhongguancun South Street, Haidian District, 100081
Beijing, China
1041987149@qq.com

Received June 2022; revised Sep 2022

Abstract: With the rapid development of Internet technology, text data is growing exponentially. How to effectively analyze and utilize these data and fully explore the value contained therein is the primary task of text big data analysis and processing. Text representation is an important work in the field of natural language processing, and how to better represent text semantics is an important cornerstone of practical applications in the field of natural language processing. The word embedding vector representation obtained by training can be considered to represent the word itself and its meaning. However, words have different meanings in different contexts, and polysemy is involved in different contexts. How to accurately express the word embedding vector to adapt to different contexts is also a hot topic of current research. At present, word embedding technology is still in its infancy, and there are still many problems worthy of further study. This paper makes a comparative study of the existing word embedding models, and selects the most suitable vector representation model for the automatic generation of novelty search formula to provide technical support for the scientific and technological novelty search system.

Key words: Natural language processing, Word-Embedding, Neural network, Novelty retrieval expression

1. Introduction. With the continuous development of modern society, data is becoming an important asset of individuals and enterprises, and the technology of analyzing and extracting the value of data has become the focus of attention in the era of big data. Users can use the technology of transforming natural language into structured processing to improve the efficiency and speed of mining effective information in structured data assets. At the same time, the application of this technology also lowers the threshold for non-technical users. This paper focuses on the research of Word-Embedding sub-task in the automatic generation technology of novelty search formula under the platform of novelty search system, and intends to select the most suitable model for novelty search formula. Word-Embedding model aims to map words into a vector space, each word has a dense vector of fixed length, similar words have similar vector representations, and polysemy can also be represented as multiple vectors in some models, which can facilitate quantitative text processing. Mining features between words and sentences in the text. The

representation of word vector has changed from the co-occurrence matrix and SVD decomposition based on statistics to the popular neural network based language model.

2. **Research on Word-Embedding Model.** Vector space model has been used in distributed semantics since the 1990s. At that time, many models for predicting word representations in continuous spaces had been developed, including Latent Semantic Analysis (LSA) and Latent Dirichlet Allocation (LDA). Bengio et al. [1] coined the term word embedding in 2003 and trained it jointly with model parameters in a natural language model. Collobert and Weston [2] showed the first practical application of pre-trained word-Embedding in 2008. Their landmark paper, A unified architecture for natural language processing, not only established word embedding as a useful tool for downstream tasks, but also introduced the neural network architecture that is now the basis of many methods.

But it was Mikolov [3-4] et al. Who created word2vec in 2013, a tool suite that allows seamless training and the use of pre-trained word-Embedding, that eventually made word-embedding popular. In 2014, Pennington [5] released a competitive set of pre-trained word-embedding GloVe, marking that word embedding have become mainstream. Word embedding is one of the successful applications of unsupervised learning. Their greatest benefit is undoubtedly that they do not require expensive manual annotation, but are derived from off-the-shelf big data sets that have never been annotated. Pre-trained word embedding can be used in downstream tasks that use only a small amount of annotated data. This paper will introduce the one-hot vector, word2vec, GloVe, ELMo and BERT vector, and analyze them combined with the neural network language model to get an efficient and fast automatic generation model of retrieval.

3. **Word-Embedding model**

One-hot Encoding 3.1. One-hot encoding is also known as one-bit efficient encoding. It is processed by occupying the N-bit status register, but only one of the N bits is valid, that is, 1, and the rest are all 0. This requires mapping the classification values to binary integer values, with each integer value labeled as a binary vector. Detailed explanation of the coding process: first determine the set of coding objects, and then determine the elements in the set. For example, the phrase "hello world" has 27 categories (space+26 lowercase letters). In this example, there are 11 elements, each of which has 27 features, which are converted into a binary vector representation as shown in Figure 1. (1) The 27 features are first integer encoded: a--0, B--1, C--2.., Z--25, space--26, (2) The 27 features are arranged from front to back according to the size of the integer code.

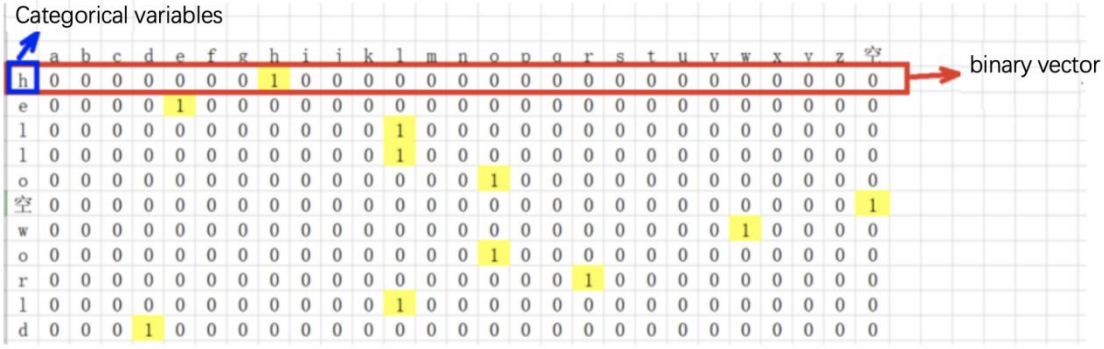

FIGURE 1. One-hot Encoding

One-hot encoding is intuitive, with each element corresponding to only one feature, making it easy to calculate the loss function or accuracy. It also has two serious disadvantages: (1) The length of the dictionary is the length of the representation of each dimensional vector, and only one position in each dimensional vector is 1, which wastes memory and is not conducive to calculation. (2) One-hot coding is a bag-of-words model, whose vector matrix is equivalent to numbering each word, and the resulting features are sparse and discrete. However, the words in the text are related to each other, which does not take into account the semantic problems of the co One-hot converts words into discrete individual symbols, and the hidden layer is a linear unit without activation functions. Using Soft-max Function for normalization, the dimensions of the output layer are the same as those of the input layer. Word2vec was proposed after the emergence of one-hot. The Purpose of Word2vec is to transform natural language into Dense Vector, and to transform sentences and words in natural language into Dense Vector through the model of Continues Bag of Words. The two models of CBOW and Skip-gram are transformed into dense vectors that computers can understand and calculate, and the dense vectors we want to obtain are actually the output units of the hidden layer. Word2vec is an intermediate result (weight matrix) generated by neural network learning, and its downstream task is to serve as the corpus input of neural probabilistic language model. Hierarchical Soft-max or Negative Sampling is used in the neural network learning process to reduce complexity. The CBOW/skip-gram model is shown in Figure 2.

