International Journal of Engineering & Technology, 7 (2.3) (2018) 43-45

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research Paper

Technology analysis of artificial intelligence using Bayesian inference for neural networks

Sunghae Jun[1]*

[1]*Department of Statistics, Cheongju University, Korea*
**Corresponding author E-mail: shjun@cju.ac.kr*

Abstract

At present, artificial intelligence (AI) technology is receiving much attention and applied in each field of society. AI is one of the key technologies to lead the fourth industrial revolution along with the internet of things and big data. Therefore, many companies and research institutes are trying to systematically analyze AI technology in order to understand the AI itself correctly. In this paper, we also study on a method to analyze AI technology based on quantitative approach. We correct the patent documents related to AI technology, and analyze them using statistical modelling. We use Bayesian inference for neural networks to build our proposed method. To verify the validity of our research, we carry out a case study using the AI patent documents.

Keywords: *Artificial intelligence; Technology analysis; Bayesian inference; Neural networks; Patent.*

1. Introduction

Artificial intelligence (AI) is to make computers intelligent [1]. The technology of AI has been rapidly dominating other technology areas. Most technologies combine with AI to improve performance in their technology areas. In the era of the fourth industrial revolution, the internet of things (IoT) has changed the world in a smarter way by combining with AI technology [2-5]. So many researchers have tried to analyze the technologies related to AI for knowing the AI itself [6-8]. One such attempt is to analyze objectively AI technology using systematic analysis techniques. In this paper, we propose an objective method to analyze AI technology. First we collect the patent documents related to AI technology. This is because the patent contains much information about the developed technology. Next we analyze the patent data using statistical modelling. We combine Bayesian inference and artificial neural networks to construct the proposed model of AI patent analysis. In addition, we performed a case study using AI patent documents to illustrate how our method could be applied to real problem.

2. Bayesian inference

An Bayesian inference is based on Bayes' rule. This consists of the conditional probability as follow [9-11].

$$P(X=x \mid Y=y)=\frac{P(X=x, Y=y)}{P(Y=y)}=\frac{P(X=x)P(Y=y \mid X=x)}{P(Y=y)} \tag{1}$$

Using this rule, we can compute accurately the unknown probability of event (X|Y) by the known probability of event (Y|X). In addition, the probability of X given Y is updated by the probability of X and the probability of Y given X. This process is called Bayesian learning in Bayesian statistics [12]. In this paper, we use this learning process for neural network model to analyse technological data related to AI. Neal (1996) showed the Bayesian learning process for artificial neural networks based on probability and Bayesian statistics [12]. The Bayesian learning process updates the model parameter β by Bayes' rule as follow [12].

$$P(\beta \mid x) \propto L(\beta \mid x)P(\beta) \tag{2}$$

Where $L(\beta|x)$ and $P(\beta)$ are likelihood function and prior distribution respectively, and we compute the posterior distribution $P(\beta|x)$ by combining $L(\beta|x)$ and $P(\beta)$. So, the posterior probability distribution is updated whenever data (x) is added. In this paper, we apply the Bayesian inference to neural network model for technology analysis of AI.

3. Technology analysis using Bayesian inference for neural networks

In this paper, we use the multilayer perceptron models based on back propagation and feed forward networks. This model basically consists of input x, and output y or $f(x)$. The y is computed by one or more inputs. Also, this model considers one or more hidden layers between input and output layers. The hidden layer is represented as follow [12-14].

$$h(x)=b_1+\sum w_1 x \tag{3}$$

Where b_1 and w_1 are bias and weight of network connection, and the output layer is shown as follow [12-14].

$$f(x)=sigmoid\,(b_2+\sum w_2 h(x)) \tag{4}$$

Where b_2 and w_2 are also bias and weight of neural network model respectively. The *sigmoid()* is an activation function such as logistic or hyperbolic tangent. Therefore, we consider the following model by combining input, hidden, and output layers.

$$y = Logistic(b_2 + \sum w_2 (b_1 + \sum w_1 x)) + \varepsilon_i, \qquad i = 1,2,...,n \quad (5)$$

Where ε_i is followed to Gaussian distribution with mean=0 and variance=σ^2. The n is data size, and *Logistic()* is the activation function as the following growth curve [15].

$$g(x) = \frac{e^x}{1+e^x} \quad (6)$$

Fig. 2 shows the proposed process of technology analysis by Bayesian inference for neural network models.

