

Research Article

Open Relation Extraction in Patent Claims with a Hybrid Network

Boting Geng 

School of Computer Science and Computing, Zhejiang University, 310063, China

Correspondence should be addressed to Boting Geng; 11321030@zju.edu.cn

Received 8 January 2021; Revised 28 February 2021; Accepted 10 April 2021; Published 28 April 2021

Academic Editor: Wenqing Wu

Copyright © 2021 Boting Geng. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Research on relation extraction from patent documents, a high-priority topic of natural language process in recent years, is of great significance to a series of patent downstream applications, such as patent content mining, patent retrieval, and patent knowledge base constructions. Due to lengthy sentences, crossdomain technical terms, and complex structure of patent claims, it is extremely difficult to extract open triples with traditional methods of Natural Language Processing (NLP) parsers. In this paper, we propose an Open Relation Extraction (ORE) approach with transforming relation extraction problem into sequence labeling problem in patent claims, which extract none predefined relationship triples from patent claims with a hybrid neural network architecture based on multihead attention mechanism. The hybrid neural network framework combined with Bi-LSTM and CNN is proposed to extract argument phrase features and relation phrase features simultaneously. The Bi-LSTM network gains long distance dependency features, and the CNN obtains local content feature; then, multihead attention mechanism is applied to get potential dependency relationship for time series of RNN model; the result of neural network proposed above applied to our constructed open patent relation dataset shows that our method outperforms both traditional classification algorithms of machine learning and the state-of-art neural network classification models in the measures of Precision, Recall, and F1.

1. Introduction

With the development of economy, patent documents, being an extremely important knowledge carrier, record a large number of valuable inventions, creative ideas, and excellent design concepts. Automatically extracting none predefined relation triples from patent claims, which contains a series of rights granted by a government for a given limited period, is a vital basic research application for some upper level applications of patent document analysis, such as patent information retrieval [1, 2], patent classification [3], patent categorization [4], and patent knowledge graph construction [5].

However, relation extraction from patent document is not an easy task. On one hand, specification requirements for patent writing leads to lengthy and complex sentence, which results in its difficulties to parse with normal NLP tools; on the other hand, traditional approaches, NLP-based linguistic method, statistics-based machine learning method, and multimethod hybrid method [6] cannot catch temporal information and long sentence-level global dependency features.

In this paper, we propose an open relation extraction model of hybrid neural network to extract relation triples from patent claims, where Bi-LSTM network can obtain temporal information from the whole sentence, and CNN pooling can gain local content information; at the same time, multihead attention is incorporated into extracting content dependency feature in order to better serve for sequence label classification problems. Our main contributions are summarized as follows:

- (1) A hybrid neural network (Bi-LSTM+CNN+CRF) of open relation extraction (ORE) model is firstly proposed to extract none predefined triples from patent document
- (2) Multihead attention technique serves for better sequence label dependency classification
- (3) We constructed an open patent relation corpus in favor of adopting supervised approaches to ORE task in patent analysis, including 1309 annotated claims with about 29850 sentences

- (4) We systematically compare the performance of a series of exiting neural model in the context of the ORE in patent claims. Meanwhile, a variety of experiments help readers to better understand reliability of our hybrid model

2. Related Work

As for the traditional semantic relation extraction from patent documents, there are mainly four methods, which are NLP-based linguistic method, statistics-based machine learning method, and multimethod hybrid method.

On one hand, in the early period of semantic relation extraction from patent documents, NLP-based linguistic methods are dominant. Most of the existing methods made use of linguistic analysis. Regular expression pattern matching techniques is proposed to parse, annotate, and extract target semantic information for knowledge sharing in machine readable format OWL [7]; extracting hyponymy lexical relations is conducted on patent documents using lexis-syntactic patterns [8] and extracting knowledge combined with domain ontology from patent unstructured data [9]. Data-intensive methods are incorporating into patent claim analysis for enhancing analysis robustness combined with symbolic grammar formalisms [10]. Conceptual graphs are extracted from patent claims for comparing patent similarity analysis or any domain of interest [5]. A patent processing system named PATExpert is designed for summarizing patent claims, where deep strategies of syntactic dependency relationship analysis operate on deep-syntactic structures of the claims for improving its readability [11]. Gabriela et al. [12] proposed an extraction of verbal content relations from patent claims using deep syntactic structures. Fantoni et al. [13] proposed a method of automatically detecting and extracting information about functions, the physical behaviours, and states of the system from patent text with a large knowledge base and a series of NLP tools. Lee et al. [14] proposed a hierarchical keyword vector for representing the dependency relationships among claim elements and a tree matching algorithm for comparing claim elements of parents to assess patent infringement risks. Taeyeoun Roh et al. [15] proposed a series of rules to structure and layer technological information in patent claims through NLP tools.

