國立中山大學資訊管理學系

碩士論文

Department of Information Management

National Sun Yat-sen University

Master Thesis

基於在線式深度非負自編碼的主題演進及分散度探索

Topic Evolution and Diffusion Discovery based on Online Deep Non-negative Autoencoder

研究生：洪紹銘

Shao-Min Hung

指導教授：康藝晃 博士

Dr. Yihuang Kang

中華民國 109 年 6 月

June 2020

論文審定書

國立中山大學研究生學位論文審定書

本校資訊管理學系碩士班

研究生洪紹銘（學號：M074020005）所提論文

基於在線式深度非負自編碼的主題演進及分散度探索
Topic Evolution and Diffusion Discovery based on Online Deep Non-negative Autoencoder

於中華民國 109 年 7 月 30 日經本委員會審查並舉行口試，符合碩士學位論文標準。

學位考試委員簽章：

召集人 黃三益 黃三益　　委　員 康藝晃 康藝晃

委　員 李珮如 李珮如　　委　員 ＿＿＿＿

委　員 ＿＿＿＿　　委　員 ＿＿＿＿

指導教授(康藝晃) 康藝晃 (簽名)

摘要

隨著資料的儲存及取得越來越便利，我們可以方便的在網路上閱讀各式各樣的內容，在如此大量的資訊中，要完全了解、閱讀所有的內容是不太可能的，我們往往依賴著分類或搜尋關鍵字的方式找出想要獲得的資訊，也因為這個快速尋找的需求，大部分的網站都會提供關鍵字搜尋及詳細的分類，可是隨著資料的增長，持續依賴人工的方式分門別類想必是一件逐漸困難的事情，透過機器學習的技巧幫助我們分群、分類資料內容將會是趨勢。以文本資料來說，最著名的分類技巧為主題模型，透過求文章的近似分佈或矩陣分解的方式將大量資料轉換成主題，即便主題模型的成熟幫助了我們分類文章內容產生主題，但主題在現實生活中是會隨著時間的改變而出現或消失，如何在主題改變的過程中有完善的解釋，是這篇論文所要探討的主題模型技巧。

本篇論文提出新穎的主題模型技巧，稱之為深度非負自編碼，並且結合在線式模型，用以探索主題隨著時間的改變，使用的文本內容是機器學習的論文，實驗結果表明，透過我們提出的方法可以快速的找到各個時間點的主題，我們也提出以網路圖、熱點圖及計算距離的方法，透過這些方式達到解釋及探討主題演進的目標。

關鍵字: 主題演進; 主題模行; 主題擴散; 深度學習; 自編碼器; 網路分析

ABSTRACT

The storage type of books, newspapers and magazines has changed from tangible papers to digital documents. This phenomenon indicates that a large number of documents are stored on the Internet. Therefore, it is infeasible for us to review all information to find out what we need from these numerous papers. We need to rely on keywords or well-defined topics to find out our requirements. Unfortunately, these topics change over time in the real world. How to correctly classify these documents has been an increasingly important issue. Our approach aims to improve the problem of the topic model, which considers time. Considering that the inference method for the posterior probability is too complicated, so for simplicity, we use an autoencoder variant to build a topic model with shared weights at different times, called Deep Non-negative Autoencoder (DNAE). This model is a multi-layer structure, the evolution of topics in each layer is also a focus of this paper. Besides, we use generalized Jensen-Shannon divergence to measure the topic diffusion and use network diagrams to observe the evolution of topics.

Keywords: Topic Evolution; Topic Modeling; Topic Diffusion; Deep learning; Autoencoder; Network Analysis.

目錄

圖目錄

表目錄

1. Introduction

With the progress of science and technology, people are more and more dependent on consumer electronics, and many applications installed on these devices are related to machine learning algorithms, such as spam identification, commercials on social media, and recommendations on online stores. Among these machine learning applications, the most important thing for users is the recommendation system. For example, researchers choose the topics of articles (e.g., machine learning, text mining) which is their interest, and then select those types of papers to read based on the content recommended by the system. Choosing the interested topics has become more and more critical. However, as the explosion of information, people do not have enough time to explore everything, and it seems labor-intensive to create appropriate topic categories manually. Furthermore, we need a method to find out which topics will appear in the future. Therefore, if this work can be done through machine learning algorithms, it can help humans save a lot of time.

In order to easily and quickly understand a large amount of text in the same field. One approach used topic model(*Handbook of Latent Semantic Analysis*, 2007) to compress the texts into features called topics rather than using all words as labels. This practical use of compressing a larger number of articles into several topics is already very mature, such as Latent Dirichlet Allocation *(LDA)* (Blei, n.d.-b) and Nonnegative Matrix

Factorization *(NMF)* (Paatero & Tapper, 1994). The difference between these two algorithms is that the NMF reduces the matrix dimension and the LDA is a probability model. However, finding topics from the real-world text corpus is not enough. What is more important is to enable the topic to change when new information appears and try to explain how the topics are generated or disappeared over time. Considering the change in the topic, one approach is to observe the change of the term in similar topics over time, and the other is evaluate the degree of topic evolution.

In this thesis, we propose a method for building topic models through autoencoder (Baldi, n.d.)(Bourlard & Kamp, 1988)(Geoffrey E Hinton & Zemel, 1994) and find out the evolution of topics from its neural network. Via the advantages of multi-layer structure, the computational complexity of extra inter-topic correlations is reduced. In other words, the limitation of the posterior of topic model is omitted. We review the background of topic models and explore related articles in the topic model applications, and briefly introduce deep learning in Section 2. Our approach is covered in Section 3, which includes how the autoencoder can be applied to generate topics, and how the topics are interpreted In Section 4, the experimental results are presented. Finally, how to improve the method and related future work is presented in Section 5.

2. Background and related work

2.1 Topic model

In recent years, collecting documents is getting simpler and easier. There are many ways to figure out the connection between documents and terms including word cloud, term frequency *(tf)*, term frequency-inverse document frequency *(tf-idf)* and topic model (Silge & Robinson, 2017). Here, we focus on topic model to find the abstract "topics" in textual dataset. For example, a traditional and simple method of exploring topics such as Latent Dirichlet Allocation *(LDA)*(Blei, n.d.-b), which is the most popular generative probabilistic topic model. Because of the probabilistic nature of LDA and its sampling-based procedure, the corpus can be explained by each group. This group is a distribution that can represent similar words, which we called topic. Especially LDA is a kind of hierarchical probabilistic models, it can be easily applied to a variety of text corpora. However, our goal is to find the relationship between each topic in the same series from different time step. Discovering collection of documents and considering its topic evolution in each time interval has been an important research in recent years. For the most part, LDA-based methods do not consider the time order of documents (Blei, n.d.-a). But the number of documents accumulate over time, it is not a fixed set. As we know, time series modeling focuses on continuous data, while topic models are designed for categorical data. Consider time series topic models, one experiment is Topics Over Time *(TOT)* (Wang & McCallum, 2006), the other is Dynamic Topic Model *(DTM)* (Blei &

Lafferty, 2006). Both of them are the extensions of the topic model. Improving the topic model to explain is one of the studies that often appears. We will explain several methods of extension topic model next.

2.2 Time series topic model

Time series topic models present the low-dimension structure change over time. TOT models include temporal information to build topic model. This model introduces continuous time information into the generation model, which generates topic change trends, but it cannot expand new data and must be remodeled. On the other hand, DTM observing the topic evolution in a sequential documented corpus by using Gaussian time series and variational inference algorithms. To put it briefly, this modus splits document with a fixed interval (e.g. year) and supposes topic in time step *(t)* evolves from the one topic of previous time step *(t-1)*. That means topic model can take into account the time factor through gives a complete posterior computation. Both of them are LDA-based methods, and they already have successful applications. Actually, topic modeling algorithms can be not only probabilistic based, but also based on matrix factorization method such as Nonnegative Matrix Factorization *(NMF)* (Lee & Seung, 1999).

Again, our target is to find the topics by timestamp with ordering of documents in the collection. To explore topic, matrix factorization techniques decompose the input matrix into multiple low-rank submatrices, and this property can be used to find topics.

In this thesis, we adopted the concept of NMF, which can be exploited for topic modeling. The non-negative constraint of NMF help understand the topic components via lower-rank matrices. For example, let *V* be an *n* by *p* non-negative document-term matrix. Such that

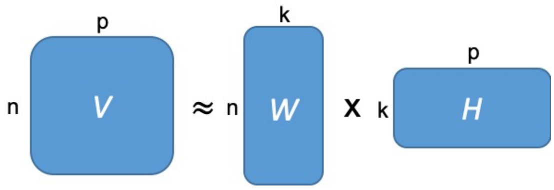

Figure 2-1: The architecture of the NMF

where the basis mmatrix *W* is *n* by *k* and coefficient matrix *H* is *k* by *p*. NMF attempts to reach the best approximation between *W H* and the original matrix *V* by minimizing the Frobenius norm ||. ||F as

$$minf(W,H) = \frac{1}{2}\|x - WH\|_F^2, s.t. W \geq 0, H \geq 0 \ ,$$

where *k* < *min(n,p)* and all elements in *W* and I are also non-negative. This kind of matrix compression presents the advantage of part-based. NMF have been widely used, such as face recognition, music transcription of signal processing, and topic modeling. Researchers have been using NMF as the base of dynamic topic models. Here is a method using NMF for dynamic topic modeling to Political Agenda of the European Parliament (Greene & Cross, 2016). Another method exploring the topic diffusion of Machine

Learning articles by constraining NMF (Kang et al., 2018). Both methods have a good way to explain the changes in the theme over time. Once again, it proves that a good topic model, whether it is LDA or NMF, can generate an easy-to-understand topic evolution through reasonable posterior methods.