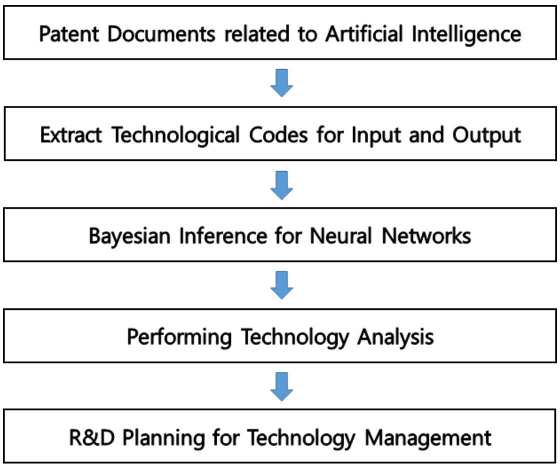

Fig. 1: Technology analysis process by Bayesian inference for neural network models.

We collect the patent documents related to AI from the world patent databases [16,17]. Next we extract the technological codes such as the IPC (international patent classification) code [18]. The codes are used for input and output layers. From the results of technology analysis regarding to AI, we build the research and development (R&D) strategy for management of technology in our research.

4. Experimental results

We collected the patent documents related to AI from the patent databases [16,17]. The number of all searched patent documents is 13,858 after checking valid patents, and total 366 IPC codes are included in the patent documents. The IPC code with the highest frequency was G06F. This represents the technology related to the electronic digital data processing [18]. So we decided the IPC code (G06F) as response (output) variable in our model. Next, from the IPC codes, we selected top 20 IPC codes with the highest frequency except G06F for explanatory (input) variables as follows; G06K, H04N, G10L, G06T, A61B, H04M, G01N, H04R, G06N, G01S, H04L, G06Q, H04B, H04W, G09G, G02B, G11B, G08B, G01B, and B41J. In this paper, we assigned the IPC codes to four groups according to their frequency values. Group 1 has the largest frequency and Group 4 has the smallest frequency. To show the frequency distributions of the input IPC codes, we built the boxplots of them in Fig. 2.

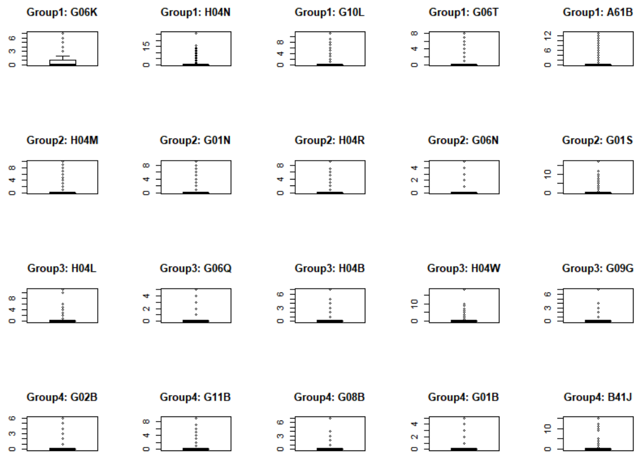

Fig. 2: Boxplots of top 20 IPC codes used for input variables.

The boxplot provides effective visualization of data by using five number summary (minimum, 25th percentile, median, 75th percentile, maximum) [11]. We saw that the frequency of most of the code was very small, including zero. To carry out the proposed method with patent documents preprocessing, we used the R data language and the packages provided by the R [19-22]. To carry out the Bayesian inference for neural networks model, we considered the proposed method for four Group models as follows.

