DE GRUYTER
OPEN

Research Paper

A Topic Detection Method Based on Word-attention Networks

Zheng Xie[†]

College of Liberal Arts and Sciences, National University of Defense Technology, Changsha, China

Citation: Xie, Z. (2021). A topic detection method based on word-attention networks. *Journal of Data and Information Science*, 6(4), 139–163. https://doi.org/10.2478/jdis-2021-0032

Received: Jun. 19, 2021
Revised: Jul. 21, 2021
Accepted: Jul. 23, 2021

Abstract

Purpose: We proposed a method to represent scientific papers by a complex network, which combines the approaches of neural and complex networks.

Design/methodology/approach: Its novelty is representing a paper by a word branch, which carries the sequential structure of words in sentences. The branches are generated by the attention mechanism in deep learning models. We connected those branches at the positions of their common words to generate networks, called word-attention networks, and then detect their communities, defined as topics.

Findings: Those detected topics can carry the sequential structure of words in sentences, represent the intra- and inter-sentential dependencies among words, and reveal the roles of words playing in them by network indexes.

Research limitations: The parameter setting of our method may depend on practical data. Thus it needs human experience to find proper settings.

Practical implications: Our method is applied to the papers of the PNAS, where the discipline designations provided by authors are used as the golden labels of papers' topics.

Originality/value: This empirical study shows that the proposed method outperforms the Latent Dirichlet Allocation and is more stable.

Keywords Scientific topics; Text analysis; Deep learning

1 Introduction

One of the first things before researchers reading or writing a paper is to identify whether the topics addressed by the paper are relevant to their interests or in their scientific investigation. Knowing which topics are pursued in the scientific community helps researchers know which research areas are rising or falling in popularity. Topic finding is a main task in the natural language process (NLP),

[†] Corresponding author: Zheng Xie (E-mail: xiezheng81@nudt.edu.cn).

which requires structured representations for the unstructured text data. One of those is the Bag-of-Words, which ignores the dependencies among words (Sethy & Ramabhadran, 2008; Wallach, 2006). However, topics are extracted from sentences that carry semantic meaning. It is necessary to seek the representations that can store specific semantic information of sentences. Using multiple words would be helpful, which can express relations of words to carry semantic meaning (Doucet & Ahonen-Myka, 2010).

The sequential structure of words in a sentence carries specific semantic meaning. It expresses the dependencies among words. Neural networks are widely used to learn these dependencies, especially the intra-sentential ones, where deep learning models can be directly applied to data without the requirement of feature engineering. We can express the sequential structure by a "tree branch" with words as its nodes (word branch hereafter) and then generate a word network by connecting the branches from their common words. That network also carries specific semantic meaning and makes it possible to utilize the approaches in the analysis of complex networks to find large-scale dependencies beyond the scale in the sense of intra-sentential ones.

Learning dependencies using neural networks can be regarded as the feature engineering for the construction of that word network. Learning dependencies at the scale of sentences' length is important to a fine construction because ignoring them will lose many meaningful relations. Meanwhile, connecting all pairs of words in a sentence would generate numerous pseudo dependencies. The attention mechanism in the NLP field provides a solution. It has the ability to relate different positions of a single sequence, e.g. a sentence, giving a representation of the sequence. This mechanism has been used successfully in a variety of tasks, including machine reading (Cheng, Dong, & Lapata, 2016), abstractive summarization (Li et al., 2017), and learning task-independent sentence representations (Yin et al., 2016).

The representation of Bag-of-Words does not have sequential semantics. Meanwhile, the attention between words or tokens computed by attention architectures can describe their dependencies by a real value. However, this relationship is short, normally restricted in a sentence. In order to capture the long dependencies across sentences, we proposed a new representation of paper, namely word branch, which is different from the vector representation used in those attention architectures. Connecting word branches to form a network, we gave a new path to bridge two research fields, complex networks and text analysis, which is the contribution of our work.

The particular model used to compute the attention between words is the Transformer, which is based solely on attention mechanisms (Vaswani et al., 2017). It extracts dependencies without regard to their distance in the input sequences and the previous output sequences (Kim et al., 2017). The Transformer is trained on the datasets consisting of the preprocessed sentences in abstract and titles, where a training sample is the same as its corresponding target. We input the preprocessed title of a paper as a sequence to the Transformer, which will output a sequence, a list of tokens.

For each position of an output sequence, we not only record the token placed on this position but also record a given number of candidate tokens according to the probability to be placed on this position, from high to low. For each position, we generated direct edges from the token with the highest probability to all of the recorded tokens on the next position. This generates a "word branch", a representation of a paper. An advantage of the branch representation is that its directed edges carry the sequential information of the corresponding title and that learned from abstracts, thereby carrying more semantic information than the representation in the form of token set or distribution.

We constructed a network by connecting the branches at the positions of their common words and called it word-attention network. Then, we used the methods of detecting communities to detect topics. Due to the aforementioned merit of branches, the topics in the form of subgraph also capture the sequential structure of tokens in sentences, not just the frequency information of tokens. Our method is applied to the papers in the Proceedings of the National Academy of Sciences (PNAS). The results of this empirical study showed that our method outperforms the Latent Dirichlet Allocation (LDA), which is assessed based on papers' discipline designations. The evolution of those topics not only illustrates the trend of the development of sciences but also reveals the change of the roles of concepts playing in topics.

This paper is organized as follows. The literature review is provided in Section 2. The empirical data and the method are described in Sections 3, 4. The results are analyzed and compared in Sections 5, 6. The results are discussed, and conclusions are drawn in Section 7.

2 Literature review

In the NLP field, topic detection is a task of clustering words based on specific relations among them, allowing us to extract features from texts automatically. Frequent itemset mining (FIM) is a simplistic model proposed to find concepts or fine-grained topics as frequent itemsets of words, using Bag-of-Words presentation and working on the assumption that the high frequency of a term means high importance. Frequent items are those meeting a thresh-old (Agrawal & Srikant, 1994).

Topic models can improve the clustering quality of texts by grouping related words together rather than using a single word as a feature. The toolkit "genism" supports multiple forms of learning and inference on many topic models[1], including the default form that supports multi-threaded training and inference on multi-core machines. Specifically, it can use a collapsed Gibbs sampler or the collapsed variational Bayes approximation for the LDA to improve its computing speed.

A kind of topic models use Bag-of-Words, and the LDA is a typical one (Asuncion et al., 2012; Blei, Ng, & Jordan, 2002; Griffiths & Steyvers, 2004). This model regards a text as a mixture of inferred topics, each being a distribution of words. The LDA is an unsupervised method. There is a supervised version, called Labeled LDA, where the input data contain the information of labels (Ramage, Manning, & Dumais, 2011). It can be used to discover which parts of a text go with each label and to learn how the words best associate with labels globally. Partially Labeled LDA (PLDA) generalizes the Labeled LDA, allowing more than one latent topic per label and a set of background labels. Learning and inference in this generalized one are much like those of the Labeled LDA, expect the requirement of additionally specifying the number of topics associated with each label. The PLDA has been widely extended in different ways and applied to the study of topics in a large dataset of scientific papers and theses (Talley et al., 2011).

Bag-of-Words ignores the sequential structure of words in a sentence and thus drops the semantic information carried by the structure. Extracting and classifying relations of words from a text or a sentence is a NLP task called Relation Extraction (RE). The extracted relations usually occur between two or more specific words known as entities of a certain type (such as person and location) and fall into a number of semantic categories (such as employed by and lives in). The dependencies among more than two entities are referred to *n*-ary relations, a typical representation of which is the cliques of the graph formed by entities as nodes and their binary dependencies as edges (McDonald et al., 2005). This drops the semantic information carried by the sequential structure of entities. The method Semantic Roles Labelling does better, which extracts *n*-ary relations by identifying the predicates and arguments in a sentence (Gildea & Jurafsky, 2002; Kingsbury & Palmer, 2002).