2.3 Multi-layer topic model

To sum up, time series topic model is a kind of model extension method. Differentiate according to which base models is used, topic modeling can be roughly divided into two categories, probabilistic model and matrix factorization model. Both of them can be built with multi-layer structure. Not only each timestamp has evolution. Hierarchical-based method is another concept of model extension, in order to understand the evolution of the topic in the same timestamp. There is an approach present the nested Chinese restaurant process in hierarchical LDA-based method *(hLDA)* (Thomas L. Griffiths et al., 2004). This research proves that multi-layer structure can be an effective tool in text corpus. The other is the NMF-based method, which presents the evolution of a topic (Tu et al., 2018)(Song & Lee, 2013). The method of hierarchical NMF *(hNMF)* , which is different with NMF because it continues decomposing the coefficient matrix *(H)*, such that:

$$V \approx W_1 H_1$$
$$H_1 \approx W_2 H_2$$
$$\dots$$

$$H_{n-1} \approx W_n H_n$$

where H_1 can explained as detailed topic, H_2 as the rough topic and so on. In addition, there are many related literatures on multi-layer models that sufficient to prove the value of multi-layer methods. Considering the multi-layer model, Deep Learning is one of the hottest methods in recent years. It is known for its powerful ability to extract features and representation learning technology.

2.4 Deep Learning

Deep learning is a kind of machine learning which is first implemented by deep neural network *(DNN)* (LeCun et al., 2015). This is the way to learning representation from data . It uses multi-layer structure to enable algorithms to extract representation that are more abstract and higher level from the original input data. Therefore, deep learning has become so popular in recent years not only because of the advances in hardware technology, but more importantly because these layers of feature uses unsupervised or semi-supervised representation learning rather than human experts. For this reason, it is used in a variety of fields and has much higher accuracy compared to conventional machine learning algorithms , such as computer vision, speech recognition, and natural language processing.

If we consider the topic as a feature of an article, we will look forward to finding the topic through deep learning and try to understand how the topic evolves by each layer. However, deep learning has many ways to achieve. Among the method, the autoencoder in unsupervised learning is a model that conforms to the topic model. Because the final goal of autoencoder is to minimize the discrepancy between the input data and its reconstruction. The composition of autoencoder can be divided into encoder and decoder. The encoder reduces the data dimension and the decoder restores the dimension identical with the original input data (G. E. Hinton & Salakhutdinov, 2006). Here is a study using the improved Autoencoder and successfully applied to community detection. Their approach is Deep Autoencoder-like Nonnegative Matrix Factorization *(DANMF)* (Ye et al., 2018), they used NMF as pre-train model and apply this pre-train output as weight of decoder part of autoencoder model. According to the article, this approach can explain more nodes in community detection. This result is applied to the topic model, which may also be able to explain more topics, such as the relationship between the topics of multiple layers. In terms of implementation, this deep neural network is much simpler than the Hierarchical-based method mentioned in the previous section.

2.5 Online Learning

After understanding that deep learning can reduce dimensionality, how to make deep learning consider the time factor is our next step. In section 2.2 we have introduced a topic model containing time, and deep learning model training can also be combined with time, one of which is called online learning. The online model is a mathematical model that can track and mirror the model in real time. It realizes the model update over time with automatic adaptability. Figure 2 illustrates the data slice according to time. At the beginning of the model training, the weights of the first period will be used to initialize the weights, and the weights will be fixed after the model fitting. When the data for the next period is obtained, it can be put into the same model for training to update the weights. With this approach, the results of model training can be easily combined with time. The benefits of using online learning not only can be quickly trained with the model, but also used in topic evolution. It is also more convenient for us to find topic evolution without the need to calculate the similarity between topics.

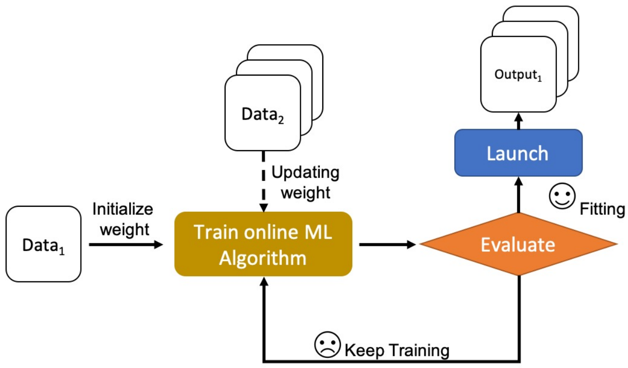

Figure 2-2: The architecture of the online learning

3. Methodology

In order to more easily find the topics and explain the evolution of the topics, we propose a model based on Autoencoder, called Deep Non-negative Autoencoder *(DNAE)*. In this chapter we explain how we deal with text data and our model architecture. Section 3.1 explains how to use the Autoencoder-compressed matrices to generate topics and explains the benefits of using this method. However, our approach hopes to find the differences between topics in different points of time, so we will explain in Section 3.2 how to use our method combined with Online Learning to make the topic model consider the time factor. After successfully finding the topics of each period, we will introduce how we explain the topic evolution by Jensen-Shannon divergence in Section 3.3.

Before introducing our method, we simply explain the preparation of the data. To finding topics, we usually use a large text corpus, including n documents and a well-defined domain-specific dictionary with m terms. We denote V as a non-negative document-term matrix, which includes X non-negative document-term vectors. Every vector records the frequency of terms in each document. Figure 3 shows the notations that we use when describing the data set.

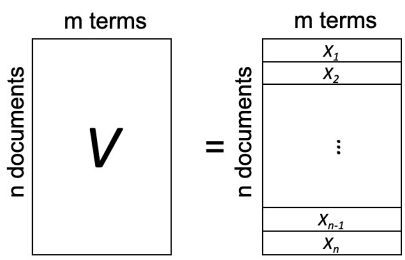

Figure 3-1: The notations of the dataset.

3.1 Topic model based on Autoencoder

Autoencoder is a dimensionality reduction model similar to NMF or SVD, so we think it can be used to build a topic model. Our method was inspired by hNMF, if we remove the autoencoder bias and add non-negative constraints on the weights, autoencoder can be more similar to hNMF. As Figure 4 shown, consider $H_1, H_2, \ldots H_n$ as coefficient matrices and $W_1, W_2, \ldots W_n$ as basis matrices. Matrices compressed by the Autoencoder will be like the compressed matrices from the hNMF we introduced in Section 2.3. Another similarity is both of them are trained to minimize reconstruction error of the raw input, such that:

$$\|X - X'\|_F^2 = \|X - XH_1H_2 \ldots H_nH_n' \ldots H_2'H_1'\|_F^2 \text{ , s.t. } H \geq 0, n > 0$$

where n is the number of hidden layers and H_n is a non-negative coefficient matrix used to encode/decode the document-term vector X. It is worth mentioning that autoencoder is a batch training method rather than NMF trains with matrix of all corpus. In other words, autoencoder compresses one vector at a time, so X here is a vector of document term matrix (V).

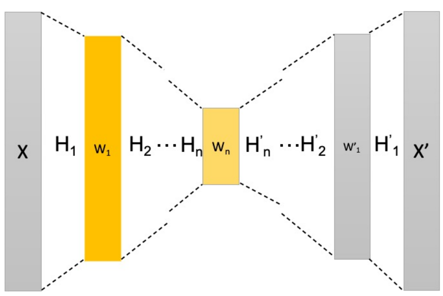

Figure 3-2: The architecture of the Autoencoder

Several studies have shown that features can be successfully found when using autoencoder to compress data dimensions (Kang et al., 2019). We believe that the corpus can successfully find features (topics) in the same way, and we have made some modifications to improve interpretability. Since the analysis of text sentiment is not included in our method, the negative relationship between words and topics cannot be explained. To explain the relationship between the topics, our method does not use any non-linear functions. The reason is that training neural network through a linear function will keep the output of each layer positive. We also constraints the weights to non-negative numbers. Both adjustments use to increase the terms interpretability in topic, and the other modification is to explore the evolution between topics. In the process of topic compression, we do not need the bias to strengthen or weaken the input value, so the bias will be removed during the training process.

The above describes the method of applying the topic model to autoencoder, as well as the restrictions we added. Here we consider how the document-term matrix *(V)* works in autoencoder. To be more specific, we use a two-layer autoencoder as an example. Figure 5 shows the process of finding the topic by dimension reduction of the matrix. It can be obviously seen from the picture that only the column dimension is reduced in the process of the autoencoder compressing the matrix. This is because the autoencoder training process obtains documents in batches instead of compressing the entire document-term matrix (NMF). In this compression process, the neural network will continue to do backpropagation to generate neurons *(W_1, W_2, W_1')*, and update the weights *(H_1, H_2, H'_2, H'_1)* until the restored matrix *(V')* similar to the original matrix *(V)*. After the model fitting, we can obtain the subtopic-term matrix *(H_1)* in the first layer and the subtopic-topic matrix *(H_2)* in the second layer. When these two matrices are multiplied, we can find the topic-term matrix with autoencoder.

