Creation of Intellectual Property Based on Natural Language Generation Model

1st Antonio Oliveira Nzinga Rene
Department of Information Systems Engineering
Toyama Prefectural University
5180 Kurokawa, Imizu, Toyama 939-0398, Japan
rene@pu-toyama.ac.jp

2nd Takeshi Matsui
Faculty of Social and Information Studies
Gunma University
4-2 Aramaki-machi, Maebashi, Gunma, 371-8510, Japan
tak-matsui@gunma-u.ac.jp

3rd Shigeaki Onoda
Sofbank Corp.
1-9-1 Higashi-shimbashi, Minato-ku, Tokyo, 105-7303, Japan
zoom.taro542@gmail.com

4th Koji Okuhara
Department of Information Systems Engineering
Toyama Prefectural University
5180 Kurokawa, Imizu, Toyama 939-0398, Japan
okuhara@pu-toyama.ac.jp

***Abstract*—Privacy concern in individual and public or private organizational levels is a crucial point. The increase in such a matter is highly evident nowadays, with the development of high technology. This study proposes a system for text mining analyzing characteristics related to language. Quantitative and qualitative evaluations verify the usefulness of the system.**

***Index Terms*—Natural Language Processing, Intellectual Property, Patent Analysis, Text Mining**

I. Introduction

Patent information is a kind of archive of the invention that archives the information of the past invention. It can be useful to society widely, such as management strategy and technical development by utilizing it. Using such open data and platforms to conduct multi-faceted patent analysis is essential in the field of patents and data mining.

According to [1], applications of Information and Technology in the field of patents are roughly classified as follows: classification of patent information processing systems, business promotion systems, management system, and analysis of evaluation systems. The usage of these technologies makes it possible to streamline the work in patent application and analysis.

An actively conducted research within the studies related to patents is the analysis of patents using multi-lingual patent translation and statistics. Patient data is mostly unstructured, confused by tabular forms such as text and citation counts. Studies that take into account such multiple modalities are small amounts compared to the studies mentioned above.

Most of the idea-based support systems classified as management-based systems are idea-based support theory such as KJ law and TRIZ. There are a few mathematical systems based on specific patent data. From this point of view, the need for a new framework to support decision-making is necessary when considering several parameters of patents. Additional to text information, the number of citations, the inventor, and the year of application represent data as well. While considering each of the data, presenting a combination of new patents, etc. will support the decision making of management and development purposes. It is possible to investigate similar existing inventions by visualizing patent information and discovering relevant technical fields simultaneously.

The present study proposes a support system to utilize multimodal data. For the development of the system, we have derived a framework to efficiently collect and store patent data, and create a model that captures the value of patents and the characteristics of keywords based on the data. Finally, we will try to develop an efficient idea support system that includes an interface that users can use using the model. Moreover, if we can map the information and the relationship of each patent to n-dimensional space, we can search for similar patents in search of cosine similarity, vector operations such as addition.

Traditionally, such mapping is done using an autoencoder, commonly used in the field of the image. In the field of natural language processing, the number of dimensions of the input data varies greatly depending on the sentence. The data cannot be applied as it is because it becomes sparse. Therefore, this study proposes a suitable translation model for language data and implements a model that considers patent data other than language. This factor makes it possible to generate a new fictitious system while analyzing the patent within a bird's-eye view, and present keywords to support an idea. The evaluation of the system is performed quantitatively and qualitatively. As a quantitative evaluation, it verifies the usefulness of the system by verifying how much the model can reflect the patent properties and, therefore, verify how more accessible and useful the system is for the people.

II. Model Framework and Formulation

A. Patent Information Processing System

A patent is a type of intellectual property granted to an invention exclusively and exclusively for the protection of an invention, also known as a patent right. There are over 340,000 patents filed in Japan, and inventions in a variety of fields. Also, the composition of the patent consists of natural language text and supplemental information and diagrams. Therefore, using the art of natural language processing for the patent field can be useful to produce industrial value. It leads to academic and industrial results by evaluating and verifying patent information processing technologies in engineering. Patents may mean as intellectual property mentioned above, or they may include administrative processing to apply for themselves.