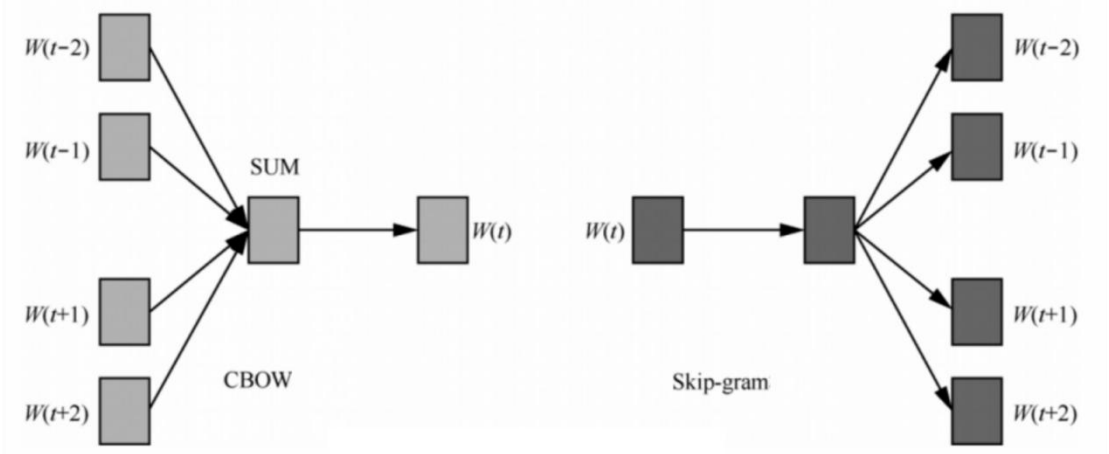

FIGURE 2. CBOW/Skip-gram model

In the figure, the former word represents the word, and the latter word represents the word. The running window of the CBOW model in the figure 2, which is predicted by the upper and lower words of the target word. The prediction process is as follows: first, the dimension of the word vector is set, here for example, dimension d, and then all words are randomly initialized as the word vector with the set dimension d. Then the vector of the target word is predicted by the hidden layer vector obtained by encoding the word vectors of the two words, and above and below the target word. Finally, it can be seen from the figure that in the bag-of-words model, he simply adds the vectors of the predicted target words, and then does Soft-max

normalization to get the target words. For example, if there are V words in a vocabulary set, the D-dimensional hidden layer vector is multiplied by the W matrix and converted into a V-dimensional vector for soft-max classification. Generally, the number of vocabulary is very large, and the number of occurrences of different words varies greatly. In the experiment, soft-max and negative sampling are generally used to optimize the problem of large amount of calculation.

The Skip-gram model predicts the number of words in the window size of the context through the target word, that is, the target word is input and mapped to the hidden layer vector, and the upper and lower words of the target word are predicted according to the vector of the target words. The problem of unbalanced vocabularies and samples is handled in the same way as CBOW.

GloVe model 3.2. GloVe model is a Word2vec model that integrates the overall information, and it can also be considered as an improved Word2vec model. From the previous chapter, it can be seen that CBOW and Skip-gram models only consider the local information of words in the window, but lose the semantic information of words outside the window, so GloVe uses co-occurrence matrix to consider the fusion of local information and global information. The title of this thesis is used to introduce the running window of GloVe. The size of the window is set to 5, that is, the two words before and after the head word.

TABLE 1. Window content

Window label	Center words	Window content
0	Research on	Research on, Word-Embedding, Model
1	Word-Embedding	Research on, Word-Embedding, Model, for
2	Model	Research on, Word-Embedding, Model, for, Automatic
3	for	Word-Embedding, Model, for, Automatic, Generation
4	Automatic	Model, for, Automatic, Generation, of
5	Generation	for, Automatic, Generation, of, Novelty
6	of	Automatic, Generation, of, Novelty,Retrieval
7	Novelty	Generation, of, Novelty,Retrieval, Expression
8	Retrieval	of, Novelty,Retrieval, Expression
9	Expression	Novelty,Retrieval, Expression

In order to better express the contents in the table with vectors, the following symbols are used to analyze the rules:

$X_{i,j}$: Obtained by traversing all the window content statistics, $X_{i,j}$ representing the number of times J appears in the window content with I as the central word.

X_i: Count how many other words the word I is the head of. $X_i = \sum_{j=1}^{N} X_{i,j}$.

$P_{i,k}$: The proportion of word K near word I to all nearby words of word I. $P_{i,k} = \frac{X_{i,k}}{X_i}$.

$Ratio_{i,j,k}$: proportion of the proportion of word K occurring near word I and word

J,$Ratio_{i,j,k}=\frac{P_{i,k}}{P_{j,k}}$.

The following table is obtained by statistical analysis of the values of $Ratio_{i,j,k}$:

TABLE 2. $Ratio_{i,j,k}$ Correlation scores with words

$Ratio_{i,j,k}$	Words J, K related	Words J, K not related
Word I, K correlation	approaches 1	Very large
Word I, K correlation	Very small	approaches 1

ELMo Model 2.4. Although GloVe incorporates local information and global information, it has not yet solved an existing problem, that is, polysemy. Words have different meanings in different sentences. For example, "bank" can refer to either a bank or a riverbank. In Chinese, it is even more so. For example, "apple" can refer to either a brand in electronic products, a mobile phone, a tablet or a computer, or an apple in the fruit world. In the GloVe model, these two words represent the same vector in different contexts. ELMo [6] solved this problem by extracting semantic and syntactic features from LSTM.

$$P(t_1, t_2, \bullet\bullet\bullet, t_N) = \prod_{k=1}^{N} p(t_k / t_1, t_2, \bullet\bullet\bullet, t_{k-1}) \quad (1)$$

$$P(t_1, t_2, \bullet\bullet\bullet, t_N = \prod_{k=1}^{N} p(t_k / t_{k+1}, t_{k+2}, \bullet\bullet\bullet, t_N) \quad (2)$$

In the forward process of Formula 1, the word from the first word to the (K−1)th word is used to predict the kth word, and in the backward process of Formula 2, the word from the (K+1)th word to the Nth word is used to predict the kth word in the reverse direction. ELMo predicts the target word through a double-layer bidirectional recurrent neural network RNN, as shown in fig. 3, where E represents the input of word features. The language model is trained by extracting syntactic features and semantic features using two-layer LSTM, and then the regression processing is performed using softmax.

Specifically, words are predicted in both directions, and the specific encoding method can be expressed by LSTM as:

$$\sum_{k=1}^{N} (logp(t_k / t_1, t_2, \bullet\bullet\bullet, t_{k-1}; \Theta_x, \overrightarrow{\Theta_{LSTM}}, \Theta_s) + (logp(t_k / t_{t+1}, t_{k+2}, \bullet\bullet\bullet, t_N; \Theta_x, \overleftarrow{\Theta_{LSTM}}, \Theta_s)) \quad (3)$$

Θ_x is the vector representation of Token and Θ_s is the parameter of softmax.