On the other hand, statistical-based machine learning is frequently applied for processing patent analysis in recent years. Gabriela [16] proposed a two-stage method of rule-based claim paragraph segmentation and machine learning-based of conditional random field (CRF) lengthy sentence segmentation which will help automatically detect division phrases for forming meaningful shorter sentences. Wang et al. [17] present an approach to extracting principle knowledge from process patents classifying with contraction matrix. Okamoto et al. [18] proposed an information-based technique to grasp the patent claim structure through entity mention extraction and the relation extraction method with DeepDive [19] platform which using Markov logic network-based inference [20] and distant supervision-based labeling [21] to extract relations from unstructured text. Deng et al. [22] proposed to construct knowledge graph for

facilitating technology transfer where common knowledge base can reveal the technical details of technical documents and assist with the identification of suitable technologies.

Besides, with the rise of deep learning technology, especially its wide application in natural language processing, hybrid technologies as above have emerged for patent mining, such as patent information extraction, patent relations extraction, and construction of patent semantic knowledge base. Yang and Soo [5] proposed a method to convert a patent claim into a formally defined conceptual graph with hybrid techniques of part-of-speech tags, conceptual graphs, domain ontology, and dependency tree. Korobkin et al. [23] proposed a hybrid methodology of LDA-based statistical and semantic text analysis to extract a physical knowledge in the form of physical effects and their practical applications. Carvalho et al. [24] present a hybrid method of extracting semantic information from patent claims by using semantic annotations phrasal structures, abstracting domain ontology information, and outputting ontology-friendly structures to achieve generalization. Lv et al. [25] proposed a hybrid method of patent terminology relation extraction combined with attention mechanism and Bi-LSTM [26] model to construct the patent knowledge graph.

Different from traditional relation extraction, where categories of relationships are classified at advance, open relation extraction (ORE) extract none predefined triples from unstructured text. ORE is firstly defined by Banko et al. [27] who proposed to extract none predefined relations from web, attracting extensive attention and follow-up researches in various fields. Del Corro and Gemulla [28] then proposed dependency parsing-based clause IE framework to detect and extract “useful” pieces of information clauses. Neural network are also incorporated into ORE [29, 30] with end-to-end sequence model or encoder-decoder model.

Our work is similar with Lv et al. [25] and [29–31], but Bi-LSTM and attention mechanism, together with open relation extraction, are firstly proposed to extract the none predefined relationship from the patent documents forming Subject-Relation-Object triples. As we believe that NLP-based parsing tools cannot catch long dependency relationship of lengthy patent sentences, different phased attention would improve the end-to-end sequence labeling classifications. We propose a hybrid neural network framework of extracting open relations from patent claims with multi-head attention. Although Bidirectional Encoder Representation from Transformers (BERT) [32–35], another neural network model based on bidirectional transformer, performs excellent in a series of natural language processing tasks including sequence tagging, we would leave it for the future work.