Information contained in patents consists mainly of specific claims, specifications, and summary. These elements are the target to focus patent information exclusively on engineering applications. Moreover, although there is a patent related to the chart, it is assumed that the chart is removed from the collection object since the present study does not target image processing and image recognition.

Consider, for example, a description of a patent in this study as follows. The patent consists of the title and summary Abstract, the international patent classification (IPC), which shows the patent classification as Classification, Description indicating the text, Claims representing the claim s e.r. and information such as the id and filing date of the patent itself.

The search platform also shows the information on patent citations. Google Patents, for example, has an item called Citation, which lists patents and utility ideas cited by the patent. Similarly, the item Cited by also shown the number of citations. Since the patent information platform does not provide citation information, it is necessary to aggregate it on its own or use a commercial platform. The IPC will be classified into eight sections from A to H in 2020. The IPC is divided into elegant hierarchies such as classes, subclasses, primary groups, and subgroups under sections.

For example, if one abstracts this, the section is ”processing operation and transportation” in addition to IPC, fi, a Japanese patent classification, uspc unique classification in the United States. There is a European-owned classification ECLA, but it is not published at the time of collection, so the study does not use only IPC. Similarly, there is a classification system that separates patents by the purpose of inventions, fields of use, materials, etc. However, this is not taken into account.

B. *Support for Ideas*

Thinking support is a methodology that assists people in thinking so that they can start making decisions or ideas. One of the most famous is the KJ method. However, using the KJ method and other ideas support theory does not use the knowledge date, and each idea itself needs to be thought of by human beings. Therefore, we propose idea processing using electronic data. For example, there is a study that stimulates new ideas by performing cluster analysis on words that users have been playing against, presenting words in clusters that are low-profile with clusters containing the user's idea [2].

In this study, we first performed a hierarchical cluster analysis of text information relevant to the subject that the user wants an idea. We showed that it is possible to present a relationship that is not recognized by the user who is not aware of the subject's knowledge by providing a hint database of words with different fields in this hierarchy, or word co-up, as a hint for the database.

It is easy to use in the regular expression by the word, is excluded as an unnecessary word, including parts of the body such as the head. The word which is not directly related to the idea is removed. On the other hand, there is a mechanism to evaluate the rarity and novelty of the idea, which combines two products that man conceived. There is a research which supported the product creation by ranking and presenting the evaluation for a massive amount of product ideas [3].

Specifically, by entering one product name for the user to seek an idea, we search the database for multiple combinations of products and create phrases for new functions. For example, if a pc has a function called ”draw a picture” and a fragrance ”scent the food,” the phrase which combines the two can be obtained in two different means of ”smell to the picture” and ”Draw food.”

It is a mechanism to generate multiple such files, evaluate unusual flavors from the number of web search engine hits, and present the rarer one as output. Although the existing system is useful in this way, brainstorming, which is still a source of ideas between people, is still essential, and the form of input is limited, and it is not highly versatile. Therefore, the author thought that a more advanced system could be constructed using the language generation, which is a field of patent data and natural language processing that accumulated a large-scale invention.

III. Parsing and constructing text data

A. *Natural Language Processing System*

Natural language processing (NLP) refers to a typical computer processing, and it is a research field related to the natural language we use in our daily life. Among others, the characteristics of language can be summarized as follows:

- The interpretation is arbitrary.
- Not only it can be explained logically, but often it also reflects social conventions.
- The vocabulary and usage changes with time and have different connotations for different areas of expertise.
- The meaning follows a network structure.
- The expression is many-to-many; that is, it has a sense of ambiguity.