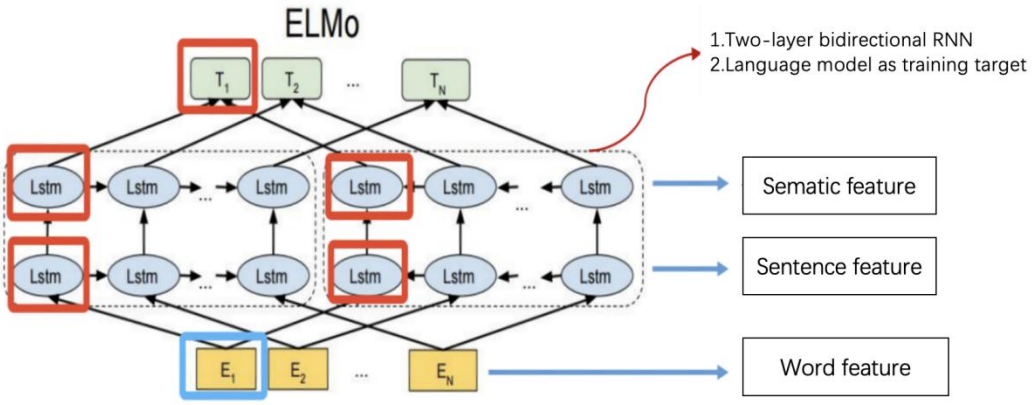

FIGURE 3. ELMo model

Bidirectional LSTM concatenates multiple layers, and the vector output of each hidden layer is used as the vector input of the next layer. For the word E_1, the two-layer bidirectional LSTM language model as shown in Figure 3 can obtain two vector representations at each layer, and then +1 is the vector representation of the Token output layer, and the five vectors are combined to obtain the vector representation of each Token of ELMo. Then for the kth word, each Token will have 2L+1 vector representations.

$$\mathrm{ELMo}_k^{task} = E(R_x;\Theta^{task}) = r^{task}\sum_{j=0}^{L} s_j^{task} h_{k,j}^{LM} \tag{4}$$

$h_{k,j}^{LM}$ is the hidden layer output for the jth layer of Token K, and represents the vector representation of the input layer Token when J = 0, s_j^{task} is a normalization of soft-max and r^{task} is a scaling parameter, representing the proportion of each feature, which allows the target model to scale the vector of ELMo. ELMo solves the problem of polysemy, and the part of speech is also corresponding, but it still has two problems: ELMo's ability of using LSTM as language model to extract features is weaker than Transformer proposed by Google; LSTM, even BiLSTM, only considers unilateral data and sums them at the loss function, which is weak in the ability of bidirectional fusion of features. As a sequential model, the parallel computing ability is poor.

BERT model 2.5. BERT [7] has a wide range of applications, and has designed extremely simple downstream interfaces for sequence labeling, classification tasks, and sentence relation judgment tasks in the four major tasks of natural language processing. Figure 4 below is the interface designed for the downstream task in the BERT paper. BERT can be regarded as a black box, which works in a similar way to ELMo. It is a Pre-trained in the current large-scale corpus, and then the downstream task is input to compare the lightweight Fine-tuning. Downstream tasks can be put into BERT's Fine-tuning model with some slight modifications. It can be seen that for sentence-relational tasks, as long as the start and end symbols of sentences are added,

and the division symbols are added between sentences. Then the output of the last position of the BERT model is connected to a softmax classifier. For the model of sequence labeling, it is also enough to add the start and end symbols, and add a linear classifier to the output of each position of the final BERT.

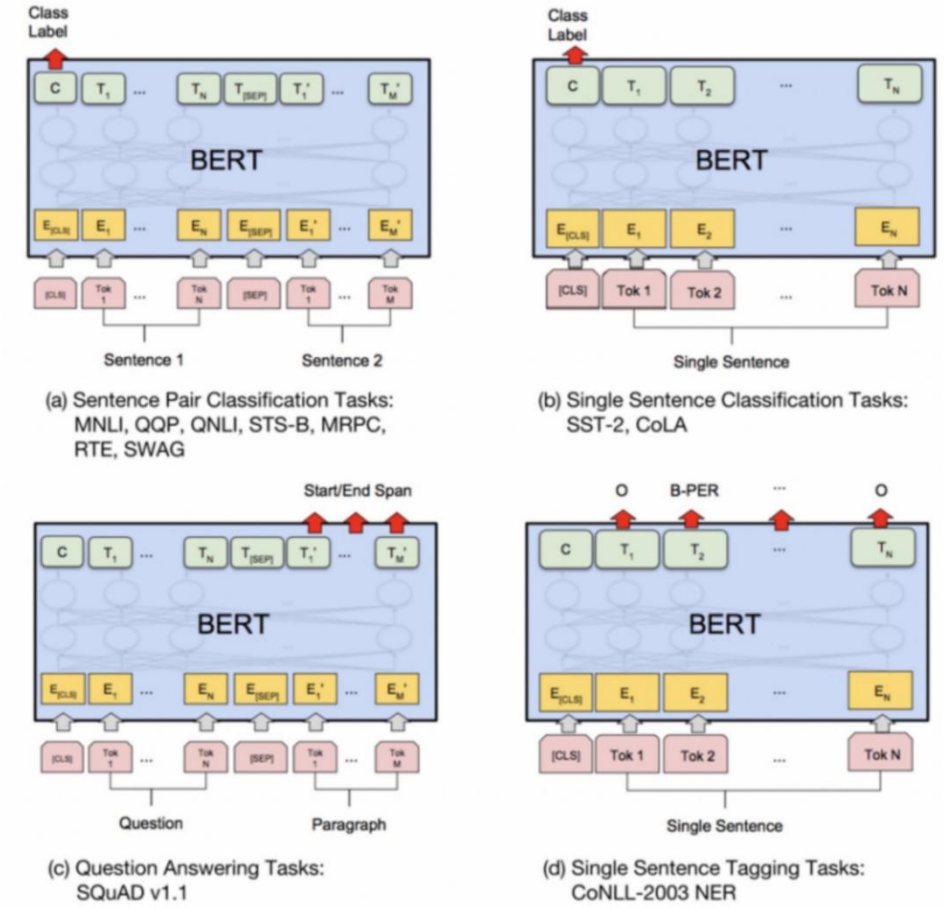

FIGURE 4. BERT model

The subsequent BERT uses Transformer for encoding. The whole Transformer structure is divided into two parts as shown in Figure 5 below. One side is Encoder, and the other side is Decoder. The reason why Transformer will replace recurrent neural network and convolution neural network as the mainstream coding method of natural language processing is that convolution neural network extracts local features, but for text data, it ignores long-distance dependence, and the coding ability of CNN in text is weaker than that of RNN, while RNN is a sequence model with poor parallelism. It is slow to compute and can only consider information in one direction. Transformer can comprehensively consider the information in two directions, and comprehensively consider the context features in both directions when predicting words.