Data labeling is a challenging task in relation extraction. Supervised relation extraction approaches are specific to a certain vertical domain as the corresponding training data is typically small and specific. Distant Supervision or similar ideas are alternative approaches that use semantic data to automatically obtain the labels of relation types (Mintz et al., 2009). There exist models at different levels to extract intra- or inter-sentential dependencies (Swampillai & Stevenson, 2011). RE models

[1] https://radimrehurek.com/gensim/

at the global level take a text corpus and a list of entity sets needed to find their semantic relations as input. RE models at the mentioned level take an entity set and the sentence that contains it as input.

Convolutional neural networks have been used to extract lexical and sentential level features without requiring feature engineering (Zeng et al., 2014). The ability of a network to learn word dependencies is affected by the length of a path forward and backward that signals have to traverse in a neural network. The shorter the path between the positions of two words in a sentence, the easier it is to learn the dependencies of the words (Schmidhuber, 2001). Attention mechanism was proposed to offset the effect of path length, which has become a module in most models on the RE task (Hochreiter & Schmidhuber, 1997).

Vaswani et al. proposed a famous network architecture, the Transformer (Vaswani et al., 2017), based solely on attention mechanism, dispensing with recurrence and convolutions entirely. Its attention function is defined as mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The output is a sequence of words that are selected according to a weighted sum of the values, where the weight assigned to each value is computed by a compatibility function of the query with the corresponding key. In the Transformer, the number of operations required to relate signals from two arbitrary input or output positions is a constant. Note that this number grows linearly for Con-vS2S (Gehring et al., 2017) and logarithmically for ByteNet (Kalchbrenner et al., 2016). Here, the Transformer will be used to construct a network of words.

Topic detection has been applied and then presents challenges to the community of informetrics. Boyack et al. analyzed a corpus of 2.15 million records (titles, abstracts, and subject headings) from MEDLINE 2004–2008 by cosine similarity using term frequency-inverse document frequency vectors, latent semantic analysis, topic modeling, and two Poisson-based language models (Boyack et al., 2011). Small et al. used a different function to identify emerging topics in science and technology, which is based on direct citations and co-citations (Small, Boyack, & Klavans, 2014). Velden et al. provided a framework of how to describe and distinguish approaches to topic extraction from bibliographic data of scientific papers (Velden et al., 2017). Zhang et al. proposed a kernel *k*-means clustering method incorporated with a word embedding model to extract topics from bibliometric data (Zhang et al., 2018).

3 Empirical data

We applied the proposed method to 56,740 papers of the PNAS from 1999 to 2014. The results would contribute to understanding the aspects of scientific topics due to the influence and representability of the PNAS. There are 50,444 papers that are categorized into three major disciplines: Biological, Physical, and Social Sciences. The data contains 43,487 papers that are categorized into Biological Sciences (including 3,569 papers categorized into a minor discipline Biophysics), 5,684 papers into Physical Sciences, and 1,273 papers into Social Sciences. Figure 1 shows the annual number of papers of each major discipline, where the fraction of papers of Physical Sciences and that of Social sciences increase over time. Figure 1 also shows the large fraction of papers of Biophysics in Biological Sciences, which indicates the difficulty of distinguishing the papers of Biological Sciences and those of Physical Sciences.

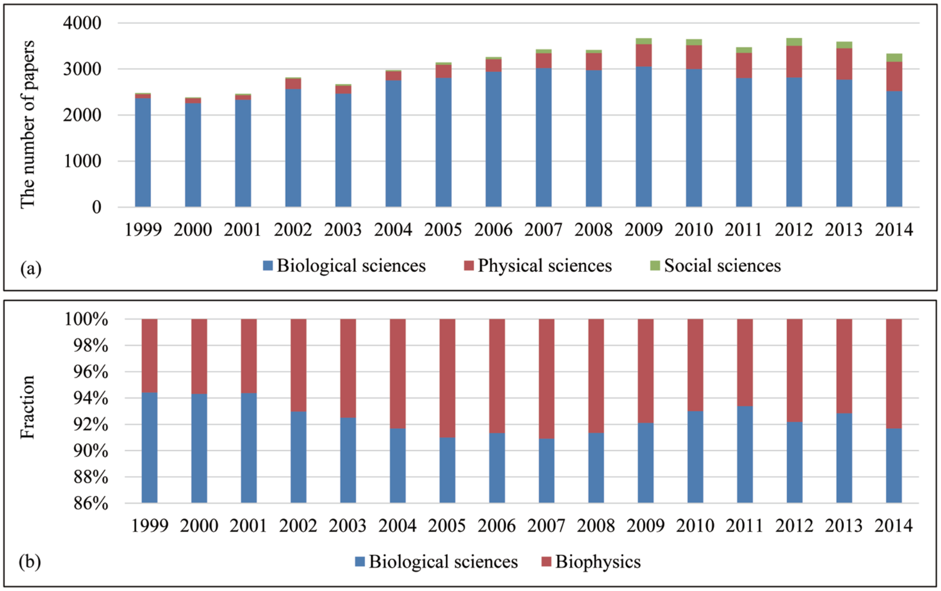

Figure 1. The statistics of the PNAS papers' in three major disciplines. There are 90.44% of papers belonging to major disciplines, namely Biological Sciences, Physical Sciences, and Social Sciences. Panel (a) shows the annual number of papers of each discipline. Panel (b) shows the fraction of papers of Biophysics in Biological Sciences.

We preprocessed data to make the output useful for the purpose of topic identification (Algorithm 1). We converted the texts to lowercase, only kept words, removed stopwords (e.g. "the"), and shorten words to their roots or stems. The remains are the tokens analyzed here. We further removed tokens that appear in less than six papers' abstract and title (0.1% of the number of papers) because rare words tell little about the similarity of papers.

Algorithm 1. Data preprocessing.

```
Input:
    titles and abstracts of papers;
Output:
    list of token lists.
stem words using the PorterStemmer of NLTK[2];
remove stopwords using the stopword corpus of NLTK;
remove the words that appear in less than $x = 6$ papers' abstract and title.
```

The training dataset consists of the preprocessed sentences in abstracts and titles. Notably, each training sample is the same as its corresponding target. In order to show the evolution of topics, we split the training dataset annually. That is, we trained the Transformer on those annual datasets respectively and used it on the preprocessed titles of the papers at the corresponding year.

4 The method

In Subsection 4.1, we showed a new representation of scientific papers (word branch) based on the attention mechanism in deep learning . In Subsection 4.2, we showed a topic detection method based on this representation, where topics are detected by detecting the communities of a network formed by connecting word branches. In Sections 5 and 6, we showed the applications of this method, the results of carrying it out, and the experimental comparisons with the LDA. Figure 2 illustrates the operations of our method. The topics output here are directed subgraphs with edges carrying the sequential structure of sentences. Therefore, our method can capture the semantic information carried by the sequential structure, which cannot be captured by the models using Bag-of-Words.

4.1 Generating word branches

The first step of our method is constructing word branches by an attention model. The Transformer is used here, the structure of which is encoder-decoder. The encoder maps an input sequence of tokens to a sequence of vectors, and the decoder then outputs a sequence of tokens one by one. At each step, the Transformer uses the previously generated tokens as additional input to generate the next token and uses stacked self-attention and point-wise, fully connected layers for both the encoder and decoder. Here, there are four layers for both the encoder and decoder, and four heads in multi-head self-attention. The dimension of token embedding and the hidden dimension of feed forward layer are set to be 32.