3. Our Hybrid ORE Neural Framework

The paper proposes a supervised neural network of extracting open relations from patent claims without predefined relation categories, which enables a supervised machine learning approach to ORE in patent claims. We define the task as a sequence tagging problem, and we develop an

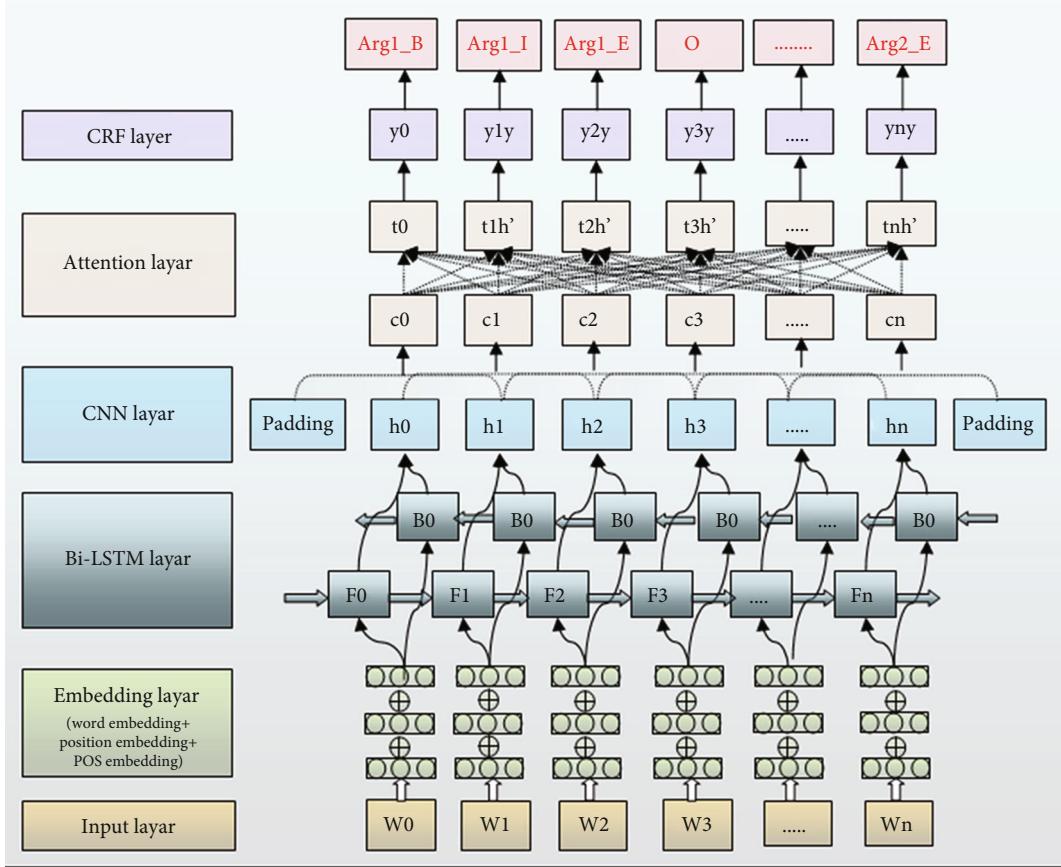


FIGURE 1: Our hybrid network architecture.

end-to-end neural mode with Bi-LSTM and CNN with multihead attention to classify labels above. At first stage, as for the lengthy and complex structure, a machine learning-based method is used to detect segmentation word or phrases for splitting meaningful pieces of short sentences. And then word features and part-of-speech features are incorporated into the Bi-LSTM network. At then, multihead attention mechanism is applied to Bi-LSTM features for help dependency relationship label classification. Postprocessing operation is done for getting Argument1-relation-Argument2-like triples. Our neural ORE architecture is shown in Figure 1.

3.1. Task Formulation. In this paper, we define our neural ORE model as extracting triples from patent claims, where a triple often consists of a predicate and two arguments with contiguous spans from the sentence. As we show in the follow table, the formulation is defined with a more expressive BIEOS tagging scheme as shown the dashed lines, which can better capture dependency relationships from content than BIO tagging scheme. The relation phrase labels are encoded as Verb, Prep, or Noun labels type, while arguments are represented as Arg labels, where Arg1 stands for the first argument and Arg2 acts as the second argument. Several examples are shown in Table 1.