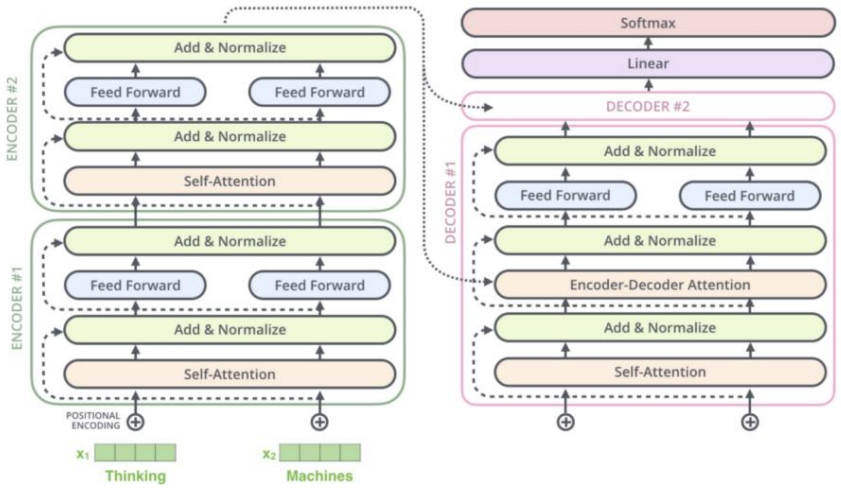

FIGURE 5. Transformer model

By looking at the block ENCODER # 1, for example, the input is a vector of a line of sentences with its position vector encoded and then passed through a Self-Attention. Self-Attention first maps the embedding of each Token to three vectors, Query (Q), Key (K), and Value (V), through three different matrices. Figure 6 below assumes that the length of the sentence is only 2 tokens. Since the three W conversion matrices of Q, K, and V share parameters at different positions, matrix operations can be used. The Q and K vectors are used to calculate the weight parameters of Attention. The calculated weight parameters are multiplied by the V vector, and then weighted and summed to obtain the vector of each token. The calculation process is as follows: For example, to calculate the vector of position 1, first, Q_1 is multiplied by K_1 to K_n points to obtain n scalars. This scalar is divided by $\sqrt{d^k}$ and d^k is the dimension of the vector of K. After softmax normalization processing, the normalized weight is correspondingly multiplied by V_1 to V_n, and the Token vector of position 1 is obtained by summation.

Self-Attention will also superimpose H independent Self-Attention mechanisms in parallel. The Token at each position will have H vectors. The H vectors will be spliced and a linear transformation will be made to get the final vector. H is called the head number of Attention. Because this Self-Attention structure will ignore the location information, unlike CNN and RNN, which will naturally encode the location information. The location information is very important for the sequence text. Add the location code at the output, and then make a standardization, corresponding to Add & Normalize in Figure 5. The position code here is calculated as follows:

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{model}}) \tag{5}$$

$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{model}}) \tag{6}$$

The position of each POS is represented by a d-dimensional vector, the even-numbered position of the vector is calculated by a sine sin function, the odd-numbered position is calculated by a cosine cos function, and a value between -1 and 1 is obtained; the trigonometric function is used because the characteristic that the

sum and difference transformation of the trigonometric function can be linearly transformed is utilized, After a Feed Forward neural network and Add & Normalize, an encoding is completed and passed to the next encoding block. The structure and operation mechanism of Decoder are basically the same as those of Encoder. The difference is that the K and V vectors at the top level of Encoder will be passed to the Encoder-Decoder Attention component of each Block of Decoder. Finally, the final result is obtained through a linear transformation and the softmax classifier.

3. **Neural network language model.** The essence of deep learning lies in vector representation, but natural language processing has not done the most fundamental representation learning---Word-Embedding and language model well, and no model can achieve excellent results alone. Combining the existing models as functional modules to construct a more complex model is the mainstream idea to solve the problem of automatic generation of novelty retrieval.

From the beginning of BP neural network, neural network model began to get a lot of attention. Convolution Neural Network (CNN) is an efficient feature extraction method developed in recent years and has made significant breakthroughs in many tasks. CNN is a convolution neural network, which is used for image processing and extracting high-latitude features. Then Recurrent Neural Network [9] (RNN) appears. RNN has the problem of dimension explosion and gradient disappearance. Therefore, the Long Short-Term Memory (LSTM) model proposed by Hochreiter and Schmidhuber et al. [10] is improved. In 2014, K. Cho [11] proposed a gated recurrent unit network (GRU), which is another RNN gating architecture that has attracted attention after LSTM. GRU is a variant of LSTM with fast speed. In 2013-2015, Y. Bahdanau, etc. [12] proposed a series of RNN algorithms such as encoder-decoder and self-attention layer, and applied them to machine translation problems [13]. Seq2seq is used to solve the problem of unfixed length of input and output, and attention mechanism is used to solve the problem of information overload.

The Seq2seq model is also known as the Encoder-Decoder model. In deep learning, the encoder-decoder framework is a classical scheme to deal with the problem of sequence-to-sequence mapping. Another output sequence y is generated from one input sequence X. Seq2seq model can be applied in many aspects, such as machine translation, document extraction, question answering system and so on. The encoder is to encode the input sequence into a vector of fixed length, and the decoder is to convert this fixed vector into a sequence output again. The Encoder-Decoder model diagram is shown in Figure 6, where the decoder and encoder are not fixed, and the optional models are CNN/RNN/Bi-RNN/GRU/LSTM, etc. Encoder and Decoder can be freely combined from these models.

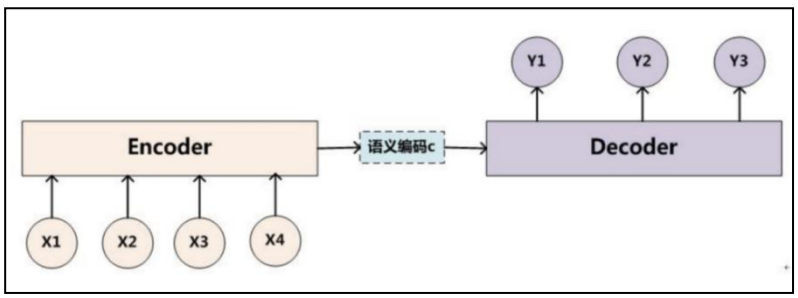

FIGURE 6. Encoder-Decoder architecture diagram

The loss function during the training of the Encoder-Decoder model generally uses the cross-entropy loss, that is, to maximize the probability of generating the annotated text in the training data, and $P(y_{i,k}) | y_{(1,2,\cdots,i-1)})$ represents the probability of Decoder generating a word with an ID of K in the ith step, so given an annotated text, the logarithmic probability of being generated by the model is:

$$P(y_1, \ldots, y_m) = -\log \prod_{i=1}^{m} p(y_{i,k}) = \sum_{i=1}^{m} \log (y_{i,k}) \tag{7}$$