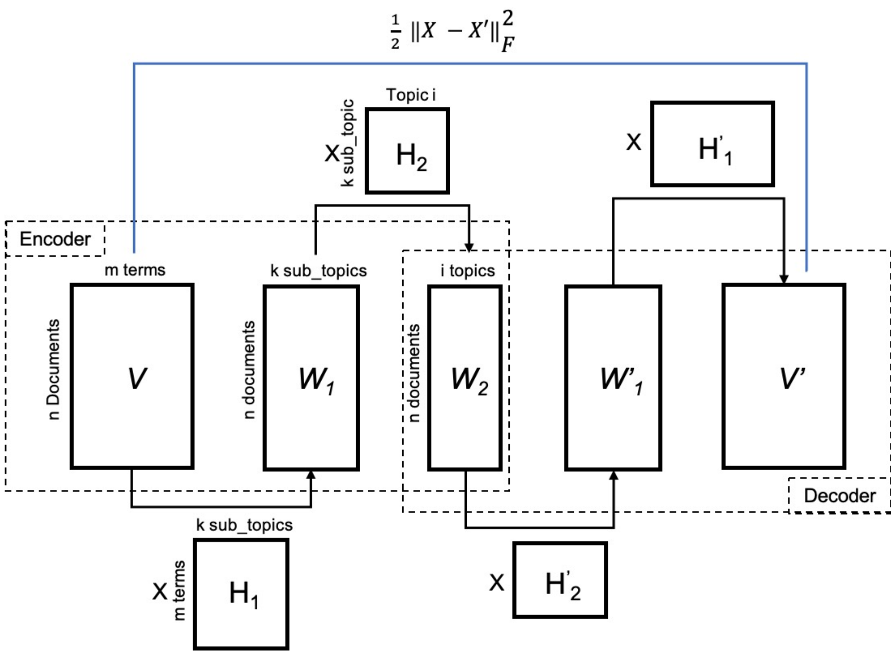

Figure 3-3: The architecture of the DNAE with document-term matrix

3.2 Online Deep Non-negative Autoencoder

In this section we will combine the improved autoencoder DNAE that we talk about above and the Online Learning introduced in Section 2.5. Figure 6 shows the online DNAE training process. V represents the document-term matrix, and it is cut into t pieces according to the time period. In the first period V_1 will initialize the weights according to the training method and use RMSE as a function to measure the original matrix V and the reconstruction matrix V' to obtain the final weight. After training, the topic-term matrix O_1 can be get by multiplying the weight of each layer as we described in the previous section. Then keep using the data of different time periods $(V_2, V_3, \ldots, Vn)$ to update the weights, and we can get the following output of topic-term matrix $(O_2, O_3, \ldots, O_n)$ of each time period. Through this approach, we can effectively combine the topic model of deep learning training with time to achieve our ultimate goal to discussing topic evolution.

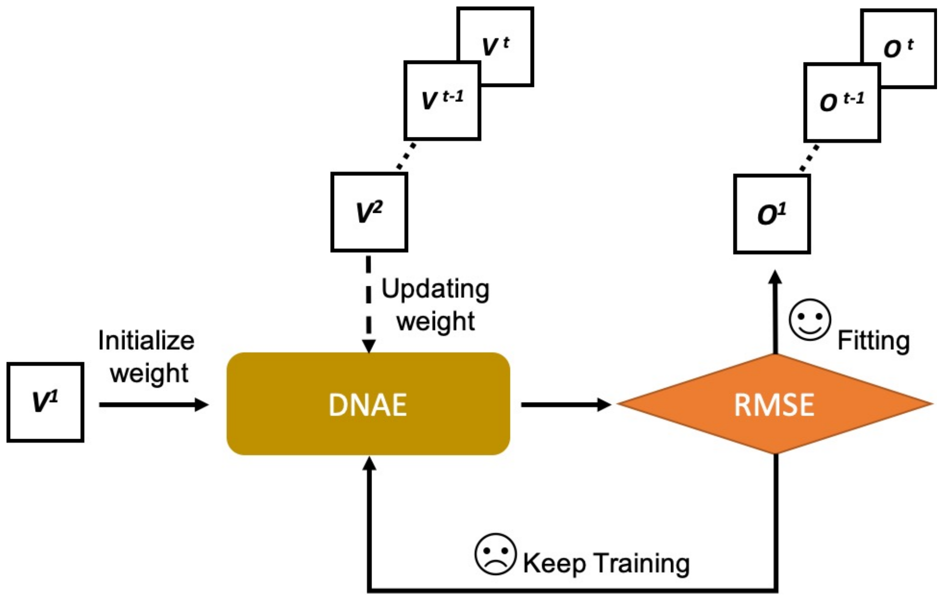

Figure 3-4: The architecture of the Online Deep Non-negative autoencoder

3.3 Evaluation of topic diffusion

When we use online DNAE successfully find topics in different time periods, the next step is to observe the term diffusions in topics. First step is normalizing the weight of term that found in topic-term matrix according to each topic, so that the summation of each term in topic equal to one. With this operation, we can consider the conditional probability of term in the topics (i.e. $P(topic_k|term_i)$), then compare the probabilities of each period to explain whether the term in the topic is different or not. Table 1 shows an example of topic-term matrix that normalize the weight of terms to one in each topic. In this example, suppose we have three terms and three topics.

Table 3-1: An example that probability distribution of terms and topics

$P(topic_k \mid term_i)$	Topic 1	Topic 2	Topic 3
Term 1	0.2	0.3	0.5
Term 2	0.4	0.4	0.2
Term 3	0.33	0.33	0.33

Next, we use generalized Jensen-Shannon divergence (D_{GJS})(Grosse et al., 2002)(Kang & Zadorozhny, 2016) as the basis for determining whether the term is diffusion in topic. The D_{GJS} is defined as:

$$D_{GJS}(P_1, P_2, \dots, P_t) = H(\sum_{i=1}^{t} \pi_i P_i) - \sum_{i=1}^{t} \pi_i \, H(P_i)$$

where π_i is the weight for each discrete probability distribution. The notation *t* use to define different time of term's probability in same topic. In the algorithm, *H(x)* is *k-ary* Shannon entropy defined as:

$$H(x) = -\sum_{i=1}^{k} P(X_i) log_k P(X_i)$$

Using this method, we can observe the term diffusions in topic with different slices (e.g. The distance of term *i* from data slice *t* and from data slice *t+1*). We also use a statistical significance threshold of D_{GJS} to evaluate the degree of topic diffusions, which defined as:

$$D_{GJS|k,t} \approx \frac{X^2_{df,1-\alpha}}{2N\ (ln_k)}$$

where *df = (k – 1) (t – 1)* is the degree of freedom, *α* is the statistical significance level (usually 0.05 or 0.01), and *N* is the total number of cells (*k* by *t*) used in calculating the Chi-square statistic χ^2 in different times.

3.4 Visualization of topic evolution

In this section, we present the relationship between topics in a visual way. We use Network(Ognyanova, n.d.) to present a more intuitive relationship diagram. In this way, we can quickly observe the repeatability of the text among topics. The main visualization will be divided into the following categories. The first one is to check whether the similarity of the topics in the same year is high or not. That is, the distance between the topics is not close. The second is to observe the similarity of the terms in same topic with each time period. In this way, we can clearly observe the textual changes in a certain period of time.

Drawing network diagram will mainly divide the topic and term into two Nodes *(N),* and the relationship between the nodes will be recorded with Edges *(E)*. Table 2 lists the contents of the Node's data table. All of terms *(i)* and topics *(k)* will be recorded in this data table. Then record the relationship between the nodes through the edge data table as shown in Table 3. The edge table will record which topic the term belongs to and its probability (i.e. From $term_i$ to $topic_k$ and probability is $P(topic_k|term_i)$). The higher the probability, the thicker the line between the nodes. We will show the complete network diagram in the next chapter.

Table 3-2: The data frame of Nodes

id	shape	label	color.background
$term_1$	circle	$term_1$	lightblue
…	…	…	…
$term_i$	circle	$term_i$	lightblue
$topic_1$	box	$topic_1$	yellow
…	…	…	…
$topic_k$	box	$topic_k$	yellow

Table 3-3: The data frame of Edges

from	to	value
$term_1$	$topic_1$	$P(topic_1\|term_1)$
…	…	…
$term_n$	$topic_k$	$P(topic_k\|term_n)$

3.5 Topic Evolution and Diffusion Discovery based on online DNAE

Through the D_{GJS} measurements and network diagrams, we can discover the diffusion of term in the topics and observe the relationship between topics, and thus achieving our research goals. The overall workflow of our analysis method is shown in Figure 7. To reiterate again, the topic diffusion discover is to observe whether the term has changed over time from the probability distribution. For example, a vocabulary may change over time, and the probability of appearing in different topics may also increase, rather than being limited to the same topic. An example of our life, the word "Apple" has a high chance of appearing in the topic of food in the past, but in the era when gravity was discovered in Newton, it may appear in the academic topic. As for now, the appearance of this word has a high probability to represent the symbol of entertainment or electronic products. Exploring this vocabulary with different meanings over time is the essence of this paper. We will explain more examples of the topic evolution and term diffusion with machine learning related paper in the next section.