3.2. Feature Embedding. Word embedding is an operation of transforming a word token into a real-valued vector to represent syntactic and semantic information from content. Given a sentence consisting of n words $S = \{x_1, x_2, x_3, \dots, x_n\}$, every word x_i is converted into a real-valued vector e_i by looking up the embedding matrix $W^{\text{word}} \in \mathbb{R}^{d^w \times |V|}$, where V stands for the whole vocabulary and d^w represents as the size of word embedding. We use Glove [26] as our word embedding model. Part of speech embedding is transforming POS of each word x_i in sentence S into a one-hot vector p_i , which comes from annotated brown corpus with 36 types. Finally, the concatenation of word embedding e_i and POS embedding p_i is input feature of our neural model.

3.3. Bi-LSTM Network. As deep learning technology and natural language processing combine more and more closely, long short-term memory (LSTM) network, which is firstly proposed by Hochreiter and Schmidhuber in 1997 to solve gradient vanishing problem, shows its good merit on capturing long distance relationship in different NLP subtasks. The transfer diagram of adjacent units in LSTM neural network is shown in Figure 2.

The core design philosophy of LSTM is an adaptive gating mechanism, which decides the degree to which LSTM units keep the previous state and memorize the

TABLE 1: Our neural open relation extraction model.

Neural open relation extraction model
(a) A mono-block engine having a cylinder head structure according to claim 2. (A mono-block engine; having; a cylinder head structure) AB_Arg1 mono-blockI_Arg1 engineE_Arg1 havingS_Verb aB_Arg2 cylinderI_Arg2 headI_Arg2 structureE_Arg2 accordingO toO claimO 2O .O
(b) A ported housing having at least one housing port, said housing having an inside diameter and an outside diameter (a ported housing; having; at least one housing port) (said housing; having; an inside diameter and an outside diameter) AB_Arg1 portedI_Arg1 housingE_Arg1 havingS_Verb atB_Arg2 leastI_Arg2 oneI_Arg2 housingI_Arg2 portE_Arg2,O saidB_Arg1 housingE_Arg1 havingS_Verb anB_Arg2 insideI_Arg2 diameterI_Arg2 and I_Arg2 an I_Arg2 outsideI_Arg2 diameterE_Arg2 ;O
(c) Wherein a flexible press plate, anchored in the counterpart at one of the edges running in the longitudinal direction of the counterpart, is arranged on the surface of the counterpart located in the press zone (a flexible press plate; anchored in; the counterpart) (a flexible press plate; running; in the longitudinal direction of the counterpart) (a flexible press plate; is arranged on; the surface of the counterpart) whereinO aB_Arg1 flexibleI_Arg1 pressI_Arg1 plateE_Arg1,O anchoredB_Verb inE_Verb theB_Arg2 counterpartI_Arg2 atO oneO ofO theO edgesO runningS_Verb inB_Arg2 theI_Arg2 longitudinal I_Arg2 directionI_Arg2 ofI_Arg2 theI_Arg2 counterpartE_Arg2, isB_Verb arrangedI_Verb onE_Verb theB_Arg2 surfaceI_Arg2 ofI_Arg2 theI_Arg2 counterpartE_Arg2 locatedO inO theO pressO zone O .O

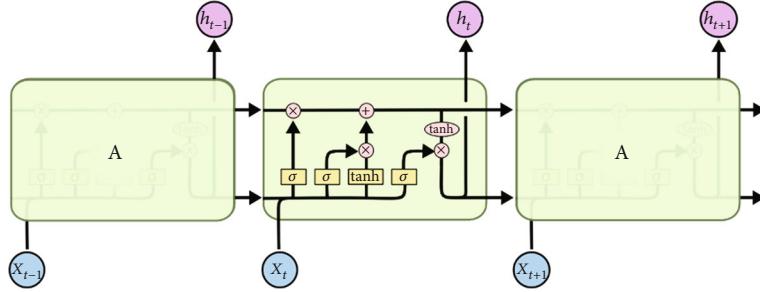


FIGURE 2: Basic neural LSTM unit.

extracted features of current data input [36]. The calculation process is as follows:

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f), \quad (1)$$

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i), \quad (2)$$

$$\tilde{C}_t = \tanh(W_c * [h_{t-1}, x_t] + b_c), \quad (3)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t, \quad (4)$$

$$O_t = \sigma(W_o * [h_{t-1}, x_t] + b_o), \quad (5)$$

$$h_t = O_t * \tanh(C_t). \quad (6)$$

A typical LSTM network consists of four parts: one forget gate f_t , one input gate i_t , one current cell state C_t , and one output gate O_t . Through four parts of the iterative calculation above, cell units decide whether to take the inputs, forget the memory stored before, and output the state generated later. Bidirectional LSTM network is the combination of forward LSTM networks and backward LSTM networks, where the hidden layer of the latter network flows in opposite position as that of the former, which can cap-

ture the future information as well as the past one. Thus, the Bi-LSTM model is able to exploit information both from the past and the future, more suitable for the sequence tagging model tasks. In this paper, we use the Bi-LSTM model to obtain the semantic and syntactic information from the sentence, and we get the combined hidden information h_t with element-wise sum operation as the following equation from two subnetworks of the forward hidden state \vec{h}_t and backward hidden state \overleftarrow{h}_t .

$$h_t = [\vec{h}_t \oplus \overleftarrow{h}_t]. \quad (7)$$

3.4. Multihead Attention. Attention mechanism has now become a predominant concept in neural network literature in recent years and has received varying degrees of attention and research within the artificial intelligence (AI) community in a large number of applications, such as speech recognition, computer vision, natural language processing, and statistical learning. In this paper, we adopt the multi-head attention, which has shown excellent performance in many tasks, such as reading comprehension [36] (Cheng et al., 2016), text inheritance [37] (textual ailment/Parikh

et al., 2016), and automatic text summary [38] (Paulus et al., 2017). The essence of multihead attention is to do multiple calculations of self-attention, which can enable sequence-to-sequence neural model to obtain more features from different representation subspaces, so that the model can capture more context information of sentences. The relevant attention equations are described as below:

$$\begin{aligned} \text{Attention}(Q, K, V) &= \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) * V, \\ \text{head}_i &= \text{Attention}\left(QW_i^Q, KW_i^K, VW_i^V\right), \\ \text{MultiHead}(Q, K, V) &= \text{concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_h), \end{aligned} \quad (8)$$

where Q , K , and V represent query matrix, key matrix, and value matrix of the multihead attention mechanism, and in the above equations, $Q = n * d_k$, $K = n * d_k$, $V = n * d_k$, and $i \in h$. For each head attention, we compute the attention weight by Equation (2), and finally, we concatenate each head as output results of attention layer.

3.5. CNN Network. Convolutional neural networks (CNN) is a good means of capturing salient local features from whole sequence as for its capability of learning local semantic patterns by its flexible convolutional structure in multidimensional feature extraction [39]. Convolution is often thought of as the product of a weight vector and a sequence vector. The weights matrix is regarded as the filter for the convolution [40]. Given various convolution window length, different outputs are fed to a max-pooling layer, where we can get a feature vector of fixed length.

3.6. CRF Layer. The output of the softmax layer does not affect each other and is independent of each other, while Bi-LSTM can learn semantic and syntactic information about the content. But as for some tasks, such as Noun chunking and Named Entity Recognition (NER), output labels are mutually restrictive. Taking “aB_Arg1 flexibleI_Arg1 pressI_Arg1 plateE_Arg1” for an example, label B_Arg1 must be in front of I_Arg1, and label E_Arg1 must come after label B_Arg1 and I_Arg1, and other sequence is illegal. And the result label calculation of the CRF layer is realized by dynamic programming optimization, which would obviously outperform the model without the CRF layer in the time series estimation problem.

4. Experiments

4.1. Dataset. We extract 1309 claims from patent documents from USPTO and annotate the claims with thirty undergraduates for about 2 months. The constructed dataset finally contains 29850 sentences, where 60% for training, 20% for verification, and 20% for test. For argument1 and argument2, we use BIEOS label mechanism, which is also suitable for relation phrase labels. There are three relationships in the whole labeled dataset, and each relationship contains a single tag “S” or two more tags “BE” or “BIE.”

TABLE 2: Annotation statistics of our constructed dataset.