Encoder and Decoder commonly use RNN-like units such as Vanilla RNN, GRU, and LSTM. The RNN receives the current input and the hidden state from the previous input at each step through the self-loop, that is, the hidden state at the current time is determined by the state at the previous time and the current time input, which is expressed by the formula:

$$h_t = f\ (h_{t-1}, h_t) \tag{8}$$

The information of each hidden layer is combined to generate a semantic vector of

$$C = q\ (h_1, h_2, h_3, \ldots h_{Tx}) \tag{9}$$

A simple way is to take the last hidden layer as the semantic vector C, which is

$$C = q\ (h_1, h_2, h_3, \ldots h_{Tx}) = h_{Tx} \tag{10}$$

Decoding is done by combining a given semantic vector C with the output sequence $y_1, y_2, y_3, \ldots y_t$ to predict the output of the next word y_{t-1}, which is

$$y_t = argmax P\ (y_t) = \prod_{t=1}^{T} P\ (y_t | \{y_1, \ldots, y_{t-1}\}, C) \tag{11}$$

In the formula, C represents the semantic vector, and yt-1 is the output representation of the previous time period, which will be used as the input at this time again.

Although the Encoder-Decoder model is classical, its limitation is that encoding and decoding are only related by a fixed-length semantic vector C, which has two defects. One is that the fixed-length semantic vector can not fully express the input information, and the other is that the content input first will be covered and diluted by the content input later.

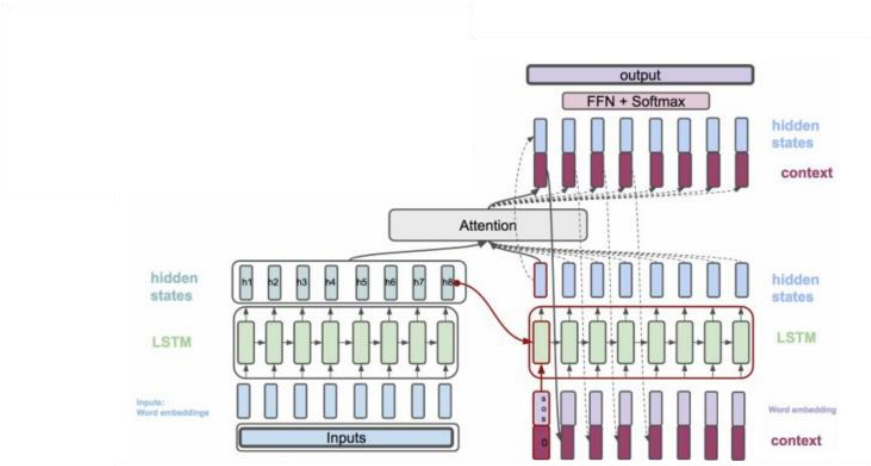

FIGURE 7. Seq2seq attention architecture diagram

In the process of generating search formula for novelty search, it is necessary to summarize and analyze the generation strategy of search formula for novelty search, the text vector representation model and the text generation model. This chapter introduces the word embedding model and the generation model used in the automatic generation model of search formula, which is intended to provide technical support for the realization of automatic generation function of search formula, and provide reference for the improvement of some models in this paper.

4. **Characteristic acquisition of scientific and technological literature**

Literature acquisition and preprocessing 4.1. Due to the scarcity of the orders provided to the novelty retrieval system platform at present, it can not provide sufficient data for this experiment, so the abstract of scientific literature is selected as the novelty retrieval point. In this paper, the scientific literature in the fields of computer, information science, chemistry and pharmacy are selected as the experimental objects, and the experimental data are retrieved from Wan-Fang Database Knowledge Service Platform. The retrieval strategy is to limit the abstract structure to "purpose: method: result and conclusion", and the time is from 2015 to 2020. The specific number of relevant documents retrieved is shown in Figure 8.

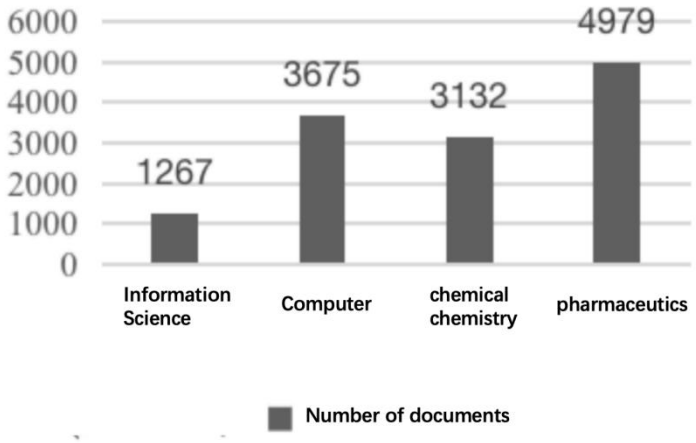

FIGURE 8. Distribution of Literature Quantity

It can be seen from the figure that the number of literature in the field of pharmacy

is the largest, and the number of literature in the field of information science is the smallest. In order to obtain the literature source of the specification, remove the irrelevant literature. It is necessary to preprocess the acquired scientific and technological literature. The specific process is shown in the figure below. Firstly, data collection and preprocessing are carried out, including merging duplicate items, deleting missing items, and extracting "title" and "abstract" fields to form a corpus to be analyzed and processed. Then the corpus is processed by word segmentation, stop word filtering and other steps to form the data set needed for the final experiment.

Text word segmentation is to generate a sequence feature document by processing the text. Chinese word segmentation is a unique concept in the process of Chinese processing. English text has obvious spaces to distinguish the boundaries between words, while the boundaries between words in Chinese text are not just simple punctuation marks. For this reason, the Chinese word segmentation is an important step in the preprocessing, but the Chinese word segmentation technology is not complicated. The following figure shows the word segmentation results of processing the text of the novelty search point using the appropriate word segmentation model.