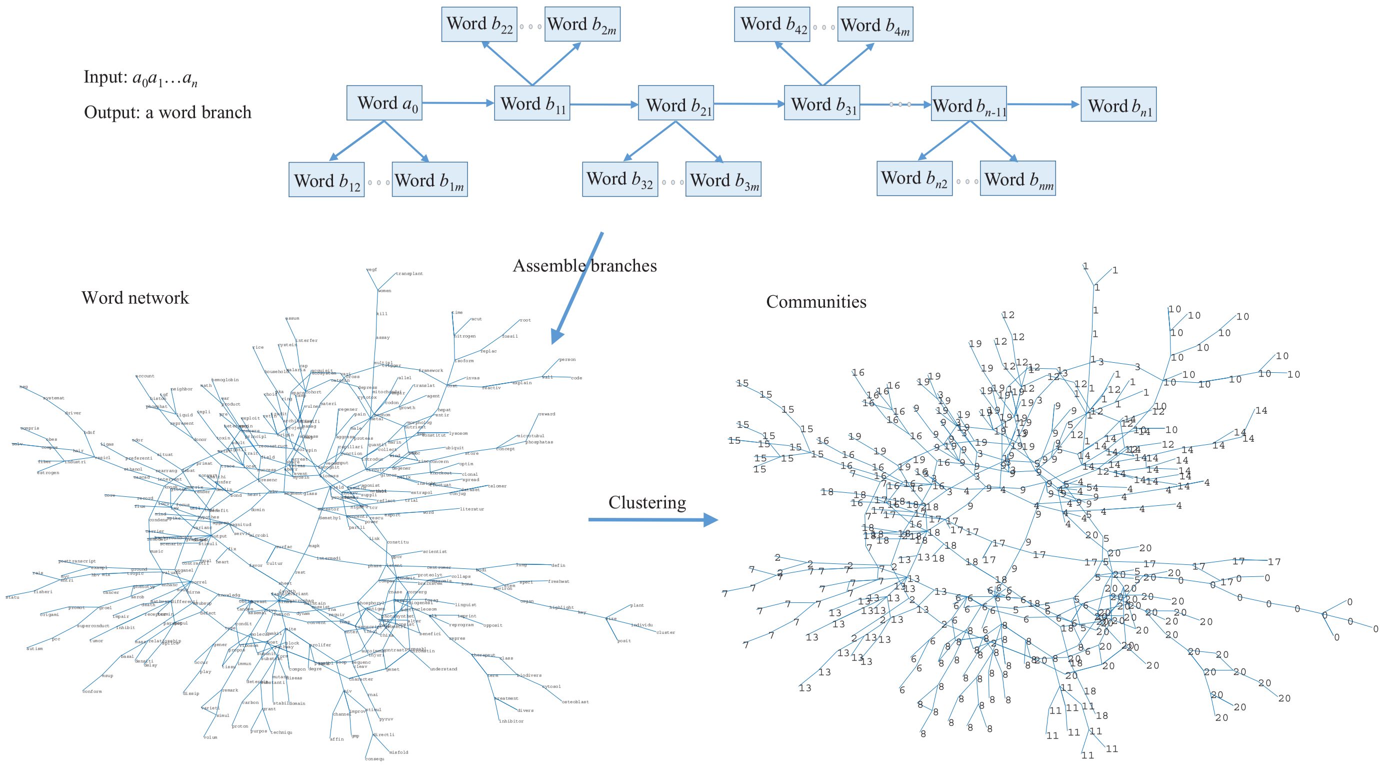

Figure 2. The illustration of the proposed method. The method consists of two steps: constructing word branches by the Transformer; using methods of community detection to partition the network that are formed by connecting word branches. The topics here are in the form of subgraph.

Algorithm 2 shows the pseudo codes of the Transformer used here, which is coded by the python toolkit “keras-transformer”[2]. We trained the Transformer on the annual datasets respectively, and then input it the preprocessed titles of the papers at the corresponding year. The output of the decoder at each step is a token sampled from a categorical distribution of tokens. In the training step, the loss function used here is the categorical cross entropy; name the cross entropy of two categorical distributions. The cross entropy of the distribution q relative to a distribution p over a given set is defined as $H(p, q) = -Ep(\log q)$, where $Ep(\ )$ is the expected value operator with respect to the distribution p. Here, there are two or more label classes, and labels are expected to be provided as integers. Therefore, we used the sparse categorical cross entropy loss.

Algorithm 2. The architecture of the Transformer used here.

```
model=get model(token num=max(len(source token dict), len(target token
    dict)), embed dim=32, encoder num=4, decoder num=4, head num=4,
    hidden dim=32, dropout rate=0.05, use same embed=False,)
model.compile(’adam’, ’sparse categorical crossentropy’)
model.fit(x=[np.array(encode input∗30),    np.array(decode input∗30)],
    y=np.array(decode output∗30), epochs=5, batch size=32, )
```

Here, the used optimizer is Adam, an algorithm for first-order gradient-based optimization of stochastic objective functions, which is based on adaptive estimates of lower-order moments.

The Transformer allows every position in the decoder to attend overall positions in the input sequence and previous positions in the output sequence. This helps find dependencies between tokens. In the experiments of this paper, the dependencies are between the tokens at most between 30 tokens, a parameter in Algorithm 2.

Now, we show how to construct a word branch by the Transformer with the input of a preprocessed title (Algorithm 3). For each preprocessed title $a_0,...,a_n$, we let $b_{01} = a_0$, and use the Transformer with the input $b_{01}, ..., b_{i1}$ ($i = 1,...,n$ 1) to predict tokens $b_{(i+1)1},..., b_{(i+1)m}$ (which are ranked according to the probability given by the Transformer). Then, we generate directed edges from $b_{(i-1)1}$ to $b_{i1},..., b_{im}$.

Note that although the input is the title, the tokens and edges of the output are learned from the training dataset, namely the sentences in abstracts and titles. That is, the edges from token $b_{(i-1)1}$ to token b_{i1} for any possible i are not only learned from the sequential structure of tokens in titles but also from the structure in abstracts. Note that the branches carry the semantic information represented by the sequential structure of tokens in a sentence, not just token frequency.

[2] https://https://pypi.org/project/keras-transformer/

Algorithm 3. Constructing word branches.

```
Input:
    titles of papers;
    parameter $m$;
Output:
    word branches.
for each preprocessed title $a_0, ..., a_n$ do
    let $b_{01} = a_0$;
    for $i$ from 1 to $n$ do
        predict $b_{i1}, ..., b_{im}$ ranked according to the probability given by the Transformer;
        if $i > 1$ then
            generate directed edges from $b_{(i-1)1}$ to $b_{i1}, ..., b_{im}$;
            use $b_{01}, ..., b_{i1}$ to predict next tokens by the Transformer;
        end if
    end for
end for
```

The ubiquitous words, e.g. “research”, cannot tell us the similarity of papers, which are filtered here by tf-idf. The matrix of tf-idf weight is calculated based on the token-paper matrix $(f_{ij})_{N\times M}$, where f_{ij} is the frequency of token i in the title and abstract of paper j. For each title input into the Transformer, we only consider the top 2% tokens according to this weight. That is, the other tokens are filtered from the corresponding branch.