	S	B	I	E
Verb	16994	13707	10921	13689
Prep	2624	902	752	874
Adj	409	1712	1186	1693

TABLE 3: Hyperparameter setting of the framework.

Layer	Hyperparameter	
Embedding	Word embedding size	300
	POS embedding size	26
	Entity embedding size	12
CNN	Kernel size	3
	Number of filters	100
Bi-LSTM	State size	300
Dropout	Dropout rate	0.5
	Batch size	50
	Initial learning rate	0.01
	Sequence length	50

TABLE 4: Results of ORE dataset experiments on models.

	P (%)	R (%)	F1-score (%)
Bi-LSTM ¹	89.1542%	90.2017%	89.6749%
Bi-LSTM+CNN+CRF ²	93.9204%	93.1501%	93.5337%
Bi-LSTM+CNN+CRF (label embedding) ³	97.0457%	96.1206%	96.5809%
Bi-LSTM+CNN+CRF (label embedding+attention) ⁴	97.3074%	97.7388%	97.5226%

The statistics of all labels are shown in Table 1. Finally, we evaluate our patent ORE mode on above dataset. The results are measured by Precision (P), Recall (R), and F1-score, which is defined in Table 2.

$$\begin{aligned} P &= \frac{\text{TP}}{\text{TP} + \text{FP}}, \\ R &= \frac{\text{TP}}{\text{TP} + \text{FN}}, \\ \text{F1}_{\text{score}} &= \frac{2 * P * R}{P + R}. \end{aligned} \quad (9)$$

4.2. Hyperparameters Setting. We implement our model with python 3.5 in Keras on NVIDIA Quadro P2000. Adam method is used to optimize our model, learning rate is set to 0.01, and batch size is 50. For multihead attention, we set the number of attention heads is 4, and we use Glove as word embedding model, and the dimension of word vectors is set as 300. Part of speech embedding size is one-hot vector and is set as 26, and relation label embedding size is also set as 12. The dropout rate is set to 0.1 to prevent overfitting, and L2 regularization is also employed in training to

prevent overfitting. The max length of the sentence is set as 100. The detailed parameters of the framework are shown in the following Table 3.

4.3. Experiments and Discussion. In our model, label embedding of our hybrid neural network model consists of word embedding, part-of-speech embedding, and relation tagging embedding. More information feature would be incorporated into the embedding layer through the concatenation by the last dimension for each word. The attention mechanism used in our model is multihead attention, which layer is followed by CNN layer. From a series of experiments in Table 4 above, we obviously conclude that hybrid neural network model performs better than traditional neural network model like Bi-LSTM, such as model 1 and model 2 in Table 4, and neural network models with the help of label embedding obviously perform better than the models without the label embedding, such as model 3 and the model 1, in the evaluation measures of Precision, Recall, and F1 score. Through the comparison with the other neural models, our model with multihead attention outperforms other model as well.

4.4. Conclusion and Future Work. In this paper, we propose a Patent Open Relation Extraction neural model. Instead of employing feature engineering, we use a hybrid Bi-LSTM +CNN+CRF neural model with multihead attention mechanism. The hybrid model outperforms the single other model obviously on our self-constructed patent sequence tagging dataset. In the future, we consider incorporating the transform model into our model, such as Bidirectional Encoder Representation from Transformers (BERT), and we also consider patent domain word embedding, which we think would potentially improve the performance.

Data Availability

The dataset used to support the findings of this study have not been made available as the dataset also forms part of an ongoing study.

Conflicts of Interest

The author declares that they have no conflicts of interest.