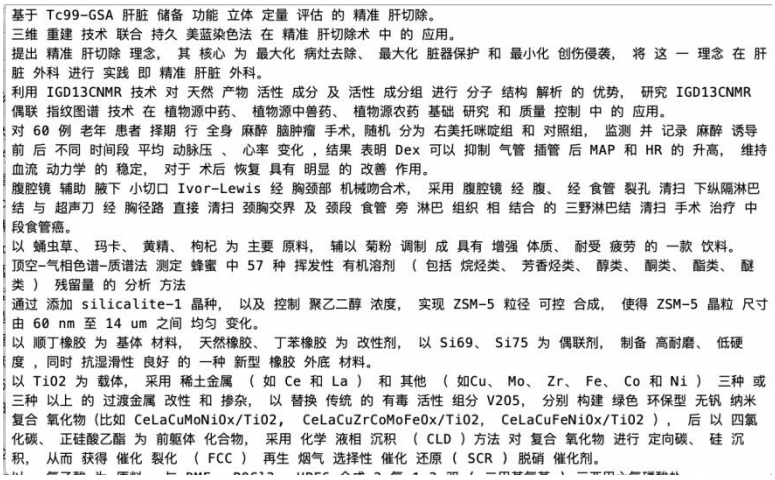
基于 Tc99-GSA 肝脏 储备 功能 立体 定量 评估 的 精准 肝切除。
三维 重建 技术 联合 持久 美蓝染色法 在 精准 肝切除术 中 的 应用。
提出 精准 肝切除 理念， 其 核心 为 最大化 病灶去除、 最大化 脏器保护 和 最小化 创伤侵袭， 将 这 一 理念 在 肝脏 外科 进行 实践 即 精准 肝脏 外科。
利用 IGD13CNMR 技术 对 天然 产物 活性 成分 及 活性 成分组 进行 分子 结构 解析 的 优势， 研究 IGD13CNMR 偶联 指纹图谱 技术 在 植物源中药、 植物源中兽药、 植物源农药 基础 研究 和 质量 控制 中 的 应用。
对 60 例 老年 患者 择期 行 全身 麻醉 脑肿瘤 手术，随机 分为 右美托咪啶组 和 对照组， 监测 并 记录 麻醉 诱导 前 后 不同 时间段 平均 动脉压 、 心率 变化 ，结果 表明 Dex 可以 抑制 气管 插管 后 MAP 和 HR 的 升高， 维持 血流 动力学 的 稳定， 对于 术后 恢复 具有 明显 的 改善 作用。
腹腔镜 辅助 腋下 小切口 Ivor-Lewis 经 胸颈部 机械吻合术， 采用 腹腔镜 经 腹、 经 食管 裂孔 清扫 下纵隔淋巴结 与 超声刀 经 胸径路 直接 清扫 颈胸交界 及 颈段 食管 旁 淋巴 组织 相 结合 的 三野淋巴结 清扫 手术 治疗 中段食管癌。
以 蛹虫草、 玛卡、 黄精、 枸杞 为 主要 原料， 辅以 菊粉 调制 成 具有 增强 体质、 耐受 疲劳 的 一款 饮料。
顶空-气相色谱-质谱法 测定 蜂蜜 中 57 种 挥发性 有机溶剂 （ 包括 烷烃类、 芳香烃类、 醇类、 酮类、 酯类、 醚类 ） 残留量 的 分析 方法
通过 添加 silicalite-1 晶种， 以及 控制 聚乙二醇 浓度， 实现 ZSM-5 粒径 可控 合成， 使得 ZSM-5 晶粒 尺寸 由 60 nm 至 14 um 之间 均匀 变化。
以 顺丁橡胶 为 基体 材料， 天然橡胶、 丁苯橡胶 为 改性剂， 以 Si69、 Si75 为 偶联剂， 制备 高耐磨、 低硬度 ，同时 抗湿滑性 良好 的 一种 新型 橡胶 外底 材料。
以 TiO2 为 载体， 采用 稀土金属 （ 如 Ce 和 La ） 和 其他 （ 如Cu、 Mo、 Zr、 Fe、 Co 和 Ni ） 三种 或 三种 以上 的 过渡金属 改性 和 掺杂， 以 替换 传统 的 有毒 活性 组分 V2O5， 分别 构建 绿色 环保型 无钒 纳米 复合 氧化物（比如 CeLaCuMoNiOx/TiO2， CeLaCuZrCoMoFeOx/TiO2， CeLaCuFeNiOx/TiO2 ）， 后 以 四氯化碳、 正硅酸乙酯 为 前驱体 化合物， 采用 化学 液相 沉积 （ CLD ）方法 对 复合 氧化物 进行 定向碳、 硅 沉积， 从而 获得 催化 裂化 （ FCC ） 再生 烟气 选择性 催化 还原 （ SCR ）脱硝 催化剂。

FIGURE 9. Example of Text Segmentation

Stop word filtering is also a key step in text processing, deleting words such as "joint", "determination", "based on" that have no clear meaning but will always appear. These words are essential in the process of text reading and comprehension, but in the process of building the model, it has been assumed that words are interchangeable, so these words are of no value to the model of this paper. During the experiment, the text is processed by using the stop word list defined by ourselves. The following figure is an example of matching results using regular expressions after removing stop words.

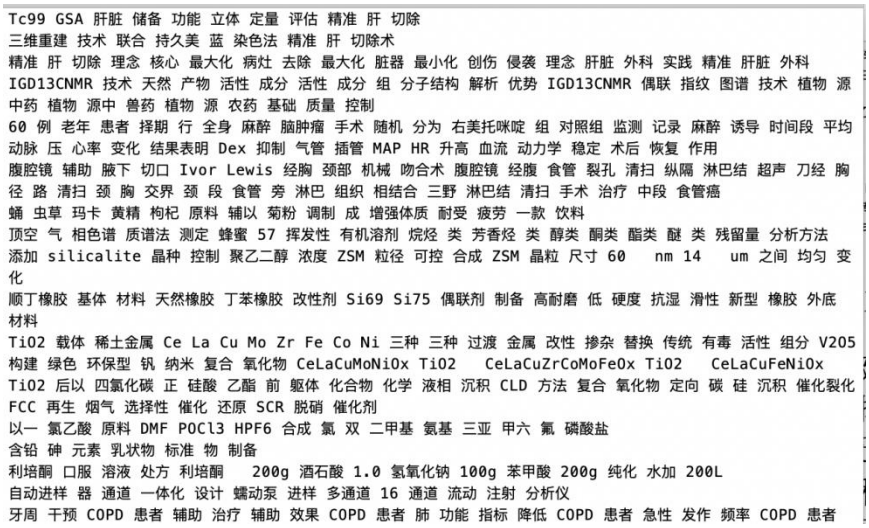

FIGURE 10. an example of a text removal stop word.

Feature extraction experiment 4.2.

One-hot encoding 4.2.1. One-hot coding is simple and easy to use, and is a commonly used method in the text feature extraction process. The flow of the extraction process is as follows: Firstly, each sentence in the corpus is divided into words and numbered, and the words in each sentence are matched with the numbered words, if they are matched, they are 1, and if they are not matched, they are 0.