In detail, we calculated the tf-idf matrix $(w_{ij})_{N\times M}$, and then ranked to- kens according to $\{w_{i1}, ..., w_{iN}\}$. For i from 1 to the number of papers M, if to- ken b_{i1} is not in the top 2% of this rank, we cropped the tokens of $\{b_{i1}, ..., b_{im}\}$ and the edges connecting those tokens; else, we cropped the tokens of $\{b_{i2}, ..., b_{im}\}$ that are not in top 2% and the edges connecting those tokens (Algorithm 4).

Algorithm 4. Cropping word branches.

```
Input:
    word branches;
Output:
    cropped word branches.
calculate the token-paper matrix $(f_{ij})_{N\times M}$;
calculate the tf-idf matrix $(w_{ij})_{N\times M}$;
for $i$ from 1 to $M$ do
    rank tokens according to $\{w_{i1}, ..., w_{iN}\}$;
    if token $b_{i1}$ is not in top $x$% then
        crop the tokens of $\{b_{i1}, ..., b_{im}\}$;
        crop the edges connecting those tokens;
    else
        crop the tokens of $\{b_{i2}, ..., b_{im}\}$ that are not in top $x$%;
        crop the edges connecting those tokens.
    end if
end for
```

The tf-idf weight evaluates the importance of a token to a record in a dataset based on the assumption that the importance increases proportionally to the number of times a token appears in the record but is offset by the frequency of the token in the dataset. The weight is multiplied by two terms, namely tf and idf. The first term is the number of times a token appears in a record divided by the total number of tokens in that record. The second term is the logarithm of the number of records in the dataset divided by the number of records where that token appears. Notably, the training datasets are not filtered by tf-idf, because filtering too many words will break the structure of sentences and decrease the prediction precision of the Transformer.

4.2 Detecting topics by network community detection

We connect branches at the positions of their common words and then construct a directed graph with weights on edges, called a word-attention network. The weight of the edge from a token w_1 to a token w_2 is the number of that edge in all branches. Here, the partitioned subgraphs or communities of this word network are defined as topics, each being a subgraph formed by a community of tokens and the direct edges between these tokens.

Notably, the edges of the word network carry specific semantic information inherited from branches, so do the edges of the topics. The directions of edges should be considered in network partition due to the potential value of the structure information of sentences. Consider two tokens w_1 and w_2. Token w_1 has high out-degree but low in-degree, while token w_2 has the reverse situation. This means that a given edge is more likely to run from w_1 to w_2 than that from w_2 to w_1. It would be unnatural to observe an edge from w_2 to w_1, compared with an edge from w_1 and w_2 (Leicht & Newman, 2008).

We now apply algorithms of community detection to a word network. The detected communities are regarded as the topics detected by our method. The number of communities relates to algorithms' resolution, which allows us to zoom in the network and to observe its structure with the desired resolution. For a dataset of scientific papers, low resolutions would lead our model to find a relatively small number of topics, e.g. the topics at the disciplinal level, whereas high resolutions will produce more specific topics that address small areas of research.

Notably, word networks would contain many edges. Therefore, the method to partition it needs a relatively low computation cost. Community detection by gradually removing the edge with the highest betweenness can also be applied to directed graphs (Girvan & Newman, 2002). The idea is that the betweenness of an edge connecting two communities is typically high. The cost of computing betweenness is high, and thus it is not suitable for dense networks.

4.3 Evaluation indexes

If the papers of a class almost have the same label, then the class can be regarded as well classified. Here, a paper class is defined to be a well-classified one at the level of *x*% if more than *x*% of its papers have the same label. For the purpose of assessment, we used an index, $R(x)$, the ratio of the papers of well-classified classes over all of the papers. It can measure the degree how the proposed method can reveal the information of papers' disciplines. The result of $R(x)$ depends on the threshold x.

The entropy operator (Shannon, 1948) can be used to assess the degree of well-classifying papers. Consider a class C_s with m papers $\{P_1, ..., P_m\}$. The lower the degree is, the more even the class distributing over disciplines $\{D_1, ..., D_n\}$, and then the higher the uncertainty which disciplines the class belongs to. The entropy operator can measure the uncertainty without the threshold x in the index $R(x)$. The probability of Cs belonging to D_i is $p_{si} = |C_s \cap D_i|/m$, and so the entropy of C_s is

$$H(C_s) = -\sum_{i=1}^{n} p_{si} \log_2 p_{si} \tag{1}$$

5 Experiments

We applied the proposed method to the PNAS dataset for each year from 1999 to 2014 respectively, specific statistical indexes of which are shown in Table 1. The information of a papers' major discipline is used as the golden label of that paper. That is, the corresponding paper classes here are regarded as topics at the disciplinary level. We classified a paper by its principle topic, which is defined as the one that shares most tokens with the paper.

We applied three typical algorithms of community detection to a word network. The random walk method is used to find community structure for directed word networks (Pons & Latapy, 2005). The louvain method is a popular way of community detection for undirected networks (Blondel et al., 2008). A word can be involved in several topics. However, the methods of community detection used above cannot give overlapped communities. Defining a community to be a set of closely interrelated edges can induce overlapped communities of nodes (Ahn, Bagrow, & Lehmann, 2010). Given a word network, we can construct a network of edges, where two edge-nodes are connected if they share nodes. Then, we can apply the louvain method to obtain communities of edges, namely the overlapped communities of tokens.

Table 1. The information of the dataset dblp.

Time	a	b	c	d	e	f
1999	2,475	3,274	95,021	0.11	2.371	0.998
2000	2,380	3,347	93,910	0.101	2.395	0.998
2001	2,455	3,477	108,954	0.108	2.355	0.999
2002	2,812	3,710	117,269	0.094	2.272	1.0
2003	2,656	3,592	115,019	0.1	2.312	0.999
2004	2,955	3,919	138,451	0.101	2.299	0.999
2005	3,131	4,084	154,041	0.099	2.275	0.999
2006	3,248	4,260	166,614	0.1	2.289	0.999
2007	3,419	4,368	184,420	0.102	2.279	0.999
2008	3,408	4,436	184,881	0.104	2.304	1.0
2009	3,658	4,609	212,771	0.098	2.218	1.0
2010	3,639	4,668	221,090	0.1	2.204	0.999
2011	3,462	4,688	220,020	0.111	2.228	0.999
2012	3,621	4,875	209,517	0.114	2.28	1.0
2013	3,593	4,846	231,959	0.096	2.189	1.0
2014	3,334	4,679	210,099	0.096	2.208	1.0

The index *a*: the number of branches (papers), *b*: the number of nodes (tokes), *c*: the number of edges, *d*: the global clustering coefficient, *e*: the average length of shortest paths, and *f*: the proportion of giant component.

Firstly, we used the random walk method to find communities for directed word networks. The idea of the method is that short random walks tend to stay in the same community. The method provides a dendrogram, thereby allowing us to zoom in the network and to observe its structure with the desired resolution. The topics discovered by using this method are found in a completely unsupervised fashion, using no prior information. Therefore, the discovered topics can be considered as purely a consequence of the structure of datasets.

Figure 3 shows the results of our method using the random walk method. Notably, the ratio of directed edges with inverted edges is only 1.94%. That is, most of the paths of word networks are one-way. To show the effect of edges' direction, we also applied the random walk method to word networks by treating them as undirected ones, but obtained the results without significant difference. One possible reason is due to the small difference between these directed and undirected networks on the average length of shortest paths, the averages of which are 2.280 and 2.624 respectively.

Secondly, we used the louvain method, which has a tunable resolution and can unfold a complete hierarchical community structure of the network at a tunable resolution (Blondel et al., 2008). It is based on the modularity optimization. Thus, it embeds an implicit definition of community: a subgraph is a module if the number of edges inside it is larger than the expected number in modularity's null model. If this is the case, the nodes of the subgraph are more tightly connected than expected, and a large positive value of modularity is expected to indicate a good partition.

