An Hotspot Prediction Approach of Scientific Research Based on Autonomously Evolutionary Learner

Yan Ma¹, Lida Zou²*, Yingkun Han¹ and Lei Ma¹

¹Shandong Electric Power Research Institute, Jinan 250002, China
²Shandong University of Finance and Economics, Jinan 250014, China

*Corresponding author: zoulida@sdufe.edu.cn

Abstract. Hotspot prediction of scientific research is a new application in information domain. Big-data analytics technique is a critical way of improving prediction efficiency. Scientific researching data have a high dimension and are from different domains. Just using a single learner is hard to solve the problem. In the paper we adopt several deep neural networks to find the proper data dimension and network structure. The genetic algorithm is introduced as well to fasten the whole prediction. The proposed approach performs well in accuracy ratio and recall ratio compared with the benchmark algorithms. The genetic algorithm and pruning method largely improve its performance as well.

1. Introduction

Scientific and technological information plays an important role in strategy establishment and plan implementation for nation, society and enterprise. Predicting scientific hotspots is a new direction in the informational domain. Scientific researchers and project managers require more forward planning. They make judgments and predictions for potential theories or valuable technologies based on current techniques and social development [1-2].

Big-data analysis technology is a critical way of improving prediction efficiency. With the development of internet and big data, it is easy to get all kinds of data such as document database, sci-tech forum, technology media and we media. Most of the data are textual data, which are used for prediction after converted into structural data [3] through word segmentation or TF-IDF technique. High dimension and large volume are characteristics of scientific data. All the data are from different domains and each data source has massive attributes. For distinct data sources, a single learner is hard to make the accurate prediction for scientific hotspots.

Deep neural network has a high fitting quality. Using several deep neural networks for different domains is a feasible solution to hotspots prediction. However, it is a heavy work to manually select features, choose words and optimize network for each model. A highly automated, large-scale training model is required to improve prediction efficiency and hotspot accuracy.

In the paper, we use autonomous training method to find the proper dimensionality and effective network structure for each model. The multiple-models training way overcomes the curse of dimensionality during hotspots prediction. Moreover, we combine autonomous training and genetic algorithm, which could fasten the process of parameter optimization.
2. Hotspot Prediction framework

In the section, we give prediction framework based on autonomously evolutionary learner. The framework aims to accurately get scientific hotspots in the future. It is especially suitable for the large-scale data sources. Fig. 1 shows its structure. It includes five modules: feature preparation, encoding, training, autonomous evolution, genetics and prediction. The proposed framework uses several deep neural networks to independently predict hot words, and combines autonomous evolution and genetic algorithm to optimize feature selection and network structure.

Feature preparation module mainly prepares data and initially designs the network structure. It collects data from different sources, structures data, cleans data and normalizes data. Since data sources are massive and research domains are different, just using a single model is ineffective and inaccurate. The module initializes multiple deep neural networks for future prediction.

Encoding module gives the encoding description each deep model. It uses codes to describe model structure and source selection, which likes the theory of genes–determine-characters in biology. The code in the paper is called genetic code. Genetic code and model can convert mutually through encoding and decoding operations. Encoding module sends the genetic codes to training module.

Autonomous evolution module is to train and optimize the models. It uses deep neural network to train models. A pruning algorithm based on features and hot words is adopted to optimize models. Effectiveness ratio is introduced to delete models with low accuracy and high cost.

Genetics module generates new models. It uses crossover and mutation operations to generate new models. Good and old models are kept as well. The generated new models are sent back to autonomous evolution module again, until all the hot words are accurately predicted.

Prediction module predicts hot words through the trained models from autonomous evolution module.

Figure 1. Hotspot prediction framework
3. The Concrete Prediction Process
Fig. 2 shows the detailed prediction process of scientific hotspot based on autonomously evolutionary learner. Next we introduce its modules separately.

3.1. Feature Preparation
Feature preparation module consists of feature selection and network structure.

3.1.1. Feature Selection. Feature selection module gets data from data sources. Source system collects textual data by crawlers or API interfaces from sci-tech website, document database and we media. The scientific words are analyzed and labeled according to word frequency, access volume and topic words [4]. Feature selection module generates the trained input data based on the above labeled words. We use $W$ to denote the set of hot words, i.e., $W = \{w_i\}$, where $w_i$ is a scientific word and $i$ is the identifier. Each model just predicts one domain. The set of objective words for model $j$ is denoted as $W_j$. The initial $W_j$ is generated in a random way.