References

- [1] L. Chen, N. Tokuda, and H. Adachi, “A patent document retrieval system addressing both semantic and syntactic properties,” in *Proceedings of ACL Workshop on Patent Corpus Processing*, pp. 1–6, Sapporo Convention Center, Sapporo, Japan, 2003.
- [2] M. Iwayama, A. Fujii, N. Kando, and Y. Marukawa, “Evaluating patent retrieval in the third NTCIR workshop,” *Information Processing & Management*, vol. 42, no. 1, pp. 207–221, 2006.
- [3] C. J. Fall, A. Törcsvári, K. Benzineb, and G. Karetka, “Automatic categorization in the international patent classification,” *ACM SIGIR Forum*, vol. 37, no. 1, pp. 10–25, 2003.
- [4] J. H. Kim and K. S. Choi, “Patent document categorization based on semantic structural information,” *Information Processing & Management*, vol. 43, no. 5, pp. 1200–1215, 2007.
- [5] S.-Y. Yang and V.-W. Soo, “Extract conceptual graphs from plain texts in patent claims,” *Engineering Applications of Artificial Intelligence*, vol. 25, no. 4, pp. 874–887, 2012.
- [6] X. Lv, X. Lv, X. You, Z. Dong, and J. Han, “Relation extraction toward patent domain based on keyword strategy and attention+BiLSTM model,” in *Collaborative Computing: Networking, Applications and Worksharing. CollaborateCom 2019*, vol. 292 of Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, , pp. 408–416, Springer, 2019.
- [7] S.-Y. L. Shi-Yao, S.-N. Lin, C.-F. Lee, S.-L. Cheng, and V.-W. Soo, “Automatic extraction of semantic relations from patent claims,” *International Journal of Electronic Business Management*, vol. 6, no. 1, pp. 45–54, 2008.
- [8] L. Andersson, M. L. J. Pallotti, F. Piroi, A. Hanbury, and A. Rauber, *Insight to hyponymy lexical relation extraction in the patent genre versus other text genres Proceedings of the First International Workshop on Patent Mining and Its Applications (IPAMIN)*, Hildesheim, Germany, 2014.
- [9] A. Souili, D. Cavallucci, and F. Rousselot, “Natural language processing (NLP) - a solution for knowledge extraction from patent unstructured data,” *Procedia Engineering*, vol. 131, pp. 635–643, 2015.
- [10] S. Sheremeteva, “Natural language analysis of patent claims,” in *Proceedings of the ACL-2003 workshop on Patent corpus processing*, pp. 66–73, Sapporo Convention Center, Sapporo, Japan, 2003.
- [11] N. Bouayad-Agha, G. Casamayor, G. Ferraro, S. Mille, V. Vidal, and L. Wanner, *Improving the Comprehension of Legal Documentation: The Case of Patent Claims*, ICAIL, Barcelona, Spain., 2009.
- [12] T. Roh, Y. Jeong, and B. Yoon, “Developing a methodology of structuring and layering technological information in patent documents through natural language processing,” *Sustainability*, vol. 9, no. 11, p. 2117, 2017.
- [13] G. Fantoni, R. Apreda, F. Dell’Orletta, and M. Monge, “Automatic extraction of function-behaviour-state information from patents,” *Advanced Engineering Informatics*, vol. 27, no. 3, pp. 317–334, 2013.
- [14] C. Lee, B. Song, and Y. Park, “How to assess patent infringement risks: a semantic patent claim analysis using dependency relationships,” *Technology Analysis & Strategic Management*, vol. 25, no. 1, pp. 23–38, 2013.
- [15] G. Ferraro and L. Wanner, “Labeling semantically motivated clusters of verbal relations,” *Procesamiento del Lenguaje Natural*, vol. 49, pp. 129–138, 2012.
- [16] G. Ferraro, H. Suominen, and J. Nualart, “Segmentation of patent claims for improving their readability,” in *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR)*, pp. 66–73, Gothenburg, Sweden, 2014.
- [17] G. Wang, X. Tian, J. Geng, R. Evans, and S. Che, “Extraction of principle knowledge from process patents for manufacturing process innovation,” *Procedia CIRP*, vol. 56, pp. 193–198, 2016.
- [18] M. Okamoto, Z. Shan, and R. Orihara, “Applying information extraction for patent structure analysis,” in *Proceedings of the 40th International ACM SIGIR Conference on Research and*