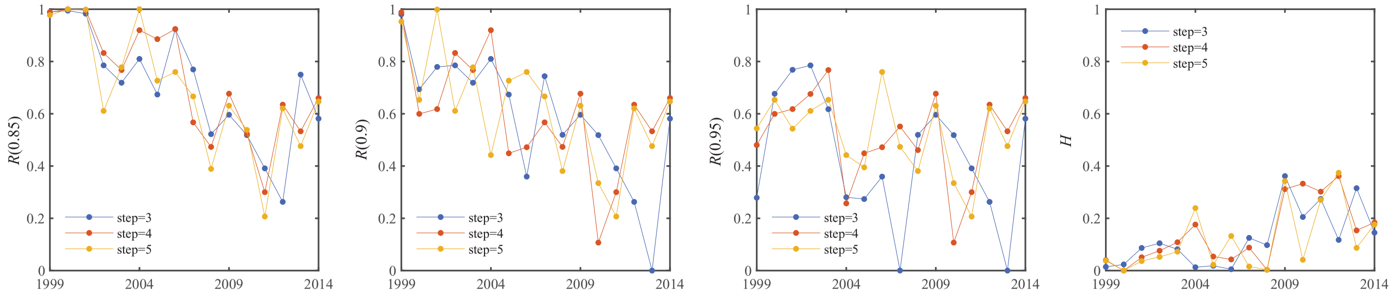

Figure 3. The performances of using the random walk method to detect node communities. The word networks are treated as directed ones. The step is the parameter of the random walk method, the length of random walks to perform. Panels show the performances of the method with the step 3 (blue dot lines), 4 (red dot lines) and 5 (orange dot lines).

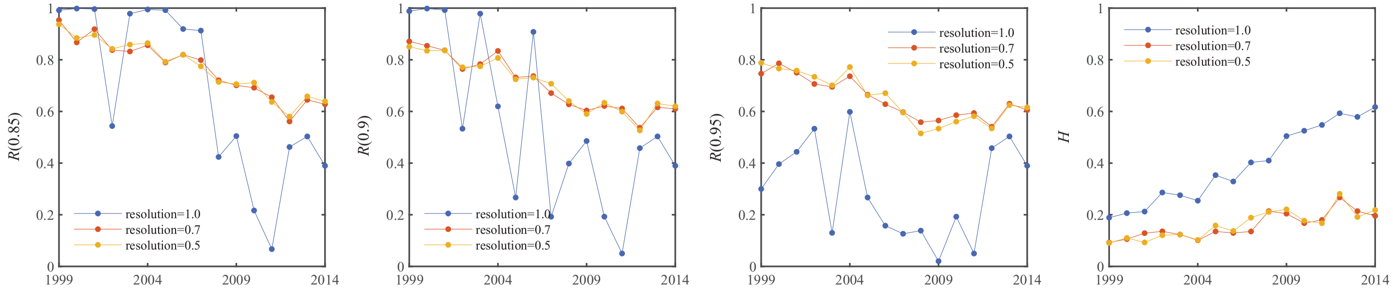

Figure 4. The performances of using the louvain method to detect node communities. The word networks are treated as undirected ones. The resolution is the parameter of the louvain method. Panels show the performances of the method with the resolution 0.5 (orange dot lines), 0.7 (red dot lines), and 1.0 (blue dot lines).

Figure 4 shows the results of our method using the louvain method. Its performance is improved when tuning the resolution parameter from 1 to 0.5 to obtain a small number of communities. The existence of 3,569 papers of Biophysics fuzzes the boundary between the paper class of Biological Sciences and that of Physical Sciences. Therefore, our method cannot reveal much difference between the two classes at the lexical level. Consequently, the performances of our method decrease with the growth of the papers of Physical and Social sciences.

Thirdly, we applied the louvain method to detect communities of edges, namely overlapped communities of tokens. Figure 5 shows its performances, which are not improved compared to that using the louvain method to detect communities of nodes. The number of edge communities is larger than that of token communities. Thus we tuned the resolution parameter from 1 to 0.7 to obtain a small number of communities. However, the performances are still not improved.

6 The difference from the results of LDA

The LDA describes the relationship between words and topics, as well as the relationship between topics and papers. It creates topics as word distributions, where the number of topics is specified. The first step of LDA is to convert a dataset into a matrix. The codes of LDA in the python toolkit “gensim” are used here to compute topics with the input sentence-word matrix, where the sentences are those in papers’ abstract and title. In order to assess the performances of the LDA, we assigned each paper to the label of its principle topic, the one with the largest probability the paper involves.

The LDA and our methods provide topics that are defined as the distributions or subgraphs of tokens respectively. Therefore, we have to classify papers based on the results of topic detection and then analyze the relationship between the obtained paper classes and discipline labels. Figure 6 shows that our method using the louvain method to detect node communities (resolution= 0.7) outperforms the LDA (the number of topics= 5), and is more stable. Considering the difference in the number of topics, we let the topic number of LDA be equal to that of our method. Figure 7 shows that there is no significant improvement in the assessment indexes.

The results of the provided method are better than those of the LDA, which validates its effectiveness on topic finding. Moreover, our method can provide substantial information on topics that are carried by network structure, not just the token frequency that can also be given by the LDA. The network indexes of the topics in the form of a subgraph can reveal specific roles of tokens playing in topics. For example, the role of the hub would emphasize the tokens that connect multiple tokens, which cannot be revealed by frequency. For a dataset consisting of papers

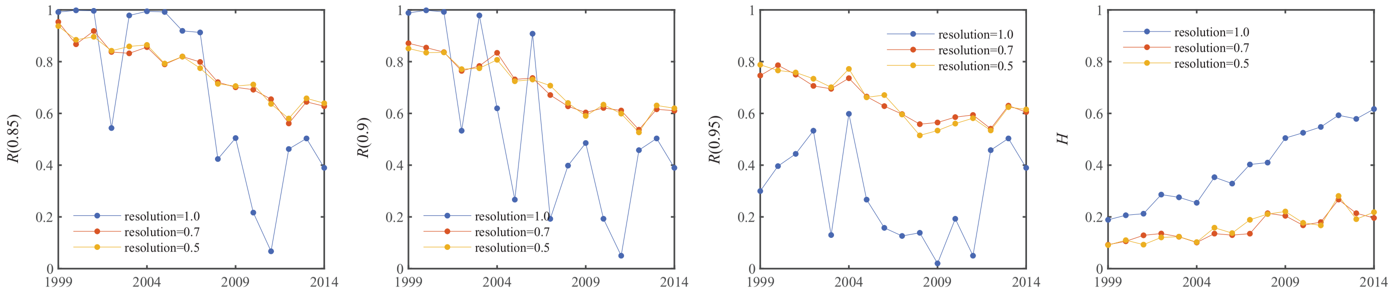

Figure 5. The performances of using the louvain method to detect edge communities. The louvain method is used to obtain overlapped communities. Panels show the performances of this method with resolution 0.5 (orange dot lines), 0.7 (red dot lines), and 1.0 (blue dot lines).