Since language is a complex intertwining of various symbols and elements, interpreting it in a machine represents an AI-complete difficult problem to solve. Besides, there are several sub-fields in NLP, which consider different interpretations depending on the country or industry. Although different methods may be employed in each field, generally, the Pre-training of Deep Bidirectional Transformers for Language Understanding (BERT) [4] can perform multiple tasks.

As a technology that is mainly involved in language generation among these fields is Information Access application technology that uses knowledge data transformed into a dialogue system, which is a related field of language generation. After the dialogue system interprets the input, it processes the knowledge data to produce an output across the language model, a particular probability model that expresses the certainty of a sentence or word. These series of frameworks are the same in the related fields such as machine translation and Idea support.

These are rule-based systems using if-then. This system is available on a CUI base. For example, ”Show me some specific examples” for input, including always is a limited but conversational system by providing a set of rules such as ”show me some specific examples”.

However, it is not practical in such a rule-based system, because human beings need to set the appropriate rules for input manually, and the appropriate answers in the conversation will vary greatly depending on the context and context. Language models are used in techniques that use statistical and machine learning to study patterns in vast corpora, such as certainty as a sentence rather than rule-based.

In language generation, there is a limit to selecting the language to generate using the if-then rule. The language model, including the appearance pattern of the language, is included. The language model is the sentence and document generation probability model, and the accuracy ratio for the language expression $w_1 w_2, \dots, w_n$. When considering $P(w_1 w_2, \dots, w_n)$, it is modeled as follows [5].

$$\begin{aligned} P(w_1 w_2, \dots, w_n) = P(w_1, w_2)P(w_3|w_1, w_2)P(w_4|w_2, w_3) \\ \dots P(w_n|w_{n-2}, w_{n-1}) \end{aligned} \tag{1}$$

Research is being conducted on how to obtain the parameter $P(w_i|w_{i-2}, w_{i-1})$ with high accuracy. There are various learning methods for this language model, such as those using Markov chains and those using neural networks.

Given an input sequence $\mathbf{x}$ and an output sequence $\mathbf{y}$, the hidden Markov model (HMM) depends only on the state immediately before each state. Assuming y_i depends only on x_i and y_i depends only on y_{i-1} $\mathbf{x}, \mathbf{y}$ The joint probability of is expressed as follows [5]. Here, (1) is transformed by using the dummy element (x_0, y_0).

$$\begin{aligned} P(\mathbf{x}, \mathbf{y}) &= P(x_k, y_k|x_{k-1}, y_{k-1})P(x_{k-1}, y_{k-1}|x_{k-2}, y_{k-2}) \\ &\quad \dots (x_2, y_2|x_1, y_1)P(x_1, y_1|x_0, y_0) \\ &= \prod_i P(x_i, y_i|x_{i-1}, y_{i-1}) \\ &= \prod_i P(x_i|y_i)P(y_i|y_{i-1}) \end{aligned} \tag{2}$$

Maximum likelihood estimation is performed on this equation to output a probability distribution of likely generated words for each input sequence from a data set that is a pair of an input sequence and an output sequence. Algorithms such as the greedy method and grid search are used for this purpose.

However, using the HMM makes it possible to handle series data, but there is also a problem. The word and the content of the previous sentence also affect it, so it should be considered when modeling it, but HMM considers only the previous state. It has been replaced by the Recurrent Neural Network (RNN), which is more expressive and general in many NLP fields such as computer and dialogue systems.

B. Language Generation

Although RNN has a simple structure, it is challenging to learn long-term dependency. It can cause gradient disappearance and gradient explosion [5]. The long term short memory (LSTM), an extended model of RNN, was devised there. This model has a gate that holds or adjusts memory. It mathematically expresses the gate by performing various operations. The formulation is as follows.