- Development in Information Retrieval*, pp. 989–992, Tokyo, Japan, 2017.
- [19] J. Shin, F. W. SenWu, C. De Sa, C. Zhang, and C. Ré, “Incremental knowledge base construction using DeepDive,” *Proceedings of the VLDB Endowment*, vol. 8, no. 11, 2015.
 - [20] P. Domingos and D. Lowd, “Markov logic: an interface layer for artificial intelligence,” *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 3, no. 1, 2009.
 - [21] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, “Distant supervision for relation extraction without labeled data,” in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - ACL-IJCNLP '09*, pp. 1003–1011, Singapore, 2009.
 - [22] W. Deng, X. Huang, and P. Zhu, “Facilitating technology transfer by patent knowledge graph,” in *Proceedings of the 52nd Hawaii International Conference on System Sciences*, Grand Wailea, United States, 2019.
 - [23] D. M. Korobkin, S. A. Fomenkov, and A. G. Kravets, “Extraction of physical effects practical applications from patent database,” in *2017 8th International Conference on Information, Intelligence, Systems & Applications (IISA)*, pp. 1–5, Larnaca, Cyprus, 2017.
 - [24] D. S. Carvalho, F. M. G. Franca, and P. M. V. Lima, “Extracting semantic information from patent claims using phrasal structure annotations,” in *2014 Brazilian Conference on Intelligent Systems*, pp. 31–36, Sao Paulo, Brazil, December 2014.
 - [25] X. Lv, X. Lv, X. You, Z. Dong, and J. Han, “Relation extraction toward patent domain based on keyword strategy and attention+BiLSTM model (short paper),” Springer, 2019.
 - [26] D. Zhang and D. Wang, “Relation classification via recurrent neural network [EB/OL],” <http://arxiv.org/abs/1508.01006>.
 - [27] M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni, “IJCAI,” in *Proceedings of the 20th international joint conference on artificial intelligence*, pp. 355–366, Hyderabad, India, 2007.
 - [28] L. Del Corro and R. Gemulla, *ClausIE: Clause-Based Open Information Extraction*, ACM, Rio de Janeiro, Brazil, 2013.
 - [29] G. Stanovsky, J. Michael, L. Zettlemoyer, and I. Dagan, “Supervised open information extraction,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 885–895, Melbourne, Australia, ACL, 2018.
 - [30] L. Cui, F. Wei, and M. Zhou, “Neural open information extraction,” <http://arxiv.org/abs/1805.04270>.
 - [31] G. Ferraro and L. Wanner, “Towards the derivation of verbal content relations from patent claims using deep syntactic structures,” *Knowledge-Based Systems*, vol. 24, no. 8, pp. 1233–1244, 2011.
 - [32] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: pre-training of deep bidirectional transformers for language understanding,” 2019, <http://arxiv.org/abs/1810.04805>.
 - [33] A. Vaswani, N. Shazeer, N. Parmar et al., “Attention is all you need,” *Advances in Neural Information Processing Systems*, pp. 5998–6008, 2017.
 - [34] E. H. Huang, R. Socher, C. D. Manning, and A. Y. Ng, “Improving word representations via global context and multiple word prototypes,” in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pp. 873–882, Minneapolis, MIN, USA, 2012.
 - [35] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, “Relation classification via convolutional deep neural network,” in *25th International Conference on Computational Linguistics COLING 2014*, pp. 2335–2344, Long Beach City, CA, USA, 2014.
 - [36] P. Zhou, W. Shi, J. Tian et al., “Attention-based bidirectional long short-term memory networks for relation classification,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 207–212, Berlin, Germany, 2016.
 - [37] J. Cheng, L. Dong, and M. Lapata, “Long short-term memory networks for machine reading,” *EMNLP*, vol. 2016, pp. 551–561, 2016.
 - [38] A. P. Parikh, O. Täckström, D. Das, and J. Uszkoreit, “A decomposable attention model for natural language inference,” in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Kunming, China, 2016.
 - [39] R. Paulus, C. Xiong, and R. Socher, “A deep reinforced model for abstractive summarization,” 2017, <http://arxiv.org/abs/1705.04304>.
 - [40] D. Zhang and W. Dong, “Relation classification: CNN or RNN? NLPCC-ICCPOL 2016,” *LNAI*, vol. 10102, pp. 665–675, 2016.