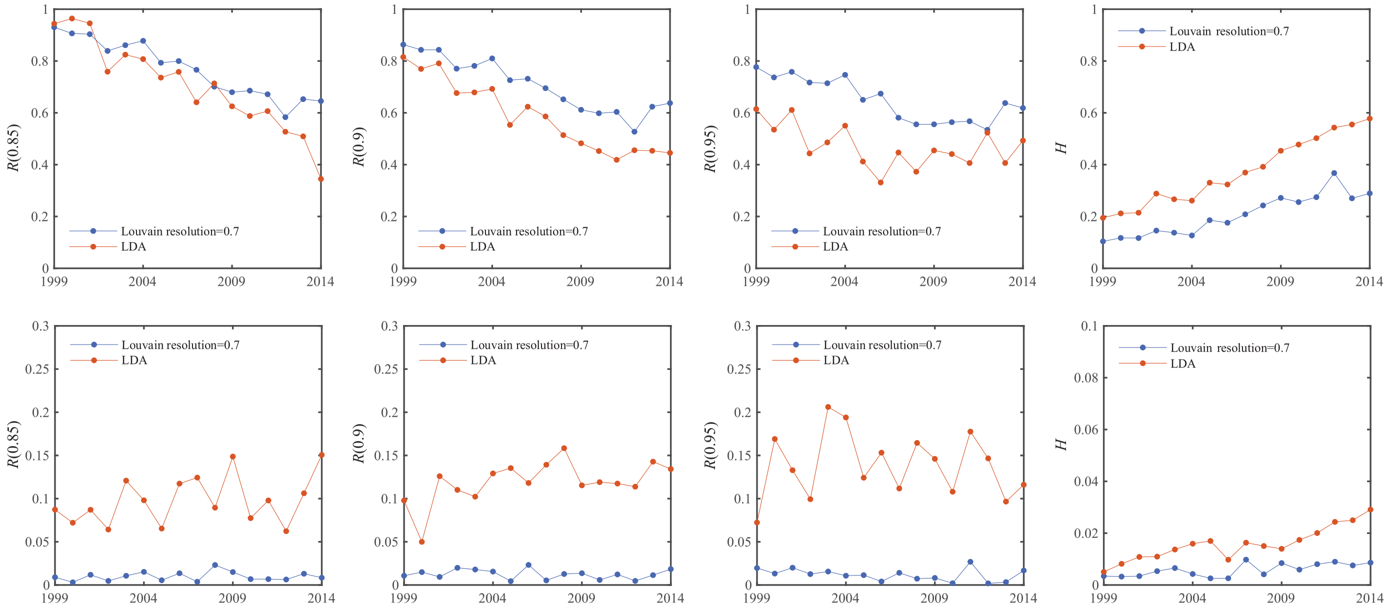

Figure 6. The comparisons between the LDA and our method. Panels (a-d) show the average values of assessment indexes, and Panels (e-h) show the standard deviations for running the LDA (red dot lines) and the provided method using the louvain method to detect node communities (resolution 0.7, blue dot lines).

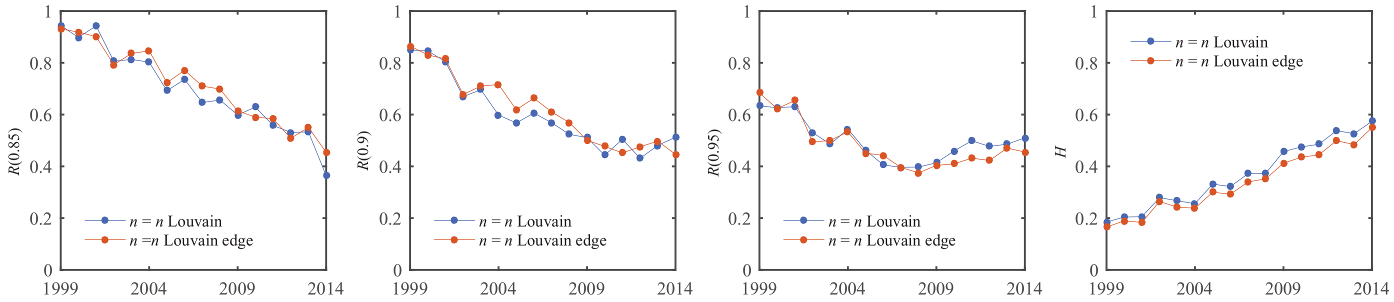

Figure 7. The performances of the LDA with the same number of topics as that of our method. Panels show the performances of the case with the number of communities of tokens (blue dot lines) and the performances with the number of communities of edges (red dot lines). These communities are detected by the louvain method.

at a particular time interval, those indexes can be used as quantitative measurements to track the changes of topics and the changes of tokens within topics.

A hub word plays the role of a networking device that allows one to connect multiple nodes to a network. Thus the concepts proxied by hub tokens would connect multiple concepts, playing critical roles in interdisciplines. Betweenness centrality measures the degree of being a hub in a network based on the shortest paths. For every pair of nodes in a connected network, there exist the shortest paths between the nodes such that either the number of edges that the paths pass through for unweighted networks or the sum of the weights of the edges for weighted networks is minimized. The betweenness centrality of a node is the number of these shortest paths that pass through that node.

Figure 8 shows that the trend (from 1999 to 2014) of the top five tokens (according to the betweenness centrality) in the top five topics (according to the number of a topic's tokens) detected by our method. The size of a token node is proportional to its betweeness centrality. The token "cell" keeps its role as a hub; meanwhile, "dna", "gene", and "protein" drop their centrality. Figure 9 shows that the trend of the top five tokens in the top five topics detected by the LDA. Note that the tokens at some years are more than 20. This is because that the results of LDA have certain randomness, and the tokens are the results of running ten times. The size of the token node is proportional to the summation of its weight over five topics, which reflects the information of word frequency. Therefore, we can conclude that "gene" and "protein" are still the tokens frequently appearing in scientific research, different from their decreasing centrality as hubs.

The differences between the tokens attended by the LDA and our method are shown in Figure 9, and Figure 8 may be due to the difference in the representation of paper, namely the difference between Bag-of-Words and word branch. The LDA attends more on the frequency of tokens, while our method on the tokens' role in networks. The LDA cannot address the positions of words in a sentence, whereby missing the structure of sentences. To further show the difference in the tokens they emphasized, we summed the weights of tokens in the top five topics detected by the LDA over topics and years, counted the top five tokens in the top five topics detected by our method also over topics and years, and then draw word clouds for these tokens emphasized by the two methods respectively. Figure 10 shows that many of the tokens emphasized by both methods are concepts proxied by nouns and some of the tokens only emphasized by our method are verbs' stem. The collocations of verbs and nouns would bring clear semantic meaning.