Word2vec feature extraction 4.2.2. The process of Word2vec feature extraction is shown in the figure below:

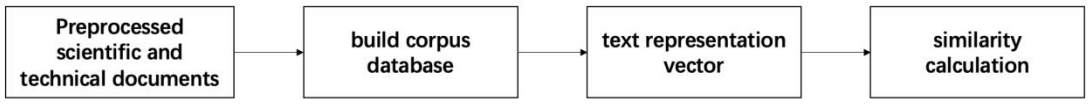

FIGURE 11. Word2vec feature extraction flow

(1) Obtaining the preprocessed literature, searching data according to the pharmaceutical field selected in the experiment, and constructing a text corpus;

(2) The Word2vec model of the training corpus, in the training process, the parameters are set as vector_size, that is, the dimension of the feature vector is 300, min_count, the minimum word frequency of the words participating in the training is 1, windows_size means that the size of the training window is set as 15, and DM training algorithm is set as 1. The number of ITER iterations is set to 500.

(3) Vectorized representation of text, that is, a word is represented as a vector.

(4) Calculate the cosine values of the two vectors. The range of the cosine values is [−1, 1]. A value approaching 1 means that the two vectors are close in direction; a value approaching − 1 means that the two vectors are opposite in direction; and a value approaching 0 means that they are nearly orthogonal.

GloVe feature extraction 4.2.3. The purpose of GloVe is to make the words vector contain the whole semantic information as much as possible. The process of feature extraction is firstly to construct the co-occurrence matrix based on the dictionary, and then to operate the vector based on the co-occurrence matrix by using Euclidean

distance or cosine similarity, and to construct the loss function. The input is a corpus and the output is a word vector. The process is shown in the following figure:

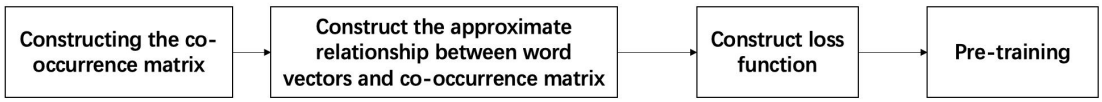

FIGURE 12. Flow chart of GloVe feature extraction

In the experiment, we need to use Linux or OS system. First, we need to download the GloVe code. The "src" folder contains four main running files. By running these four files, we can generate the vectors we need. Firstly, the vocab_count. C is executed, and a dictionary, i.e., a vocab. Txt, is generated by traversing the corpus; the "cooccur.c" file is executed, and a co-occurrence matrix is established from the corpus. It can be seen from fig.14 that the training of GloVe is performed on the basis of the co-occurrence matrix; Then, a shuffle. C is executed, and the previously generated generation matrix is disrupted through the file, namely, the order of the triples is disrupted; and finally, a glove. C training word vector is executed. Before training, you need to make and compile it, and then go through the above steps.

ELMo feature extraction 4.2.4. The process of ELMo feature extraction is as follows: (1) Learn the complex characteristics of word usage through multi-layer LSTM. (2) Through pre-train + Fine tuning, pre-train is performed on a large corpus first, and then fine tuning is performed on the corpus of the downstream task, so that the changes of these complex usages in different contexts can be learned.

BERT Feature Extraction 4.2.5. The initial word vector of each word in the text is used as the input of the BERT model. The vector can be pre-trained by word embedding model as an initial value, and can also be randomly initialized.Each character or word in the text is converted into a one-dimensional initial word vector by querying the word vector through the vocabulary, and the text vector (the value of the vector is automatically learned in the model training process, which is used to characterize the global semantic information of the text and is integrated with the semantic information of a single character/word) and the position vector are extracted. Add the three vectors according to Attention weight as the input of the model, introduce Msaked LM, use the two-way language model for pre-training, and then solve the downstream task through the Fine-tuning mode. The output of BERT model is the vector representation matrix of each character or word, which integrates the semantic information of the full text.

Comparison results of word embedding model 4.3. The scientific literature and the corresponding novelty search formula needed in the process of generating search formula based on antagonistic learning are preprocessed, and the structure, advantages and disadvantages of the word vector embedding model are compared and summarized in Table 3. Through the comparative study, it is found that the automatic construction of novelty search formula through BERT model can solve the problem of polysemy of novelty search formula. The semantic similarity is calculated by cosine similarity, which makes the novelty retrieval more accurate. In the course of the study, it is found that the Bert model also has defects. The vectors generated by the Bert model are dynamic and not interpretable, resulting in the results can not be

reproduced.

TABLE 3 Comparison of Word Vector Model

Model	Model structure	Information	Advantages	shortcomings
One-hot	N-bit status register	One-bit effective coding	The coding method is intuitive, and each element corresponds to a feature, which is convenient for calculating the loss function and accuracy.	It wastes memory, is not conducive to computation, has sparse features, and does not consider the semantic relationship of context.
Word2vec	CBOW/ skip-gram	The target word is input and mapped to the hidden layer to predict the probability of the word.	Fixed size left and right window words can be mapped to dense vector	The semantic information of the words outside the window is lost.
GloVe	Optimization of Word2vec	Word2vec model with integrated information	The co-occurrence matrix is used to consider the fusion of local and global information.	Does not address the existing phenomenon of polysemy.
ELMo	Multilayer LSTM	The pre-trained BiLSTM is used to generate word embedding, which is provided to the downstream model	Semantic and syntactic features are extracted through LSTM to solve polysemy and part of speech correspondence.	The extraction ability of LSTM is weaker than that of Transformer; even BiLSTM only considers unilateral data, so its fusion feature ability is weak, and its parallel computing ability is poor.
BERT	Bidirectional Transformer	Masked LM and Next Sentence Prediction are used to capture word-level and sentence-level representations , respectively	11 NLP tasks were pre-trained and fine -tuned to capture longer range dependencies.	Only 15% of tokens in each batch are predicted, and the convergence speed is slow.

5. **Conclusions.** Ten thousand abstracts of dissertations, journal papers and conference papers were downloaded from Wan-fang database, and the training set and

test set were made according to the ratio of 9:1. When applying the retrieval formula generation process to train the retrieval expression generation network, the method of antagonistic learning is used to solve the problem of insufficient training samples. Through the training based on BERT and antagonistic learning model, it is found that the same retrieval formula as the real data can be generated. When the generator and the discriminator in the antagonistic learning model reach Nash equilibrium, the effect is optimal, and the same search formula as the standard set can be output, and when the training times are increased, the number of search formulas generated by the model which are the same as those written by experts does not increase. In this paper, the search terms in the generated search formula are matched through the domain word list and the concept synonym list, and the relevant contents in the generated search formula are recommended for scientific researchers to choose. The following is the search formula generated by the input novelty search point through the BERT model and the antagonistic learning model, as shown in Table 4.