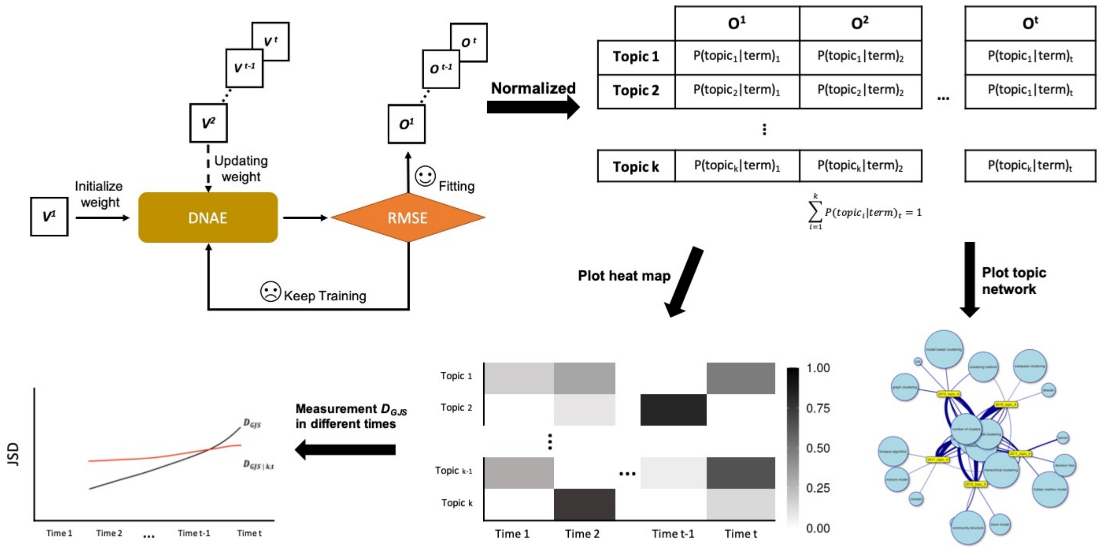

Figure 3-5: Workflow of topic evolution and diffusion discover based on Online DNAE

4. Experiment

In this section, we present the result of finding topic through DNAE. All of our work was implemented by R 3.6.0 with package Keras and ggplot2. We collect machine learning papers from arXiv.org (https://arxiv.org) through web API to evaluate the feasibility of DNAE. This dataset contains papers which main category is stat.ML from 2007/01 to 2019/12. Figure 8 shows the number of papers for each year. In this corpus, we have 31,904 documents between 2007 and 2019, then the document content is analyzed by word segmentation to sort out 18,814 keywords. In order to filter these keywords, we use tf-idf to remove redundant words and create mapping table by domain experts to merge the corresponding terms. With the pervious operations, we do not need to remove stemming and stopwords. We will explain how to use this data to train the model in the next section.

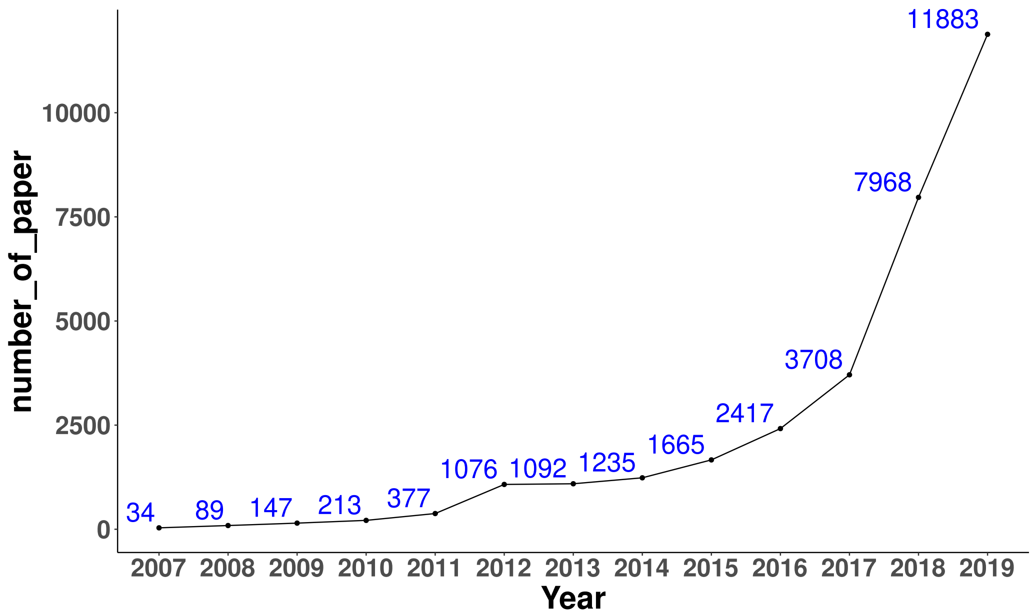

Figure 4-1: The number of papers in 2007 ~ 2019

As we discussed earlier, in order to understand the evolution in the topic between each layer, we use two-layer DNAE to train the model. In the early stage of model training, the data from 2007 to 2011 is used as the initial weight of the topic. When the topic weight of the first period is available, the data will be added one period for each subsequent training. In addition, we use online learning to observe the term diffusion in the same topic in different times. With online learning, we do not need to compare topic that each model generates. Appendix A shows the first ten words in each topic from 2011 to 2016 by our methods. Here, we use the same way to pre-process the data, which has similar results compared with previous research (Kang et al., 2018). However, in addition to similar results using our method, it also reduces a lot of training time. Next, we will show the results of using DNAE and newer data.

4.1 Online topic model with DNAE

In order to observe the topic changes in the past decade, we divide the data into five slices by every two years from 2011. In addition to the first slice of data is from 2007 to 2011 years, the rest are dividing every two years. From the Figure 8, we can easily observe that the number of data significantly increases a lot over time. If such slices are directly applied to online learning, it may cause catastrophic interference problems (McCloskey & Cohen, 1989). Faced with this situation, it will make the topic variability in each period too large, making it difficult to compare, so we will combine the data of the previous year of the period during the training process. Here we use the probability ranking of the top 10 vocabularies in each topic as the present of topic. Considering that we have to look at the evolution of the topic, this means the term of the topic may vary over time, so there is no intention to name the found term group (topic). As in the table attached to Appendix B. Our experiment finds 10 topics and list the top 10 vocabularies sort by probability of each topic to compare the differences between different years. Figure 9 shows the result of topic 6. It can be clearly seen that this is a topic related to word processing. Based on this result, we can understand that researchers on this topic mostly discussed topic models from 2012 to 2015 and the discuss terms of this topic turned into neural network-related applications in recent years. However, not all topics have such obvious changes. For example, it can be clearly seen in Figure 10 that there is not much difference in the terms of the topics discussed in each period.

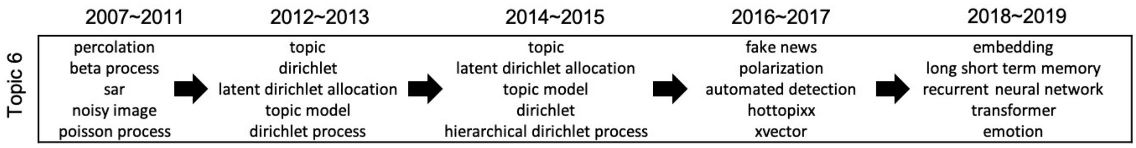

Figure 4-2: The term evolution in topic 6

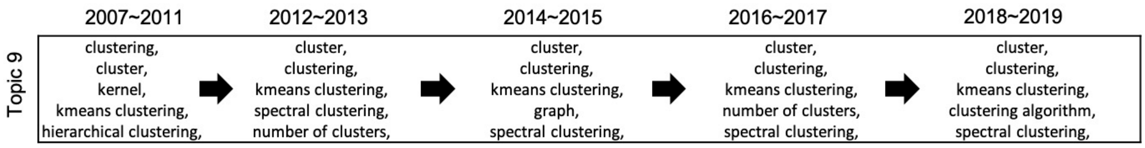

Figure 4-3: The term evolution in topic 9

4.2 Topic evolution and diffusion with DNAE

In addition to using term tables to observe changes in vocabulary over time, we hope that we can more easily observe the differences of topic in each period, so we use network diagrams as a medium for visual painting. With this graph, we can see the repeated vocabulary between topics, and observe the differences in the words of adjacent years. All network diagrams in the experiment are implemented using the VisNetwork package. The network diagram also makes it easy for us to choose the vocabulary to observe the topic relationship. In Figure 11, we choose the word "cluster" to observe the difference in topic 9, and we can find that some important words will cover all time periods. There will also be vocabulary related to the topic that has only been discussed in recent years. For example, the term general adversarial network (GAN) is a term that has been discussed frequently in recent years. This vocabulary appears in Figure 12 to discuss the topic 7 related to the graphical model, and it has appeared in recent years.