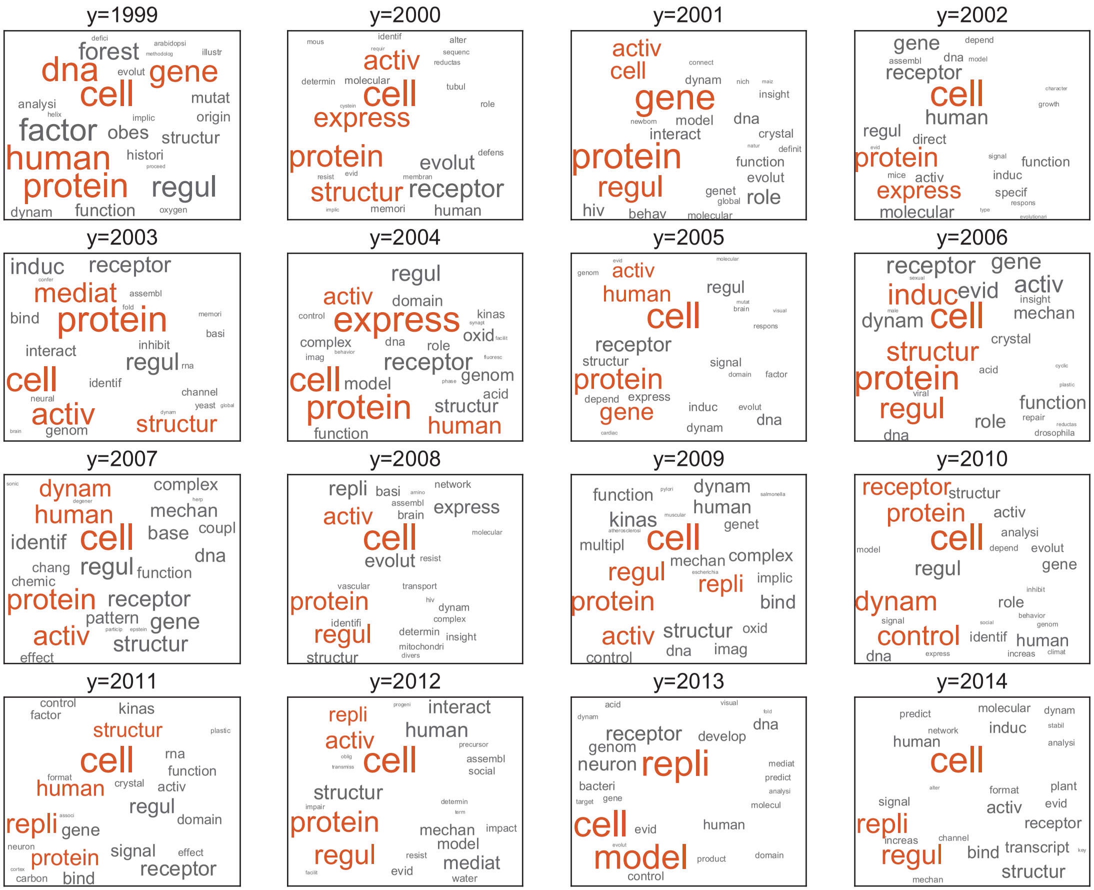

Figure 8. The evolution of the tokens is emphasized by the proposed method using the louvain method. Panels show the word clouds of the top five tokens in the top five topics according to tokens' betweenness centrality in the word network.

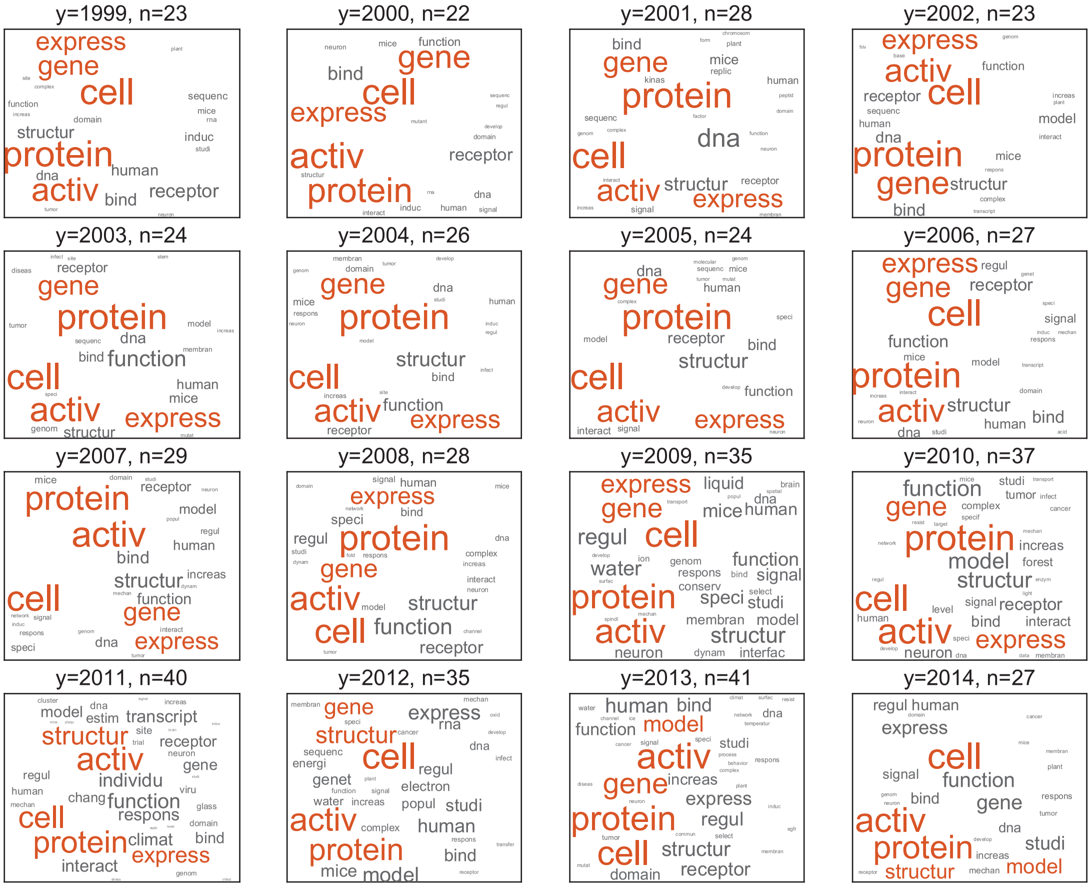

Figure 9. The evolution of the tokens emphasized by the LDA. Panels show the word clouds of the top five tokens in the top five topics according to the summation of a token's weight overall topics. The number of topics is five. The index *n* here is the number of tokens emphasized by running the LDA ten times.

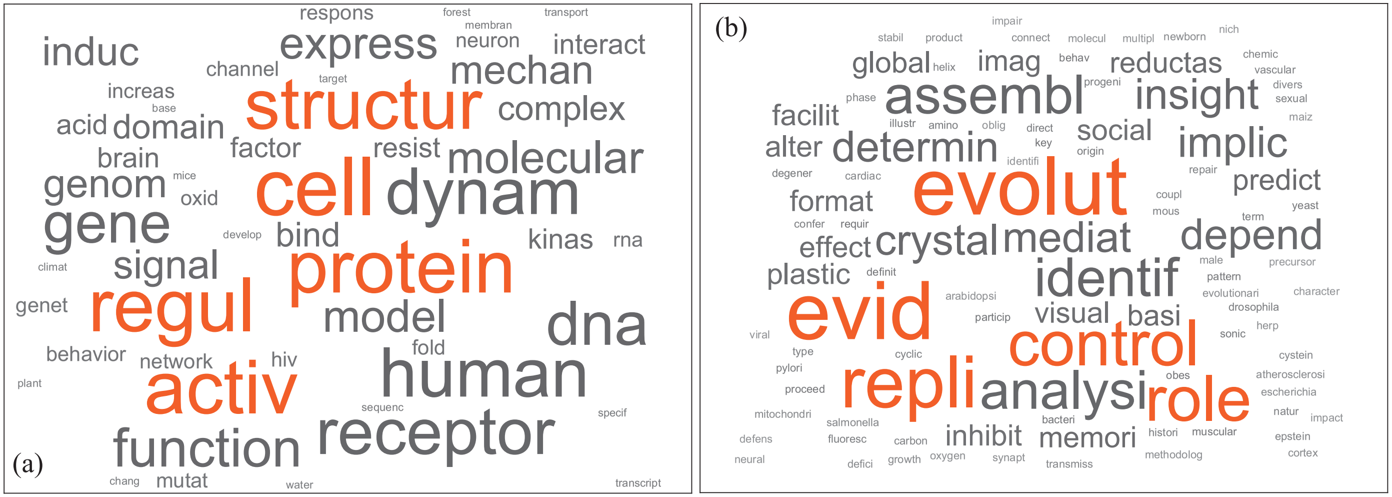

Figure 10. The difference on emphasized tokens. Panel (a) shows the tokens emphasized by both the LDA and our method. The set of tokens emphasized by the provided method includes that emphasized by the LDA. Panel (b) shows the tokens only emphasized by our method.