Group1: G06F = *f*(G06K, H04N, G10L, G06T, A61B)

Group2: G06F = *f*(H04M, G01N, H04R, G06N, G01S)

Group3: G06F = *f*(H04L, G06Q, H04B, H04W, G09G)

Group4: G06F = *f*(G02B, G11B, G08B, G01B, B41J)

In the Bayesian neural network model, we considered input layer with five neurons, one hidden layer with two neurons, and output layer with one neuron. In addition, the input and hidden layers have bias neurons. So the number of connection weights is 14, and we compute this weights by Bayesian learning for neural networks. For the performance evaluation of the model, we used a measure to evaluate the difference between the actual and the predicted values. Table 1 shows the mean absolute deviation (MAD) of Bayesian neural network models according to the groups.

Table 1: Mean Absolute deviation (MAD) of Bayesian neural network models according to groups

Group	Min	Median	Mean	Max
1	0.0003	0.3235	0.6075	9.8467
2	0.0077	0.5051	0.7360	9.0655
3	0.0058	0.5609	0.7567	9.4391
4	0.0007	0.5346	0.7586	9.4654

In this paper, the MAD is computed by following measure [23].

$$MAD = \frac{1}{n}\sum_{i=1}^{n}\left|Y_i - \hat{Y}_i\right| \quad (7)$$

Where n is the number of data. Y_i and $\hat{Y}_i$ are actual and predicted values. The smaller the value, the better the performance of the model. We found the median and mean values of MAD of Group 1 is the smallest. So we knew that the IPC codes of G06K, H04N, G10L, G06T, and A61B explain the IPC code of G06F. Fig. 3 shows the plots between actual and predicted values.

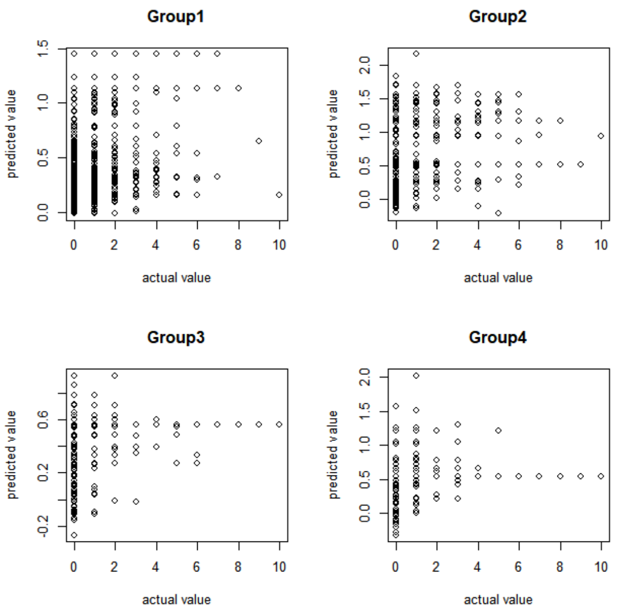

Fig. 3: Visualization of fitted models of four groups.

Most of the actual values are near 0 and the deviation of predictions for these values is the largest. This phenomenon was more pronounced in groups 3 and 4 than in groups 1 and 2. This is because most of the frequency values of the extracted IPC code data are zero. So, the zero-inflated problem should be solved to perform technology analysis using patent data. According to the fitted model of Bayesian inference for neural networks according to Fig. 2, we can construct the technology analysis model for R&D planning in management of technology area. We also confirmed the performance and problem of the proposed method in Table 1 and Fig. 3.

5. Conclusion

We studied on a technology analysis model for AI field using Bayesian inference for neural networks. Bayesian learning, in which the model is updated by the added data, has become an efficient approach to technology analysis because the technique changes over time. In addition, we applied the Bayesian learning to multilayer perceptron, one of popular neural networks. From the result of our proposed method, we can make the R&D strategy for management of technology. So, this research contributes to the intellectual property based R&D planning. But, we also found the zero-inflated problem in our technology analysis based on the IPC codes extracted from patent documents. Therefore, in our future study, we will make more advanced models to overcome the problem in technology analysis.

Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2017R1D1A3B03031152).

References

[1] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach, Third Edition*, Essex, UK, Pearson, 2014.
[2] J. Hossen, S. Sayeed, T. Bhuvaneswari, C. Venkataseshaiah, J. Emerson and C. R. Fatt, “An Automated Fuzzy Logic Based Low Cost Floor Cleaning Mobile Robot”, *Journal of Engineering and Applied Sciences*, 12(1), 119-126, 2017.
[3] N. Sao, “A Review Paper on Mobile Technology for Activity Recognition”, *Journal of Engineering and Applied Sciences*,12(23), 6118-6122, 2017.
[4] P. Dwivedi, S. Sankaranarayanan and V. Choudhary, “IoT Based Smart Garbage Management System”, International Journal of Advanced Trends in Computer Science and Engineering, 6(4), 71-76, 2017.
[5] M. Gooroochurn, D. Kerr and K. Bouazza-Marouf, “A Polynomial Neural Network Classifier based on Gabor Features for the Extraction of Ear Tragus and Eye Corners”, International Journal of Science and Applied Information Technology, 5(4), 14-27, 2016
[6] J. R. Flynn, *What is intelligence?: Beyond the Flynn effect*. Cambridge University Press, 2007.
[7] L. S. Gottfredson, “Mainstream science on intelligence: An editorial with 52 signatories, history, and bibliography”, *Intelligence*, 24(1), 13-23, 1997.
[8] M. I. Jordan and T. M. Mitchell, “Machine learning: Trends, perspectives, and prospects”, *Science*, 349(6245), 255-260, 2015.
[9] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*, Cambridge MA, MIT Press, 2012.
[10] K. Doya, S. Ishii, A. Pouget and R. P. N. Rao, *Bayesian Brain, Probabilistic Approaches to Neural Coding*, Cambridge, MA, MIT Press, 2011.
[11] M. Akritas, *Probability and Statistics with R for Engineers and Scientists*, Boston, Pearson, 2016.
[12] R. M. Neal, *Bayesian learning for neural networks* (Vol. 118), Springer Science & Business Media, 1996.
[13] D. E. Rumelhart, G. E. Hinton and R. J. Williams, “Learning Representations by back-propagation errors”, *Nature*, 323, 533-536, 1986.
[14] D. J. C. Mackay, “A Practical Bayesian Framework for backpropagation networks”, *Neural Computation*, 4, 448-472, 1992.
[15] S. Chatterjee and A. S. Hadi, *Regression Analysis by Example*, 5th edition, Hoboken, NJ, Wiley, 2012.
[16] USPTO, The United States Patent and Trademark Office, http://www.uspto.gov, 2017.
[17] WIPSON, WIPS Corporation, http://www.wipson.com, http://global.wipscorp.com, 2017.
[18] WIPO IPC, International Patent Classification(IPC), World Intellectual Property Organization, http://www.wipo.org, 2017.
[19] H. M. Koduvely, *Learning Bayesian Models with R*, Birmingham, UK, Packt, 2015.
[20] R Development Core Team, R: A language and environment for statistical computing, R Foundation for Statistical Computing, Vienna, Austria, http://www.R-project.org, 2017.
[21] I. Feinerer and K. Hornik, Package ‘tm’ Ver. 0.7-3, Text Mining Package, CRAN of R project, 2017.
[22] P. P. Rodriguez and D. Gianola, Package ‘brnn’, Version 0.6, Bayesian Regularization for Feed-Forward Neural Networks, CRAN of R project, 2017.
[23] B. L. Bowerman, R. T. O’Connell and A. B. Koehler, *Forecasting, Time Series, and Regression*, Independence, KY, Brooks/Cole, 2005.