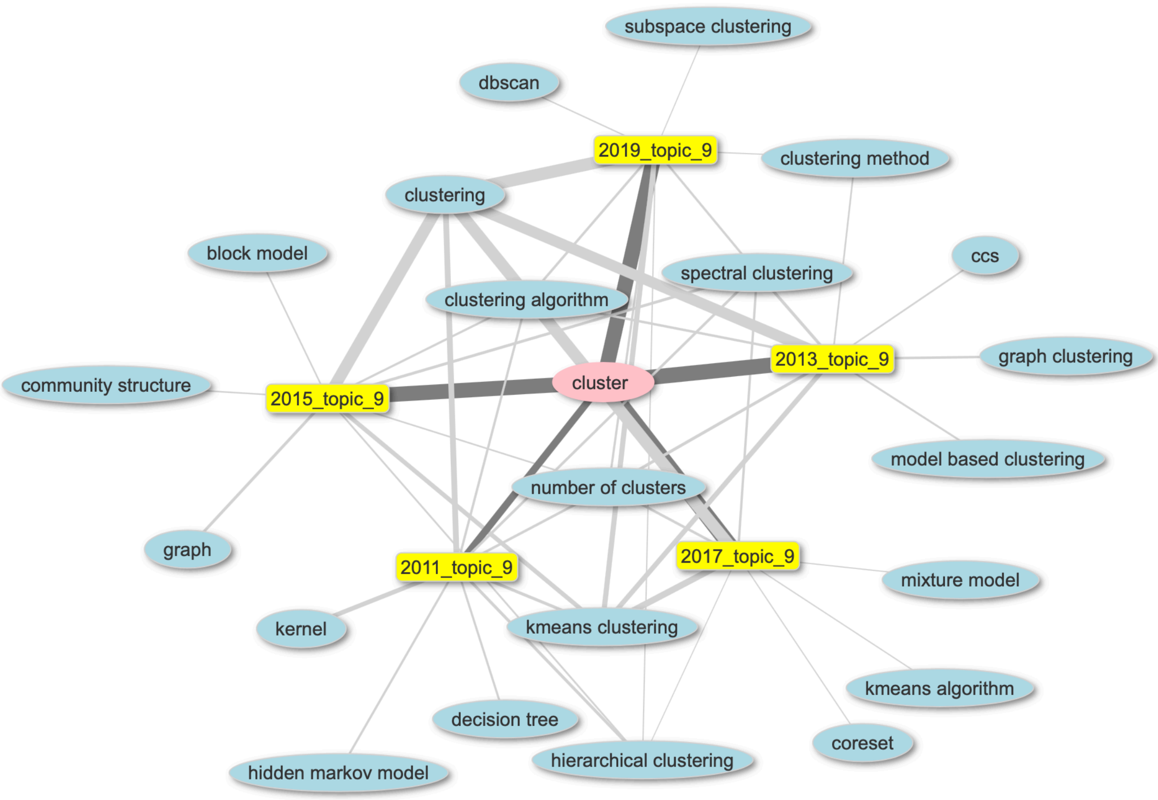

Figure 4-4: The network of topic 9 with each period.

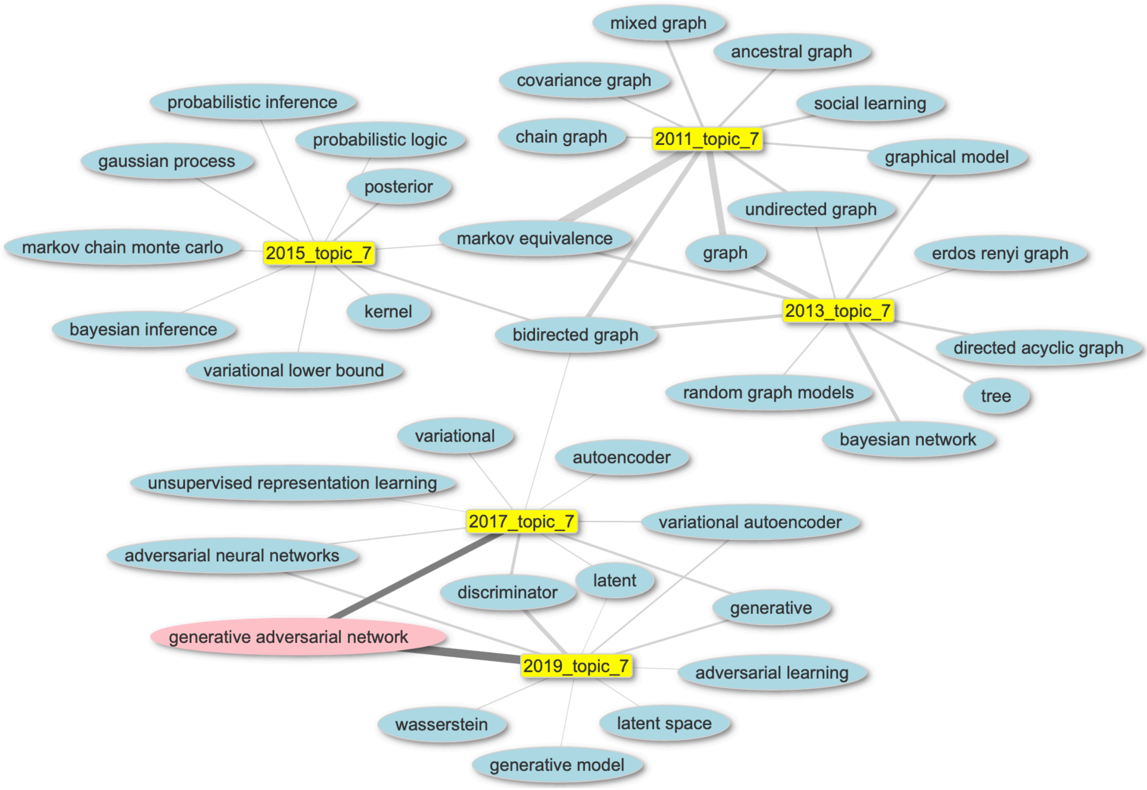

Figure 4-5: The network of topic 7 with each period.

4.3 Term evolution with DNAE

According to the top-10 terms table and topic network map generated in above, most topics can exhibit clear topic of machine learning techniques. However, the vocabulary contained in certain topics is still scattered, which makes the topic difficult to explain. As a result of this, some relevant studies have pointed out that the existing topics may evolve or new topics may appear (Stevens et al., 2012)(Greene et al., 2014). In order to have a clear method to judge whether the importance term of topics is change than before, we use the D_{GJS} introduced in Section 3.3 to calculate the distance between vocabulary in different years of the same topic.

Through this measure, the diffusion or evolution of term in the topic can be observed more precisely. For example, the topic 9 is related to clustering technical of machine learning as we discussion above, and one of the most commonly used words in articles discussing clustering is "k means cluster". We can see in Figure 13 that k means cluster is concentrated on topic 9, and there is no sign of diffusion. We call such words as narrow term.

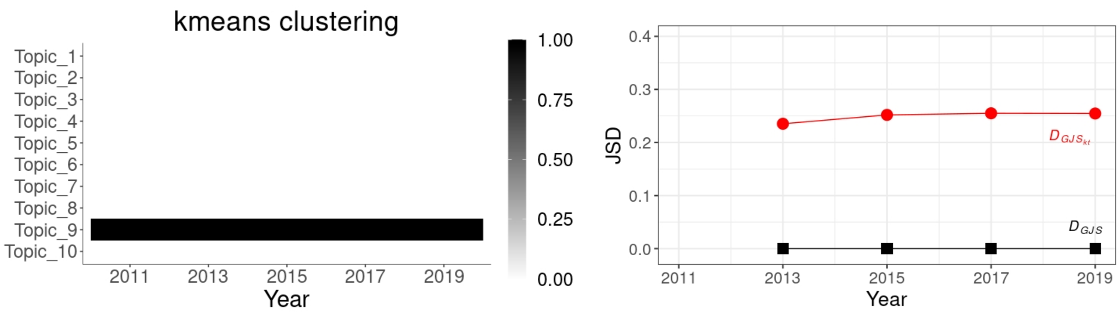

Figure 4-6: Narrow term “ k means clustering”

Then we show the result of "nonnegative matrix factorization" in Figure 14. NMF can be observed from the heatmap that the discussion of this vocabulary is not limited to a certain topic but is also spread across various topics. We call this vocabulary a broad term. And from the figure of calculating Jensen-Shannon divergence, we can observe that the term diffusion has gradually converged.

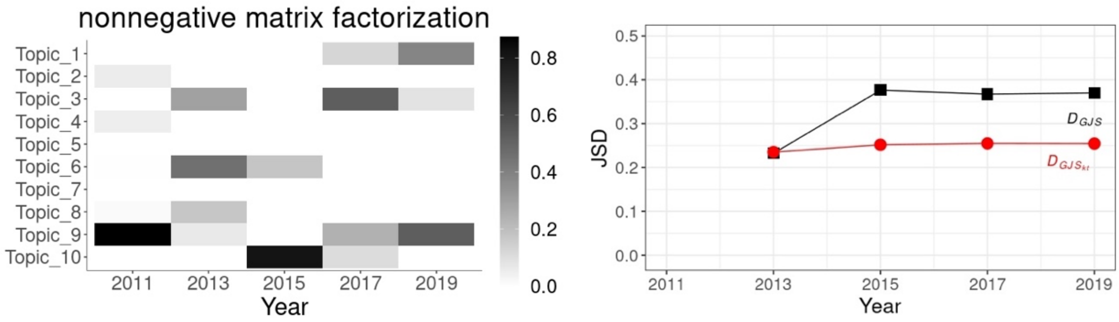

Figure 4-7: Broad term " nonnegative matrix factorization"

Another thing that can be observed from the picture is the degree of vocabulary diffusion. Through the JSD diagram, we can set the threshold value according to statistically significant level. If the calculation result exceeds this threshold, it can indicate that the vocabulary is spreading. The vocabulary "tensor factorization" in Figure 15 is a term of this type. From the figure we can observe that the term's JSD is decreasing and approaching the threshold. We call the vocabulary of this phenomenon convergent term. However, it can also be seen from the heat map that this vocabulary has been classified in topic 1 in recent years.