Since data sources of the trained data are wide, the trained data have a high dimension. If selection and deletion operations are not done, there appears the curse of dimensionality. For scientific words in some domains, some attributes are unrelated with their prediction. For example, the hotspot prediction in the water conservancy has a weak connection with we-media data for the sports. It is proper to just select the strongly related data for a domain. However, it is complicated without expert knowledge. In the paper we combine autonomous evolution and genetic algorithm to optimize the work of dimension selection. Feature preparation module initializes the used data attributes for each model through randomly selecting k dimensions. Let $D$ denote the set of data dimensions and $D_j$ is the set of selected dimensions for model $j$, where $j$ is the identifier of the model.
3.1.2. **Network Structure.** The module is used to initialize the network structure of the models. We use \( V_j = \{ v_r \mid r \in [0, R_j - 1], r \text{ is an integer} \} \) to denote a deep neural network, where \( R_j \) is the layers of model \( j \), \( v_r \) is the number of nodes on \( r \)th layer. \( R_j \) and \( v_r \) are randomly generated within a range.

3.2. **Encoding**

Encoding module is responsible to encode each model generated by feature preparation module. The encoding way is as follows. We use \( < D_j, V_j, W_j > \) to denote the code of model \( j \), where \( D_j \) is the selected dimensions for model \( j \), \( V_j \) is its network structure and \( W_j \) is the set of objective words.

3.3. **Autonomous Evolution**

The module consists of training, objective words, dimensions selection and model evaluation. It is used to improve the accuracy rate of model prediction through adjusting dimensions and optimizing models.

3.3.1. **Training.** It uses deep neural network to train data.

3.3.2. **Objective Words.** The module is to manage all the words of model prediction. Since a model corresponds to a research domain, the trained data are required to be limited. Our method is as follows.

1) For a model, the words that are predicted inaccurately are deleted. Let \( err \) denote the prediction error and delete the \( n \) words with the largest \( err \).
2) For the words with small \( err \) value, the new objective words are generated. These new words and their attributes are added into future training. The generation method is as follows.

A) First compute the parameter \( corr(a, b) \).

\[
corr(a, b) = \frac{1}{(C(a) + C(b) - C(a, b))} * \log \frac{C(a) * C(b)}{C^2(a|I)} * \sum_{j \in J} \frac{\log a_j * \log b_j}{\log o_j}
\]  

Here \( corr(a, b) \) denotes the correlation of two words, where a, b are two words, \( C(a) \) is the number of literatures including a, \( C(a, b) \) is the number of literatures including both a and b, \( C(a|I) \) is the total number of literatures, \( J \) is the set of trained literatures, \( a_j \) is the number of word a appeared in literature \( j \), \( b_j \) is the number of word b in literature \( j \) and \( o_j \) is the total number of words in literature \( j \).

B) Compute the coefficients of a word, denoted as \( l_b \).

\[
l_b = \frac{corr(a, b)}{err(a)}, \text{ where } err(a) \text{ is prediction error of word a.}
\]

C) Select the \( n \) words with the largest \( l_b \) and add into \( W_j \).

3.3.3. **Dimensions Selection.** The module is to prune the useless dimensions. It sets the input weight to 0 for each dimension, and observes the change of its average error. If the data attribute \( d \) is deleted and its average error does not change, the attribute is pruned, i.e., \( D_j = D_j - d \).

3.3.4. **Model Evaluation.** The module is to evaluate the training performance of a model. We use average error \( \overline{err} \) and performance ratio as comparative indicators. Parameter \( s \) is performance ratio and computed as \( s = \frac{err}{t} \), where \( t \) is training time. \( t \) is introduced to simplify network and save time. When \( \overline{err} \) is less than \( \eta \), the model is labeled as A model, where \( \eta \) is manually set. A model is optimized and its codes are added into gene pool. Its objective words are deleted from \( W \). When \( \overline{err} \)
of a model is larger than $\eta$ and $s$ lists in the top 80 percent, it is called B model. B models participate into genetic optimization. When $\text{err}$ of a model is larger than $\eta$ and $s$ lists in the bottom 20 percent, it is labeled as C model. C models are directly deleted and quit the optimization.

3.3.5. The Overall Process. Based on the above functions, the overall process of autonomous evolution module are given as follows.

1) For all the input models, execute steps 2) to 7).
2) Do training using deep neural network.
3) Objective words module deletes inaccurately predicted words and increases new objective words.
4) Update the trained data and do training again.
5) Prune the data dimensions using dimensions selection module.
6) Repeatedly execute step 2) to 5), until the average error tends to converge.
7) Evaluate the models and send models to genetics module.
8) The new generated models by genetics module are added into autonomous evolution module and do iterative training, until all the words are predicted with a low error value.

3.4. Genetics

Genetics module consists of crossover and mutation. They aims to improve genetic algorithm and generate better new models. Autonomous evolution cannot obtain the overall information and falls into the local optimum. The genetic algorithm is required to achieve a global optimal solution.

3.4.1. Crossover. Crossover module simulates the genetic evolutionary process based on coding. It includes selection and crossover. Selection is to choose and cross excellent codes. Crossover is to choose the proper parts from the matched coding and generate a new model.

The steps of selection are as follows. Let $W_j$ denote the set of objective words for model $j$. We cluster $W_j$ of each model and obtain $k$ clusters. Codes of each model are input to cross each other.