TABLE 4. Partial display of generated results

Novelty check point	Retrieval Expression
Purpose To investigate the clinical efficacy of hepatectomy in different extent for hepatolithiasis.MethodEighty-six patients with intrahepatic bile duct stones were treated with surgical operations (left lateral lobectomy, left hemihepatectomy, right hemihepatectomy and hepatic segmentectomy), and the clinical effects of different hepatectomies were compared.Results The residual stone rate of left lateral lobectomy was significantly higher than that of left hemihepatectomy, right hemihepatectomy and segmentectomy ($P < 0.05$ or $P < 0.01$). The complication rate of right hemihepatectomy and segmentectomy was higher than that of left lateral lobectomy and left hemihepatectomy ($P < 0.05$). The excellent and good rate was 81. 25% (65/80).The excellent and good rate of left lateral lobectomy was higher than that of left hemihepatectomy, and the difference was statistically significant ($P < 0.05$).Conclusion Hepatectomy is the most effective method for the treatment of intrahepatic bile duct stones. For intrahepatic stones not confined to the left lateral lobe, left and right hemihepatectomy and segmentectomy are superior to left lateral lobectomy. The extent of hepatectomy is closely related to the residual stones after operation and the effect of surgical treatment.	Hepatectomy and hepatolith and (left lateral lobectomy or left hemihepatectomy or segmentectomy) and residual rate and (extent of hepatectomy or residual stones after operation or segmentectomy)
Objective To investigate the clinical value of laparoscopic hepatectomy combined with mini-incision hepatectomy. Methods The clinical data of 13 patients who underwent laparoscopic hepatectomy combined with small incision hepatectomy in the Department of Hepatobiliary Surgery of Chongqing Emergency Center from June to December 2017	Hepatectomy and (liver or haloperidol) and primary liver cancer and liver inflammatory

were retrospectively analyzed. There were 10 cases of primary liver cancer, 2 cases of hepatolithiasis and 1 case of hepatic inflammatory pseudotumor. Results The operation was successfully completed in all the 13 patients, and the operation time was 120270 minutes, with an average of (176. 92 ± 51. 38) min; The mean blood loss was (146.15 ± 100.96) ml (50400 ml). One day after the operation, they could get out of bed and resume liquid diet. The mean postoperative hospital stay was (6.30 ± 2.14) days (410 days). There was no death and no operative complication. Conclusion Laparoscopic hepatectomy combined with mini-incision hepatectomy has the advantages of minimal invasion, quick recovery and simple operation, which can be used as an option for clinical hepatectomy and conversion to open surgery during laparoscopic or robotic hepatectomy.	pseudotumor and laparoscopic combined with small incision hepatectomy and (clinical hepatectomy or laparoscopic or robotic hepatectomy)
Purpose To observe the effect of rosiglitazone on premature ovarian failure induced by cyclophosphamide (CTX) in patients with systemic lupus erythematosus (SLE). Methods Eight patients with systemic lupus erythematosus were treated with rosiglitazone and treated with cyclophosphamide, and other secondary factors were excluded. Sex hormone levels (FSH, E2) and disease activity index score (SLEDAI) were detected before treatment, during menstruation, one month and three months after menstruation. Results Menstruation returned to normal in 6 patients after 4 to 6 days of rosiglitazone treatment, in 1 patient after 2 months of continuous rosiglitazone treatment, and in 1 patient without recovery of menstruation. No increase in SLEDAI. Conclusion Rosiglitazone is effective in the treatment of SLE with secondary premature ovarian failure caused by cyclophosphamide, without adverse reactions and adverse effects on the disease.	((Systemic lupus erythematosus or SLE) and tic disorder and (cyclophosphamide or secondary factor or rosiglitazone) and ((adverse reaction or SLEDAI) or menstruation)

Novelty search is very important for novelty search workers.In the era of "big data", the content contained in the Internet is blowout growth, the amount of information is miscellaneous, and the massive data sources greatly increase the burden of scientific and technological workers. Therefore, according to the characteristics of sci-tech novelty retrieval texts, this paper optimizes the process of sci-tech novelty retrieval business, and on this basis, uses the combination of word embedding model and antagonistic learning to generate a comprehensive novelty retrieval formula.

It is found that the method based on antagonistic learning can automatically generate search formulas and provide technical support for novelty retrieval system. It realizes the structured analysis technology of novelty search point text and the automatic generation technology of novelty search formula, so as to help scientific researchers quickly understand the content of the novelty search project and further improve work efficiency.

Acknowledgment. This work is partially supported by Mr Liu. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] Bengio, Y., Ducharme, R., Vincent, P., & Janvin, C. (2003). A Neural Probabilistic Language Model. The Journal of Machine Learning Research, 2003:1137-1155.

[2] Collobert, R., & Weston, J. A unified architecture for natural language processing. Proceedings of the 25th International Conference on Machine Learning - ICML 2008, 20(1), 160-167.

[3] Mikolov, T., Corrado, G., Chen, K., & Dean, J. Efficient Estimation of Word Representations in Vector Space. Proceedings of the International Conference on Learning Representations ICLR 2013:1-12.

[4] Mikolov, T., Chen, K., Corrado, G., & Dean, J. Distributed Representations of Words and Phrases and their Compositionality. NIPS,2013:1-9.

[5] Pennington, J., Socher, R., & Manning, C. D. Glove: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing.2014:1532–1543.

[6] Peters M,Neumann M,Iyyer M,et a1. Deep contextuaIized word representations[C]. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics:Human Language Technologies,Volume1(Long Papers). New Orleans,2018: 2227-2237.

[7] Devlin J,Chang M W,Lee K,et al. BERT: pre-training of deep bidirectional transformers for language understanding[C]. Proceedings of the 2019 Conference of the Association for Computational Linguistics. Stoudsburg PA:Annual Meeting of the Association for Computational Linguistics, 2019: 4171-4186.

[8] Lecun Y, Bottou L,Bengio Y,et al. Gradient-based learning applied to document recognition [J], Proceedings of the IEEE,1998, 86(11):2278-2324.

[9] Eiman, J.L. Finding structure in time[J]. Cognitive science,1990,14(2), 179-211.

[10] Salakhutdinov R. Learning deep generative models[J]. Annual Review of Statistics and Its Application,2015(2):361-385.

[11] Bengio S,Vinyals O,Jaitly N,et al. Scheduled sampling for sequence prediction with recurrent neural networks[J]. Computer Science,2015(2):1171-1179.

[12] Huszar F. How (not) to train your generative model:scheduled sampling,likelihood adversary?[J]. Computer Science,2015,7(1):11-18.

[13] Papineni K,Roukos S,Ward T,et al. BLEU:a method for automatic evaluation of machine translation[C]. Proceedings of the 40th Meeting on Association for Computational Linguistics. Stoudsburg PA:Annual Meeting of the Association for Computational Linguistics,2002:311-318.