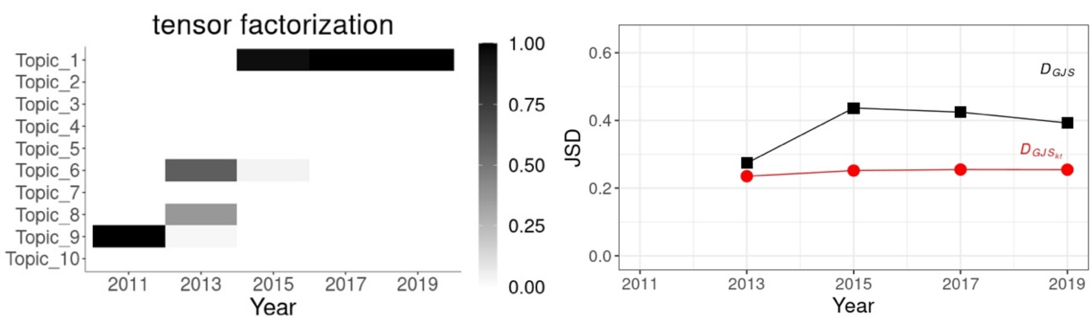

Figure 4-8: Convergent term " tensor factorization"

In addition to Convergent term, the other is the continually spread vocabulary in Figure 16, which we call divergent term. In this example, “latent dirichlet allocation” has only been discussed intensively in the topic of topic modeling (topic 6) in the past. However, in recent years, it has been gradually distributed on different topics. This means that LDA is being discussed in more and more topics. Therefore, we call these words discussed by more and more topics as divergent term.

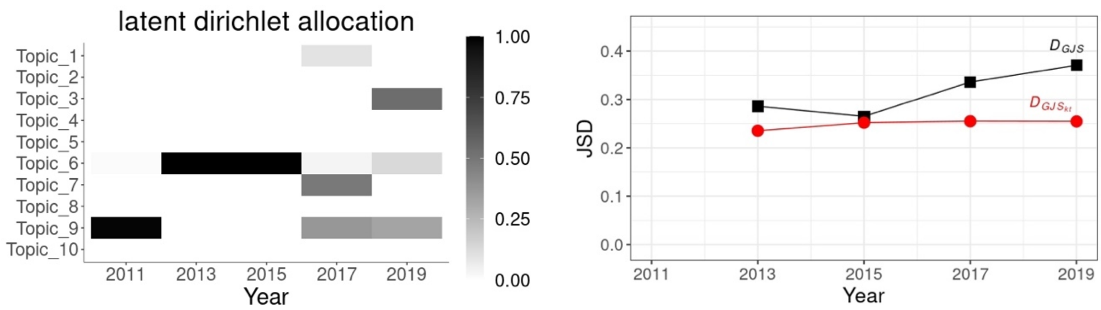

Figure 4-9: Divergent term “ latent dirichlet allocation”

5. Discussion

Whether the topic generated in the topic model is good or not is a question that has been discussed for a long time. The interpretation of a given topic with its components is a subjective matter, so we did not specifically compare the differences between DNAE with other topic models. In this thesis, we just explained the topic that generated by DANE, and did not discuss the pros and cons of each topic model. If there is a clear and well-recognized comparison method in the future, we can further discuss the quality of the topic of DNAE. In addition to judging the quality of the topic, we also list several issues that deserve more thought and discussion.

Choose *k* in each layer

The first issue is the number of topics generated (how many *k* should be appropriate). If the number of selected topics is too small, a rough topic will be generated, and if the number of topics is too large, redundant topics will be generated. This problem is also a hot issue that has been discussed for a long time in the topic model. Although we can use calculate Perplexity to choose the number of topics (T. L. Griffiths & Steyvers, 2004). But our method not only generates topics also tries to generate sub-topics. Therefore, we still need to find a way to define the appropriate number of topics for every year and each layer.

Topic evolution

Our experiments explore the topics that have been generated, but the topics may disappear with the year or be replaced by novel themes. It seems too simple to use the topic tree or network diagram to express the topic evolution. Although DNAE combined with online learning can allow novel topics to appear, but the disappeared topics still cannot be explained. It may be a good way to present the evolution of themes using a similar biological evolution diagram (*Phylogenetic Trees | Evolutionary Tree (Article) | Khan Academy*, n.d.) or evolutionary tree (Lake, n.d.).

Network diagram

We have successfully used network diagrams to represent the relationship between topics, but each network diagram relies on the topic model to generate weights and update them one by one. Considering the popularity of graph neural networks, if the topics generated by Online DNAE can be generated into network graphs (Zhou et al., 2019), then learn through graph neural networks. If this approach can learn the rules that the topic-term network changes over time, it must help explain the evolution of the topic.

6. Conclusion

According to our experimental results, it is successful for using deep learning method to build a hierarchical topic model with less overlapping. We called it Deep Non-negative Autoencoder *(DNAE)*. DNAE not only can find the topic well, but it can also be easily training with our suggestions. Therefore, we can use this method with online learning to observe topic diffusion and evolution in each period. Moreover, both of topic tree and network diagram are an easy-to-understand way and they can help check the relationships among different topics. DNAE is a flexible model that helps us track the topic with time. We believe that using our approach will build topic models with more diverse explanations.

7. Reference

Baldi, P. (n.d.). *Autoencoders, Unsupervised Learning, and Deep Architectures*. 14.

Blei, D. M. (n.d.-a). *Introduction to Probabilistic Topic Models*. 16.

Blei, D. M. (n.d.-b). *Latent Dirichlet Allocation*. 30.

Blei, D. M., & Lafferty, J. D. (2006). Dynamic topic models. *Proceedings of the 23rd International Conference on Machine Learning - ICML '06*, 113–120. https://doi.org/10.1145/1143844.1143859

Bourlard, H., & Kamp, Y. (1988). Auto-association by multilayer perceptrons and singular value decomposition. *Biological Cybernetics*, *59*(4), 291–294. https://doi.org/10.1007/BF00332918

Greene, D., & Cross, J. P. (2016). Exploring the Political Agenda of the European Parliament Using a Dynamic Topic Modeling Approach. *ArXiv:1607.03055 [Cs]*. http://arxiv.org/abs/1607.03055

Greene, D., O'Callaghan, D., & Cunningham, P. (2014). How Many Topics? Stability Analysis for Topic Models. *ArXiv:1404.4606 [Cs]*. http://arxiv.org/abs/1404.4606

Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. *Proceedings of the*

National Academy of Sciences, *101*(Supplement 1), 5228–5235. https://doi.org/10.1073/pnas.0307752101

Griffiths, Thomas L., Jordan, M. I., Tenenbaum, J. B., & Blei, D. M. (2004). Hierarchical Topic Models and the Nested Chinese Restaurant Process. In S. Thrun, L. K. Saul, & B. Schölkopf (Eds.), *Advances in Neural Information Processing Systems 16* (pp. 17–24). MIT Press. http://papers.nips.cc/paper/2466-hierarchical-topic-models-and-the-nested-chinese-restaurant-process.pdf

Grosse, I., Bernaola-Galván, P., Carpena, P., Román-Roldán, R., Oliver, J., & Stanley, H. E. (2002). Analysis of symbolic sequences using the Jensen-Shannon divergence. *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, *65*(4 Pt 1), 041905. https://doi.org/10.1103/PhysRevE.65.041905

Handbook of Latent Semantic Analysis. (2007). Routledge Handbooks Online. https://doi.org/10.4324/9780203936399

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the Dimensionality of Data with Neural Networks. *Science*, *313*(5786), 504–507. https://doi.org/10.1126/science.1127647

Hinton, Geoffrey E, & Zemel, R. S. (1994). Autoencoders, Minimum Description Length and Helmholtz Free Energy. In J. D. Cowan, G. Tesauro, & J. Alspector (Eds.), *Advances in Neural Information Processing Systems 6* (pp. 3–10).

Morgan-Kaufmann. http://papers.nips.cc/paper/798-autoencoders-minimum-description-length-and-helmholtz-free-energy.pdf

Kang, Y., Cheng, I.-L., Mao, W., Kuo, B., & Lee, P.-J. (2019). Towards Interpretable Deep Extreme Multi-label Learning. *ArXiv:1907.01723 [Cs, Stat]*. http://arxiv.org/abs/1907.01723

Kang, Y., Lin, K.-P., & Cheng, I.-L. (2018). Topic Diffusion Discovery Based on Sparseness-Constrained Non-Negative Matrix Factorization. *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, 94–101. https://doi.org/10.1109/IRI.2018.00021

Kang, Y., & Zadorozhny, V. (2016). Process Monitoring Using Maximum Sequence Divergence. *Knowledge and Information Systems*, *48*(1), 81–109. https://doi.org/10.1007/s10115-015-0858-z

Lake, J. A. (n.d.). *Reconstructing evolutionary trees from DNA and protein sequences: Parallnear distances*. 5.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, *401*(6755), 788–791. https://doi.org/10.1038/44565

McCloskey, M., & Cohen, N. J. (1989). Catastrophic Interference in Connectionist

Networks: The Sequential Learning Problem. In G. H. Bower (Ed.), *Psychology of Learning and Motivation* (Vol. 24, pp. 109–165). Academic Press. https://doi.org/10.1016/S0079-7421(08)60536-8

Ognyanova, K. (n.d.). *Network visualization with R*. 66.