7 Discussion and conclusions

We proposed a method to represent a scientific paper by a word branch that is computed by the attention mechanism of the deep learning architecture Transformer. Then we represented the dependencies among scientific papers by a word-attention network that is generated by connecting word branches. As one of its applications, we illustrated how the network could be used to find topics of papers by partitioning the network into subgraphs. The branches and so the topics capture the sequential structure of sentences in titles and abstracts and consequently carry the corresponding semantic information, which is missed by the methods of detecting topics in the form of word sets or distributions.

The method of detecting topics here is completely unsupervised, but prior information can be integrated. Prior information about community structure can be used by supervised algorithms. Due to the unsupervised feature, the topics discovered here can be considered as genuinely a consequence of the informative structure of data. The method was applied to the PNAS dataset that consists of 56,740 papers from three major disciplines. The empirical study here shows it outperforms the LDA. Notably, the parameter setting of our method may depend on practical data. Thus it needs human experience to find proper settings.

The proposed method has several applications that make it easier to understand the information contained in scientific papers. The experiment on the PNAS dataset shows the evolution of topics and the change of the roles of concepts playing in topics. Therefore, the method would be useful for the recognition of topics and concepts rising and falling in the scientific community, whereby useful for administers determining funding targets and for researchers determining study targets.

Whereas in this article, we only focused on the analysis of scientific papers, as represented by the papers published in the PNAS, the method and applications we have presented are relevant to a variety of other domains. It has the potential to make it easier for people to understand the information contained in knowledge domains, including exploring topic dynamics, discovering large-scale dependencies of words, and indicating the roles of words play in semantic content. Discovering the topics in the form of a subgraph for documents, from e-mail records and newsgroups to the entire Internet, helps visualize their content and evolution.

Acknowledgements

This study is supported by National Natural Science Foundation of China (Grant No. 61773020).

References

Agrawal, R., & Srikant, R. (1994, September). Fast algorithms for mining association rules. In Proceedings of the 20th International Conference of Very Large Data Bases. 1215, 487–499.

Ahn, Y.Y., Bagrow, J.P., & Lehmann, S. (2010). Link communities reveal multiscale complexity in networks. nature, 466(7307), 761–764.

Asuncion, A., Welling, M., Smyth, P., & Teh, Y.W. (2012). On smoothing and inference for topic models. UAI Press. arXiv:1205.2662.

Blei, D.M., Ng, A.Y., & Jordan, M.I. (2003). Latent dirichlet allocation. the Journal of machine Learning research, 3, 993–1022.

Blondel, V.D., Guillaume, J.L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and experiment, 2008(10), P10008.

Boyack, K.W., Newman, D., Duhon, R.J., Klavans, R., Patek, M., Biberstine, J.R., ... & Börner, K. (2011). Clustering more than two million biomedical publications: Comparing the accuracies of nine text-based similarity approaches. PloS one, 6(3), e18029.

Cheng, J.P., Dong, L., & Lapata, M. (2016). Long short-term memory-networks for machine reading. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 551–561.

Doucet, A., & Ahonen-Myka, H. (2010). An efficient any language approach for the integration of phrases in document retrieval. Language resources and evaluation, 44(1), 159–180.

Gehring, J., Auli, M., Grangier, D., Yarats, D., & Dauphin, Y.N. (2017, July). Convolutional sequence to sequence learning. In International Conference on Machine Learning. 1243– 1252.

Gildea, D., & Jurafsky, D. (2002). Automatic labeling of semantic roles. Computational Linguistics, 28(3), 245–288.

Girvan, M., & Newman, M.E. (2002). Community structure in social and biological networks. Proceedings of the National Academy of Sciences, 99(12), 7821–7826.

Griffiths, T.L., & Steyvers, M. (2004). Finding scientific topics. Proceedings of the National Academy of Sciences, 101(suppl 1), 5228–5235.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735–1780.

Kalchbrenner, N., et al. Espeholt, L., Simonyan, K., Oord, A.V.D., Graves, A., & Kavukcuoglu, K. (2016). Neural machine translation in linear time. arXiv preprint arXiv:1610.10099.

Kim, Y., Denton, C., Hoang, L., & Rush, A.M. (2017). Structured attention networks. In International Conference on Learning Representations. arXiv:1702.00887

Kingsbury, P., & Palmer, M. (2002). From treebank to propbank. Language Resources & Evaluation. Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02).

Leicht, E.A., & Newman, M.E. (2008). Community structure in directed networks. Physical Review Letters, 100(11), 118703.

Li, P.J., Lam, W., Bing, L., & Wang, Z. (2017). Deep Recurrent Generative Decoder for Abstractive Text Summarization. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2081–2090.

McDonald, R., Pereira, F., Kulick, S., Winters, S., Jin, Y., & White, P. (2005, June). Simple algorithms for complex relation extraction with applications to biomedical IE. In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL05). 491–498.

Mintz, M., Bills, S., Snow, R., & Jurafsky, D. (2009, August). Distant supervision for relation extraction without labeled data. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP. 1003–1011.

Pons, P., & Latapy, M. (2005, October). Computing communities in large networks using random walks. International symposium on computer and information sciences. ISCIS 2005: Computer and Information Sciences - ISCIS 2005, 284–293.

Ramage, D., Manning, C.D., & Dumais, S. (2011, August). Partially labeled topic models for interpretable text mining. In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 457–465. 457–465.

Schmidhuber, J. (2001). Gradient Flow in Recurrent Nets: The Difficulty of Learning Long-Term Dependencies. Wiley-IEEE Press.

Sethy, A., & Ramabhadran, B. (2008). Bag-of-word normalized *n*-gram models. In Ninth Annual Conference of the International Speech Communication Association. 1594–1597.

Shannon, C.E. (1948). A mathematical theory of communication. The Bell System Technical Journal, 27(3), 379–423.

Small, H., Boyack, K.W., & Klavans, R. (2014). Identifying emerging topics in science and technology. Research policy, 43(8), 1450–1467.

Swampillai, K., & Stevenson, M. (2011, September). Extracting relations within and across sentences. In Proceedings of the International Conference Recent Advances in Natural Language Processing 2011. 25–32.

Talley, E.M., Newman, D., Mimno, D., Herr, B.W., Wallach, H.M., Burns, G.A., ... & McCallum, A. (2011). Database of NIH grants using machine-learned categories and graphical clustering. Nature Methods, 8(6), 443–444.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems. 5998–6008.

Velden, T., Boyack, K.W., Gläser, J., Koopman, R., Scharnhorst, A., & Wang, S. (2017). Comparison of topic extraction approaches and their results. Scientometrics, 111(2), 1169–1221.

Wallach, H.M. (2006, June). Topic modeling: Beyond bag-of-words. In Proceedings of the 23rd international conference on Machine learning, 977–984.

Yin, W., Schütze, H., Xiang, B., & Zhou, B. (2016). Abcnn: Attention-based convolutional neural network for modeling sentence pairs. Transactions of the Association for Computational Linguistics, 4, 259–272.

Zeng, D.J., Liu, K., Lai, S., Zhou, G., & Zhao, J. (2014, August). Relation classification via convolutional deep neural network. In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. 2335–2344.

Zhang, Y., Lu, J., Liu, F., Liu, Q., Porter, A., Chen, H., & Zhang, G. (2018). Does deep learning help topic extraction? A kernel *k*-means clustering method with word embedding. Journal of Informetrics, 12(4), 1099–1117.