$$f = \sigma(x_t \mathbf{W}_x^f + h_{t-1}\mathbf{W}_h^f + \mathbf{b}^f) \tag{3}$$

$$g = \tanh(x_t \mathbf{W}_x^g + h_{t-1}\mathbf{W}_h^g + \mathbf{b}^g) \tag{4}$$

$$i = \sigma(x_t \mathbf{W}_x^i + h_{t-1}\mathbf{W}_h^i + \mathbf{b}^i) \tag{5}$$

$$o = \sigma(x_t \mathbf{W}_x^o + h_{t-1}\mathbf{W}_h^o + \mathbf{b}^o) \tag{6}$$

$$c_t = f \odot c_{t-1} + g \odot i \tag{7}$$

$$h_t = o \odot \tanh(\mathbf{c}_t) \tag{8}$$

Here, x: input data, h: hidden state, t: time, W: layer weight, b: bias. The advantage is from RNN to memory cell $\mathbf{c}_t$. It can be used with the same interface and it can predict highly accurately for longer sentences. Also, the above affine transformation $(\mathbf{f}, \mathbf{g}, \mathbf{i}, \mathbf{o})$ can be transformed, as well as be calculated collectively and speeded up.

There is a sequence conversion model (sequence to sequence: *seq2seq*) as a typical model for sequence data from linguistically generated sequence data using LSTM. $seq2seq$ is a model devised by NIPS in 2014. Demonstrating the best performance at the time of the announcement in English-French translation using two LSTM networks called decoders [6].

By adding hidden state h to the probabilistic model, (9) expresses model $seq2seq$.

$$\log p(y|x) = \sum_{j=1}^{m} \log p(y_i|y_{<j}, x, h) \tag{9}$$

Where $x_1, ..., x_n$: input series, $y_1, ..., y_n$: output series, $\mathbf{h}$: hidden state, and the characteristics of $seq2seq$ are

- **Advantage**: It is possible to convert the input time series data to another time series data. Convert an input of arbitrary length to a fixed-length vector.
- **Disadvantage**: Since it converts to a fixed-length vector regardless of the length of the input document, there is information loss in the case of a long document.

This is because it is composed of ENCODER that compresses French sentence strings into tensors and DECODER that converts compressed sequences into English sentences. It is possible to capture them in an abstract framework called an encoder/decoder. Even if the encoder/decoder contents are customized in various ways, the sequence conversion will be done correctly if the interface is adhered to.

Model $seq2seq$ is used at a practical level in the translation task where the answer is prepared, but it is not easy to prepare the answer in the intellectual property creation that is the goal of this research. Language generation is performed using a variational autoencoder (VAE), which is an extension of another encoder, the so-called auto encoder (AE).

AE uses a neural network for dimension reduction of data with the algorithm proposed in 2006. Although it is unsupervised learning, it learns by treating not only the input data but also the input data as teacher data. We compress one sample into one latent variable z using a neural network that compresses that information for the input, and restore that z as the input of the decoder network. Learning is performed by correcting the errors so that they are close to each other. Square error often used as the error function.

The better the learning, the more compressed information that captures the data's characteristics can be obtained.However, it is difficult for humans to interpret the value of z. Thus, VAE, denoting z is an extension of AE; with $z \sim N(0, 1)$ [7]. There are two major contributions of the model, namely: it parameterizes the variational lower bound as loss, and speeds up learning in, i.e., it is possible to obtain the probability distribution when considering data in high dimension, and it is possible to obtain new unlearned samples.

Although it was possible to drop into the probability distribution in the generated model even with the conventional model, strong assumptions on the data structure and strong approximation to the model were necessary. The formula for calculating the probability distribution of data in VAE is described below. Usually, it is challenging to calculate $\log p(X)$, so approximation by the lower boundary of variation and the KL divergence of the probability distribution is performed. The calculation reduces the calculation cost.

$$\begin{aligned}\log p_\theta(\mathbf{x}^{(i)}) = D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}^{(i)})|p_\theta(\mathbf{z}|\mathbf{x}^{(i)})) \\ +L(\theta, \phi, \mathbf{x}^{(i)})\end{aligned} \tag{10}$$

Here, the probability distribution $p(X)$ is represented by a logarithm so that it is easy to handle. Note that ϕ and θ are calculated in advance by the maximum likelihood method. Moreover, $z \sim N(\mu(X)$ and $\sigma(X))$ cannot be differentiated in the form of probability distribution if they are left as they are, noise is generated at $\varepsilon \sim N(0, I)$ and convert to $z = \mu(X) + \varepsilon * \sigma(X)$ so that back-propagation can be applied end-to-end.