The crossover operation generates the codes of a new model. Let $<D_a, V_a, W_a>$ denote a new code. $D_a$ is computed as $D_a = \bigcup D_j \mid j \in J_s$, where $J_s$ is a set of codes after clustering. $V_a$ is computed as $V_a = V_j \mid s_j \text{ is optimal}, j \in J_s$, which means $V_a$ has the same structure with the model whose $s_j$ is optimal. $W_a$ is computed as $W_a = \bigcup W_j \mid j \in J_s$. All the newly generated codes are added into library.

3.4.2. Mutation. Mutation module simulates the gene mutation of organism. It first chooses a third of models with the best $s_j$. $<D_j, V_j, W_j>$ denotes the code of model $j$. We randomly add several dimensions in $D_j$. For $V_j$, we change its number of rows by plus or minus one, and randomly change its number of nodes for each row. $W_j$ remains unchanged.

3.5. Prediction

Prediction module aims to predict hotspots based on the trained models and multiple data sources. If the word $w_j$ is the trained ones, i.e., $w_j \in W$, it is predicted by model $j$, where $w_j \in W$. If the word is newly appeared, i.e., $w_j \notin W$, it is predicted using each model. There are too many models. It is time consuming to test each model in turn. We fasten our prediction by introducing correlation between words and models. The correlation between $w_j$ and $W_j$ are computed as $\text{corr}(w_j, W_j) = \sum_{w_j, W_j} \text{corr}(w_j, w_j)$, and the model with the large correlation value is first used.
4. Experiments and Analysis
In the paper we propose a Hotspot Prediction algorithm of Scientific Research based on autonomously evolutionary learner (HPSR). In the algorithm, we use autonomous evolution, genetic algorithm and pruning methods. During the experiments, we test their performance separately. In convenience of description, HPSR-NGA denotes the method without using genetic algorithm, and HPSR-NP denotes the method without using pruning. We choose three performance indicators: recall ratio, accuracy ratio and F1 score [5]. F1 score is a harmonic mean of accuracy ratio and recall ratio, and is computed as $F_1 = \frac{2 \cdot \text{accuracy} \cdot \text{recall}}{\text{accuracy} + \text{recall}}$.

![Figure 3. Recall and accuracy ratio](image1)

![Figure 4. F1 score with time](image2)
Figure 5. F1 score with words

We first choose Recurrent Neural Network (RNN) algorithm [6] as the comparison object. Different numbers of key words are tested. From Fig. 3 we see that the proposed HPSR algorithm has much better accuracy ratio and recall ratio, especially when there are many key words. That’s because RNN just uses one model while HPSR uses several models.

Next we test the performance of using genetic algorithm. We use 5000 key words and evaluate the value of F1 score with the training time. Fig. 4 shows the performance of HPSR and HPSR-NGA. Initially HPSR algorithm performs poorly, since genetic step is time consuming. With training time going by, HPSR algorithm could break away from the local optimum and get a high F1 score. In all, both HPSR and HPSR-NGA have good convergence.

Last the function of pruning method is evaluated. Fig. 5 gives the change of F1 score with the number of key words. It is shown that the introduction of pruning largely improve the performance of hotspot prediction model. For HPSR-NP, the different domains are not well focused, especially for the massive key words.

5. Conclusions
In the paper a hotspot prediction algorithm of scientific research is proposed. In order to avoid the curse of dimensionality, autonomous evolution is introduced to adapt to different domains. We also use genetic algorithm to improve optimization efficiency. Our experiments demonstrate that the proposed HPSR algorithm has an acceptable performance compared with RNN algorithm, especially when key words are massive. The genetic algorithm could also help find the global optimum.

Acknowledgments
This work was financially supported by Science and Technology Plan for Colleges and Universities in Shandong Province (KJ2018BAN046).

References
[1] Research on the Construction of Unified Science and Technology Intelligence Sharing Platform for Scientific Research Institutions, Information Research, 271(2020)53-58.
[2] X.J. Zhuang, M.Y. Wu, The Competitive Intelligence and Its Effect to the Enterprises, Shandong Electric Power, 129(2003)39-41.
[3] M.W. Howard, M.J. Kahana. A Distributed Representation of Temporal Context. J Math
Psychol, 46(2002)269-299.

[4] Y. Ma, W. Qi, D.L. Qi, Y.F. Chen and L.D. Zou, A Prediction Approach of Hot Electric Power Researches for Heterogeneous Time Spans, IOP Conference Series: Material Science and Engineering, 2019, pp. 2575-2579.

[5] A. Li, B. Xue, M. Zhang, Multi-objective feature selection using hybridization of a genetic algorithm and direct multisearch for key quality characteristic selection, Information Sciences, 523(2020) 245-265.

[6] Y. Bengio, A. Courville and P. Vincent, Representation learning: A review and new perspectives. IEEE transactions on pattern analysis and machine intelligence, 35(2013) 1798-1828.