Paatero, P., & Tapper, U. (1994). Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, *5*(2), 111–126. https://doi.org/10.1002/env.3170050203

Phylogenetic trees | Evolutionary tree (article) | Khan Academy. (n.d.). Retrieved July 2, 2020, from https://www.khanacademy.org/science/high-school-biology/hs-evolution/hs-phylogeny/a/phylogenetic-trees

Silge, J., & Robinson, D. (2017). *Text Mining with R: A Tidy Approach*. O'Reilly Media, Inc.

Song, H. A., & Lee, S.-Y. (2013). Hierarchical Representation Using NMF. In M. Lee, A. Hirose, Z.-G. Hou, & R. M. Kil (Eds.), *Neural Information Processing* (Vol. 8226, pp. 466–473). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-42054-2_58

Stevens, K., Kegelmeyer, P., Andrzejewski, D., & Buttler, D. (2012). Exploring Topic Coherence over Many Models and Many Topics. *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and*

Computational Natural Language Learning, 952–961. https://www.aclweb.org/anthology/D12-1087

Tu, D., Chen, L., Lv, M., Shi, H., & Chen, G. (2018). Hierarchical online NMF for detecting and tracking topic hierarchies in a text stream. *Pattern Recognition*, *76*, 203–214. https://doi.org/10.1016/j.patcog.2017.11.002

Wang, X., & McCallum, A. (2006). Topics over time: A non-Markov continuous-time model of topical trends. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '06*, 424. https://doi.org/10.1145/1150402.1150450

Ye, F., Chen, C., & Zheng, Z. (2018). Deep Autoencoder-like Nonnegative Matrix Factorization for Community Detection. *Proceedings of the 27th ACM International Conference on Information and Knowledge Management - CIKM '18*, 1393–1402. https://doi.org/10.1145/3269206.3271697

Zhou, J., Cui, G., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., & Sun, M. (2019). Graph Neural Networks: A Review of Methods and Applications. *ArXiv:1812.08434 [Cs, Stat]*. http://arxiv.org/abs/1812.08434

Appendix A. Top-10 words of 10 topics from 2011 to 2016.

	~2011	~2012	~2013	~2014	~2015	~2016
Topic1	association studies, graph, mallows model, adaptive learning, random geometric graph, graphical model, mixed model, transduction, markov, minimum error	graph, markov, latent, mallows model, dirichlet, graphical model, random geometric graph, association studies, log linear model, network	dirichlet, markov, latent, topic, bayesian, posterior, graph, graphical model, heteroscedastic data, network	dirichlet, latent, topic, markov, posterior, bayesian, variational, latent dirichlet allocation, markov chain, heteroscedastic data	dirichlet, topic, latent, markov, variational, posterior, latent dirichlet allocation, bayesian, markov chain, inference	topic, dirichlet, latent dirichlet allocation, latent, topic models, topic modeling, perplexity, heteroscedastic data, optimal splitting, dirichlet process
Topic2	distortion measure, distortion, self organizing map, law of large numbers, ddalpha, model compression, exchangeable feature allocations, model parallelism, petuum, design unbiasedness	distortion measure, distortion, self organizing map, law of large numbers, ddalpha, model compression, exchangeable feature allocations, model parallelism, design unbiasedness, conditional covariance selection	distortion measure, distortion, self organizing map, ddalpha, model compression, exchangeable feature allocations, design unbiasedness, conditional covariance selection, conic relaxations, open world recognition	distortion measure, distortion, self organizing map, ddalpha, exchangeable feature allocations, model compression, design unbiasedness, conic relaxations, speckle, grassmannian gradient descent	distortion measure, distortion, self organizing map, speckle, stochastic distances, ddalpha, compression, sample compression, biomedical spectroscopy, equitability	markov, posterior, variational, markov chain, monte carlo, bayesian, inference, latent variable, graphical model, hidden markov model
Topic3	meld, survival tree, conditional inference trees, recursive partitioning, markov equivalence, metropolis hastings algorithm, survival, random geometric graph, survival analysis, recursive estimation	meld, survival tree, conditional inference trees, recursive partitioning, metropolis hastings algorithm, markov equivalence, survival, smml, survival analysis, random geometric graph	smml, minimum message length, conditional inference trees, meld, metropolis hastings algorithm, evidence based medicine, exponential family, step functions, intelligent modelling, tuning parameter calibration	smml, minimum message length, conditional inference trees, metropolis hastings algorithm, evidence based medicine, exponential family, intelligent modelling, meld, tuning parameter calibration, hosting provider	smml, minimum message length, conditional inference trees, metropolis hastings algorithm, evidence based medicine, intelligent modelling, exponential family, tuning parameter calibration, hosting provider, approximate reinforcement learning	smml, minimum message length, conditional inference trees, mml, exponential family, approximate reinforcement learning, intelligent modelling, intrinsic dimensionality estimation, interactive graph mining, functional bahadur representation
Topic4	lasso, group lasso, adaptive lasso, group sparsity, sign consistency, sparsity, non convex, asymptotic analysis, selection consistency, lasso penalty	lasso, group lasso, non convex, adaptive lasso, sign consistency, variable selection, sparsity, cyclic coordinate descent algorithm, selection consistency, sparse	lasso, group lasso, non convex, sparse, sparsity, cyclic coordinate descent algorithm, screening, variable selection, hierarchical group lasso, compound decision theory	lasso, group lasso, screening, sparse, non convex, sparsity, cyclic coordinate descent algorithm, variable selection, compound decision theory, thresholding estimator	lasso, screening, group lasso, sparsity, sparse, non convex, convex, cyclic coordinate descent algorithm, variable selection, thresholding estimator	lasso, screening, convex, sparsity, rank, sparse, cyclic coordinate descent algorithm, covariance estimation, risk, non convex
Topic5	margin, adaboost, risk, boosting, rademacher, classifier, privacy, excess risk, vc dimension, rademacher complexity	adaboost, boosting, margin, classifier, risk, support vector machine, training, support vector, weak learner, logit	adaboost, boosting, ranking, classifier, support vector, margin, support vector machine, training, ensemble, calibration	ranking, classifier, support vector, adaboost, boosting, training, support vector machine, margin, ensemble, calibration	training, classifier, ranking, support vector, neural networks, support vector machine, dropout, adaboost, classification, network	recurrent neural networks, neural networks, lstm, dropout, network, training, machine translation, deep learning, convolutional neural networks, autoencoder

	~2011	~2012	~2013	~2014	~2015	~2016
Topic6	distortion measure, distortion, self organizing map, law of large numbers, conditional covariance selection, low rank transformation, open world recognition, primal dual splitting, structured signal, gbicp	distortion measure, distortion, self organizing map, conditional covariance selection, primal dual splitting, low rank transformation, open world recognition, law of large numbers, design unbiasedness, model parallelism	distortion measure, distortion, self organizing map, conditional covariance selection, open world recognition, low rank transformation, design unbiasedness, primal dual splitting, inverse prediction, ddalpha	distortion measure, distortion, self organizing map, conditional covariance selection, open world recognition, low rank transformation, design unbiasedness, ddalpha, primal dual splitting, inverse prediction	distortion measure, distortion, self organizing map, conditional covariance selection, open world recognition, ddalpha, low rank transformation, primal dual splitting, design unbiasedness, inverse prediction	adversarial, adversarial examples, privacy, perturbation, classifier, brain activity analysis, nonlinear manifold learning, johnson lindenstrauss lemma, robustness, class noise
Topic7	dendrogram, ultrametric, ultrametric topology, conflict analysis, hierarchical clustering, nearest neighbor algorithm, baire distance, information visualization, agglomerative, single linkage	dendrogram, ultrametric, ultrametric topology, hierarchical clustering, conflict analysis, clustering, agglomerative, baire distance, nearest neighbor algorithm, hierarchy	clustering, dendrogram, ultrametric topology, ultrametric, hierarchical clustering, spectral clustering, conflict analysis, information visualization, graph, agglomerative	clustering, ultrametric topology, graph, dendrogram, hierarchical clustering, spectral clustering, conflict analysis, ultrametric, information visualization, community	clustering, graph, ultrametric topology, hierarchical clustering, spectral clustering, conflict analysis, information visualization, community, dendrogram, subspace clustering	clustering, graph, ultrametric topology, hierarchical clustering, spectral clustering, conflict analysis, community, information visualization, similarity, dissimilarity
Topic8	regret, bandit, ucb, pure exploration, optimal allocation, regret analysis, peaking, bandit problems, repeated games, distributed learning	regret, bandit, pure exploration, ucb, optimal allocation, peaking, regret analysis, online convex optimization, repeated games, bandit problems	regret, bandit, peaking, ucb, optimal allocation, approximate sampling, repeated games, regret bounds, bandit convex optimization, emulation	regret, bandit, peaking, ucb, regret bounds, optimal allocation, repeated games, approximate sampling, bandit convex optimization, online	regret, bandit, ucb, regret bounds, peaking, repeated games, optimal allocation, online, approximate sampling, implicit updates	regret, bandit, ucb, online, regret bounds, feedback, exact discovery, contextual bandit, implicit updates, online learning
Topic9	regret, bandit, ucb, pure exploration, optimal allocation, regret analysis, peaking, bandit problems, repeated games, distributed learning	kernel, reproducing kernel, clustering, reproducing kernel hilbert space, hilbert space, local principal component analysis, gaussian kernel, spectral embedding, riemannian geometry, composite hypotheses	kernel, reproducing kernel, hilbert space, reproducing kernel hilbert space, riemannian geometry, gam, composite hypotheses, kernel learning, laplace operator, gaussian kernel	kernel, reproducing kernel, hilbert space, reproducing kernel hilbert space, riemannian geometry, kernel learning, composite hypotheses, laplace operator, gaussian kernel, weak topology	kernel, reproducing kernel, hilbert space, reproducing kernel hilbert space, gaussian process, gaussian kernel, riemannian geometry, composite hypotheses, kernel learning, laplace operator	kernel, reproducing kernel, hilbert space, support vector, reproducing kernel hilbert space, support vector machine, gaussian kernel, embedding, kernel learning, feature map
Topic10	sar, sar image, em simulator, bistatic radar, atr, synthetic aperture radar, scattering, synthetic database, time varying graph, right censored data	sar, sar image, bistatic radar, em simulator, synthetic aperture radar, atr, scattering, best arm identification, polarimetric sar, speckle	sar, synthetic aperture radar, bistatic radar, sar image, em simulator, speckle, scattering, atr, best arm identification, polarimetric sar	sar, synthetic aperture radar, bistatic radar, sar image, em simulator, speckle, scattering, atr, best arm identification, polarimetric sar	sar, synthetic aperture radar, bistatic radar, sar image, scattering, speckle, em simulator, atr, polarimetric sar, stochastic distances	sar, component analysis, scattering, principal component analysis, bistatic radar, synthetic aperture radar, speckle, sar image, atr, independent component analysis