Encoder-decoder models using RNNs tend to give a high likelihood of frequently-used words. To prevent ratification of appearing words, improve network and loss function, and adjust learning strategy Specifically, by adding the number of patent citations and the number of citations to the loss function. It is possible to increase the probability of generating words that are more valuable and prevent rut and to generate high-value sentences. Therefore, we decided to introduce "Add information other than text to the reward/penalties in the loss function". Equation (11), represents the loss function of the model, if the loss of Sentence-VAE is l_{VAE}.

$$Loss = l_{VAE} + l_p \tag{11}$$

Here, l_{VAE} is as follows, and KL_{weight} is a hyperparameter for adjusting the degree of addition to the loss of KL divergence devised by Sentence-VAE.

$$\begin{aligned}l_{VAE} = KL_{weight} \cdot D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}^{(i)})|p_\theta(\mathbf{z})) \\ +\mathbf{E}_{q(z|X)}[\log p(\mathbf{x}^{(i)}|\mathbf{z})]\end{aligned} \tag{12}$$

However, l_p is a reward term that reflects the parameters of the patent, and it can be expressed as

$$l_p = -(\alpha \cdot citation + \beta \cdot citedby) \tag{13}$$

IV. Proposed Model

A. System Architecture

The Japanese domain Patent-Google is a resource used for quantitative analysis of patents available. The domain is one of the Google search tools, and patent data from all over the world is published in html format. Its main advantage is that organizing and collecting data is more natural than unstructured data such as PDF. On the other hand, Google Patent, another search platform, has its domain with few differences in its interface and the search results. Fig.1 shows the current state of Google's patents search. Google Patent and Patent-In Google, the patent articles are published on the patents.google.com domain. Essentially, there is no significant difference between the two patent documents.

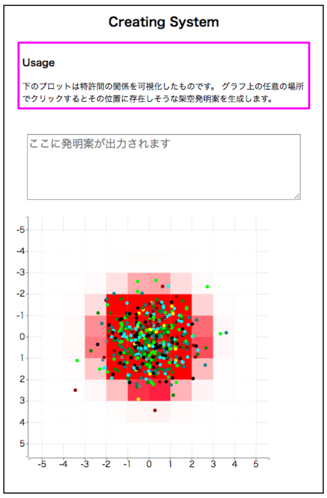
This study uses the Patent-Google, a platform for information collection, which has many search options and can be used as a standard Google search engine. We have collected English patents using the patent classification in the 2008 report published by the World Intellectual Property Organization (WIPO), a specialized agency of the United Nations for patent acquisition.

There are eight categories of data to collect: patent ID, patent title, inventor, approval date, abstract, text, number of cited patents, and number of cited cases. There are seven types of data, where each word is related to each patent. The number of types that can be extracted is different, so we used MongoDB, which is a highly scalable NoSQL, and constructed a separate all-word dictionary to use all the word types included in all the collected patents for analysis.

It is possible to collect in both Japanese and English. Mainly, when collecting in Japanese, morphological analysis is performed, but it is saved in English as it is into the database since words were divided from the beginning.

Following is the description of how we handle the collected data. Firstly, among patent data, the number of citations and citations is the most important index for the value of a patent [1]. Other data include the number of citations and the number of citations of papers. It is virtually impossible to collect on the platform used, and the information on the approval date does not exist in the pending patent, of course. Therefore, we used an index that can be collected by any patent.

When employing an extraction system for input data [8] on essential keywords, the text data for the patent data is divided into three parts: summary, detailed part, and claim.



Creating System

Usage

下のプロットは特許間の関係を可視化したものです。 グラフ上の任意の場所でクリックするとその位置に存在しそうな架空発明案を生成します。

ここに発明案が出力されます

Fig. 1. Example of the system interface

In this study, we do not focus on the accurate prediction for long sentences' improvement, but to a pretreatment using only the summary part. To summarize sentences from data, frequent and less meaningful words in patents such as Figure were removed, and lower cases were considered to treat all the words in a unified way. We used *spacy*, a natural language processing toolkit in Python and NLTK, a tool for speech tagging, plus $termextract$ a software developed jointly by the University of Tokyo and Yokohama National University for extraction of technical terms and comparison models, etc. Hence, $termextract$ extracts technical terms from English sentences using NLTK morphological analyzer.

B. Quantitative Evaluation Experiment

Through this study, we created a web app using the output of $word2vec$ as shown in Fig.4, the first stage of Usage is an explanation column of how to use the system, and the second text box is the part where the output of the system by the proposed method is displayed. In the last part of the graph, each patent mapped in two dimensions by the system is shaped heat map by calculating the multivariate mixed Gaussian distribution from the plot of the data.

TABLE I
CONFIGURATION OF TWO MODELS AND ELBO

Method	Loss function	ELBO
Model A	l_{VAE}	585.6
Model B	$Loss$	633.9

The heat map first determines the ticks, which are the area following the scale, as shown in Fig.2. There are a total of $J = (1, 2, \cdots, j)$, and the color of the heat map is displayed based on the value to determine the probability of actual data. The point in the middle of the region represents the point of Z_j for that region. The task focus on calculating the mixed Gaussian distribution for the point Z_j. Note that Z_j is a two-dimensional data with x and y coordinates, for instance $Z_4 = (3, 1)$. The heatmap was in a two-dimensional plot drawn on the system. It results from employing the graph drawing library d3.js of javascript for graph shaping. It displays 100 of the 30000 existing plots for the sake of increasing the visibility of the map.

$$P(\mathbf{Z_j}) = \sum^{i} P_i(\mathbf{Z_j}) \tag{14}$$

$$P_i(\mathbf{Z_j}) = \frac{1}{(2\pi)\sqrt{|\Sigma_i|}} \times \tag{15}$$

$$\times \exp\left\{-\frac{1}{2}(\mathbf{Z_j} - \mu_\mathbf{i})^\top \Sigma_i^{-1} (\mathbf{Z_j} - \mu_\mathbf{i})\right\}$$

where, $\sum$ is a generalization of variance in the case of dimensions.

Additionally, σ and μ use outputs learned by the encoder of the proposed method. If the standard deviation of the data is too small from the mechanism to calculate the mixed Gaussian distribution, the difference in the color of the heatmap hardly appears because any Gaussian distribution takes a value of almost 0. The comparison of model A's sentence-VAE to model B with improved loss functions is shown in Table 4 at 0.1. Model A and Model B are 256 for hidden layers, word-dropout is 0.5, epochs are 50, and the embedded layer is 300.

Hyperparameters use the same value, but the loss function uses only l_{VAE} in model A, but in model B, it uses l_p with patent citation information added. The value of the peripheral likelihood (evidence lower bound: ELBO) after the learning is shown in Table 4. However, this ELBO is not meaningful to compare values with different loss functions because the scale varies depending on the value of the loss, so it is shown in reference.

Also, the logistic function controls the value of KL divergence used in the loss of both models A and B. As soon as the learning of KL divergence began in the author's environment, it was confirmed that other losses were difficult to fall. Since the percentage of KL divergence loss is less than negative log-likelihood, learning negative log-likelihood tends to decrease the overall loss.

$$\text{Logistic}(x) = \frac{step}{1 + e^{-k(x-x_0)}} \tag{16}$$

Here, $step$ represents the step of learning. x_0, k is a hyper parameter and k is the greater the steepness, The larger the x_0, KL$_{loss}$, datasize/batchsize, the smaller the sections.

$$KL_{\text{loss}} = \text{Logistic}(x) * D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}^{(i)})|p_\theta(\mathbf{z})) \tag{17}$$

When the point in the interface of the proposed system is pressed, the output example is introduced. It should be noted that this output example does not perform any post-processing, and displays what each model outputs as it is.