Appendix B. Top-10 words of 10 topics from 2011 to 2019 slice by two years.

	2007 ~ 2011	2012 ~ 2013	2014 ~ 2015	2016 ~ 2017	2018 ~ 2019
Topic1	padic, dendrogram, gps trajectory, gradient reversal, prosody, wellbeing, oneclass classifier, load forecasting, ultrametric, neural processes	smml, padic, step functions, gradient reversal, prosody, adversarial machine learning, xvector, 3d unet, neural processes, gps trajectory	tensor, smml, padic, tensor decomposition, tensor factorization, tensor completion, tensor recovery, spectral norm, tensor rank,	tensor, padic, tensor completion, tensor decomposition, rank, low rank, tensor factorization, nuclear norm, singular value decomposition, matrix completion	tensor, padic, tensor decomposition, low rank, tensor completion, hypergraph, rank, nuclear norm, recovery, singular value decomposition
Topic2	regret, policy, bandit, regret bound, game, distributed learning, approachability, reward, multiarmed bandit, bandit problem	regret, bandit, policy, reward, regret bound, markov decision process, multiarmed bandit, bandit problem, thompson sampling, distributed learning	convex, alternating direction method of multiplier, strongly convex, support vector machine, convergence rate, classifier, risk, convexity, stochastic gradient descent, rademacher	policy, agent, reward, markov decision process, reinforcement learning, reward function, deep qnetwork, qlearning, missing mass, policy gradient	policy, agent, reward, reinforcement learning, markov decision process, reward function, deep reinforcement learning, policy gradient, trajectory, value function
Topic3	distortion measure, distortion, self organizing map, tor, voronoi, minima, law of large numbers, asymptotic convergence, string, social learning	dictionary, distortion measure, dictionary learning, sparse coding, overcomplete dictionaries, greedy approximation, sparse representation, hadamard matrix, ksvd, tor	didictionary, distortion measure, dictionary learning, sparse coding, ksvd, overcomplete dictionaries, sparse representation, sparse signal, tor, haar basis	dictionary, distortion measure, dictionary learning, sparse representation, sparse coding, overcomplete dictionaries, photometric stereo, photometric, reconstruction algorithm, reconstruction	convolution neural network, deep network, distortion measure, deep convolutional neural network, accuracy, pruning, gradient descent, deep neural network, stochastic gradient descent, classifier
Topic4	regret, policy, bandit, regret bound, distributed learning, game, reward, multiarmed bandit, approachability, bandit problem	regret, bandit, policy, reward, regret bound, bandit problem, multiarmed bandit, markov decision process, thompson sampling, distributed learning	regret, bandit, policy, multiarmed bandit, reward, regret bound, bandit problem, thompson sampling, markov decision process, semibandit	regret, bandit, multiarmed bandit, regret bound, bandit problem, thompson sampling, best arm identification, multiarmed bandit problem, feedback, contextual bandit	regret, bandit, multiarmed bandit, regret bound, thompson sampling, contextual bandit, bandit problem, online algorithm, multiarmed bandit problem, online learning
Topic5	rumor, centrality, spreads, tree, epidemics, heterogeneity, prosody, stochastic networks, misinformation, ml estimator	copula, rumor, copula density, gaussian copula, prosody, epidemics, misinformation, financial time series, number of trees, model evidence	mean absolute percentage error, meanabsolute error, copula, rumor, vc dimension, universal consistency, weighted mean, neural network learning, covering number, erm	attack, adversarial example, adversarial, adversarial perturbation, adversarial training, adversarial sample, adversarial attack, fgsm, adversarial images, black box attacks	attack, adversarial example, adversarial, adversarial attack, adversarial perturbation, adversarial training, adversarial robustness, adversarial images, defense, adversarial sample

	2007 ~ 2011	2012 ~ 2013	2014 ~ 2015	2016 ~ 2017	2018 ~ 2019
Topic6	ercolation, beta process, sar, noisy image, poisson process, radar, pixels, atr, image analysis, sar image	topic, dirichlet, latent dirichlet allocation, topic model, dirichlet process, atr, posterior, hierarchical dirichlet process, mixture, latent	topic, latent dirichlet allocation, topic model, dirichlet, hierarchical dirichlet process, atr, gibbs sampling, variational inference, dirichlet process, automated detection	fake news, polarization, automated detection, hottopixx, xvector, meanfield variational inference, netgan, twintotwin transfusion syndrome, 3d object detection, conceptors	embedding, long short term memory, recurrent neural network, transformer, emotion, word embedding, recurrent, audio, attention, language model
Topic7	markov equivalence, graph, bidirected graph, undirected graph, mixed graph, social learning, ancestral graph, graphical model, covariance graph, chain graph	graph, bayesian network, bidirected graph, graphical model, markov equivalence, directed acyclic graph, tree, undirected graph, random graph models, erdos renyi graph	bidirected graph, posterior, kernel, gaussian process, probabilistic inference, bayesian inference, variational lower bound, markov equivalence, probabilistic logic, markov chain monte carlo	generative adversarial network, discriminator, generative, variational autoencoder, adversarial neural networks, latent, variational, autoencoder, bidirected graph, unsupervised representation learning	generative adversarial network, discriminator, adversarial neural networks, generative, variational autoencoder, wasserstein, latent space, adversarial learning, generative model, latent
Topic8	lasso, group lasso, irrelevant variables, primal dual, sparsity, variable selection, adaptive lasso, residual variance, glasso, recovery	lasso, irrelevant variables, 1 norm, group lasso, adaptive sampling, minimax lower bound, sparsistency, loss minimization, highdimensional linear model, unknown noise	lasso, screening, group lasso, variable selection, highdimensional linear model, sparse group lasso, high dimensional regression, elastic net, selection consistency, safe screening	lasso, tree, forest, survival, classifier, random forest, screening, ensemble, treatment, highdimensional linear model	anomaly, anomaly detection, detection, time series, outlier, outlier detection, ood, regularized estimation, local outlier factor, unsupervised anomaly detection
Topic9	clustering, cluster, kernel, kmeans clustering, hierarchical clustering, spectral clustering, number of clusters, clustering algorithm, hidden markov model, decision tree	cluster, clustering, kmeans clustering, spectral clustering, number of clusters, graph clustering, clustering algorithm, model based clustering, ccs, clustering method	cluster, clustering, kmeans clustering, graph, spectral clustering, clustering algorithm, hierarchical clustering, number of clusters, block model, community structure	cluster, clustering, kmeans clustering, number of clusters, spectral clustering, clustering algorithm, kmeans algorithm, coreset, mixture model, hierarchical clustering	cluster, clustering, kmeans clustering, clustering algorithm, spectral clustering, number of clusters, dbscan, hierarchical clustering, clustering method, subspace clustering
Topic10	padic, dendrogram, prosody, gps trajectory, xvector, wellbeing, gradient noise, 3d unet, gradient reversal, load forecasting	smml, padic, step functions, prosody, xvector, 3d unet, gradient noise, gradient reversal, faster rcnn, gps trajectory	smml, subspace, matrix completion, nonnegative matrix factorization, low rank, lowrank matrix, leverage score, matrix factorization, principal component analysis, nuclear norm	kernel, graph, rank minimization, approximate nearest neighbors, coding theory, stochastic variance reduced gradient, reconstruction algorithm, active subspace, stochastic gradient, robustness to noise	graph, graph convolution neural network, graph neural network, gegenbauer neural network, approximate nearest neighbors, graph embedding, graph convolution, graph kernel, coding theory, adjacency matrix