Including other examples, technical terms for both model A and model B: "proanthocyanidin," "chromium-based," "proanthocyanidin" can be seen to be generated.

Output of model A

marbleized implantable hearing aid . the method includes : a first lens group having a plurality of pixels , and a second lens group , and a second lens group having a first lens group having a positive refractive power supply unit (1) . <eos>

Output of model B

the invention relates to a method for producing a <unk> of a first and second conjugates , and a second propylene-based group , and a second layer of the first and second second lens group , a second lens group , a second lens group , a second lens group , a second lens group , a second lens group , a second lens group , a second lens group , a second lens group , $\cdots$, a second lens group”

The area on the left side of Model A (x is negative) is the symbol ”a” as shown in a method for producing a pronthocyanidin extract, a method for producing a compound represented by formula ($\sim$ i $\sim$)”) $\sim$, a$\sim$) $\sim$) $\sim$) and a$\sim$). There were many examples of generating. Besides, when generated in an area away from the heat map, the output of only the eos symbol representing the end of the sentence was found. This suggests that there is no patent document in the value of latent variables that deviate significantly from the learned range. It will be data that assists the region estimation hypothesis in the mixed Gaussian distribution described above. On the other hand, Model B did not generate documents with symbols. It is thought that words with a higher number of quotes than symbols that appear more frequently for containing the number of patent citations are selected.

V. Concluding Remarks

By collecting the patent specification data and learning using VAE, we visualized complex patent data on two-dimensional data with continuity according to the Gaussian distribution. By embedding patent data in a higher-dimensional latent variable, we can also generate a more accurate invention support proposal. The development of the invention support system, which is the original purpose was achieved. As a future direction for the research, we would like to improve the accuracy by adapting methods to prevent overlearning, such as attention mechanism and copy mechanism, to the encoder or decoder part.

References

[1] A. Fujii, H. Tangawa, M. Iwayama, H. Namba, M. Yamamoto, and M. Ushiyama, ”Patent Information Processing: a natural language processing approach”, Corona Publishing Co., Tokyo, 2012. (in Japanese)

[2] J. Itou, T. Higashi and J. Munemori, ”Proposal and Application of Idea Generation Support System Providing Words of Low Co-occurrence Degree”, Information Processing Society of Japan, vol. 56, No. 6, pp. 1528-1540, 2015. (in Japanese)

[3] Y. Nishihara, J. Hibino, J. Fukumoto and R. Yamanishi, An Idea Generation Support System Evaluating Function’s Novelty in Product Combination, vol.27, No.4, pp. 669-679, 2015. (in Japanese)

[4] J. Devlin, M. Chang, K. Lee, K. Toutanova ”BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”, *In North American Association for Computational Linguistics (NAACL)*, arXiv preprint arXiv:1810.04805, 2019.

[5] H. Takamura, Introduction to Machine Learning for Natural Language Processing, Corona Publishing Co., Tokyo, 2010. (in Japanese)

[6] I. Sutskever, O. Vinyals, Q. V. Le, ”Sequence to Sequence Learning with Neural Networks”, *In Advances in Neural Information Processing Systems (NIPS 2014)*, 2014.

[7] D. P. Kingma, M. Welling, ”Auto-Encoding Variational Bayes”, arXiv:1312.6114, 2014.

[8] Automatic extraction of technical terms (keyword) python module termextract, The University of Tokyo, February 24, 2018. Accessed on: October 25, 2019. [Online]. Available: http://gensen.dl.itc.u-tokyo.ac.jp/pytermextract/

[9] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R, Jozefowicz, S. Bengio, ”Generating Sentences from a Continuous Space”, *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, K16-1002, 2016.