

A New Algorithm for the Acquisition of Knowledge from Scientific Literature in Specific Fields Based on Natural Language Comprehension

Hui Wei, Zhilong Dai

Department of Computer Science and Engineering, Fudan University, Shanghai, China
E-mail: weihui@fudan.edu.cn

Received September 11, 2011; revised October 1, 2011; accepted October 8, 2011

Abstract

The acquisition of knowledge and the representation of that acquisition have always been viewed as the bottleneck in the construction of knowledge-based systems. The traditional methods of acquiring knowledge are based on knowledge engineering and communication with field experts. However, these methods cannot produce systematic knowledge effectively, automatically construct knowledge-based systems, or benefit knowledge reasoning. It has been noted that, in specific professional fields, experts often use fixed patterns to describe their expertise in the scientific articles that they publish. Abstracts and conclusions, for example, are key components of the scientific article, containing abundant field knowledge. This paper suggests a method of acquiring production rules from the abstracts and conclusions of scientific articles in specific fields based on natural language comprehension. First, the causal statements in article abstracts and conclusions are extracted using existing techniques, such as text mining. Next, antecedence and consequence fragments are extracted using causal template matching algorithms. As the final step, part-of-speech-tagging production rules are automatically generated according to a syntax parsing tree from the speech pair sequence. Experiments show that this system not only improves the efficiency of knowledge acquisition but also simultaneously generates systematic knowledge and guarantees the accuracy of acquired knowledge.

Keywords: Knowledge Acquisition, Knowledge Representation, Text Mining, Production Rules

1. Introduction

Knowledge acquisition (KA) has long been perceived as the most difficult bottleneck in the construction of knowledge-based systems (KBS). Over the past decade, knowledge engineers have argued over the best means of constructing an effective and reliable KBS. Many researchers view knowledge acquisition as critical [1-5]. For example, Edward Albert Feigenbaum once said, "There are many important problems to be solved in the use, representation, and acquisition of the knowledge. Of them all, the knowledge acquisition is the most important and critical bottleneck" [6]. It is imperative to build automated knowledge acquisition systems.

In specific professional fields, conclusive knowledge and experience are customarily represented concisely in normalized scientific language when stored in text form. Consequently, massive amounts of specialized field knowledge can be obtained from scientific articles. This

can then be used to build knowledge-based systems. From the perspective of knowledge engineering, statements in scientific articles are usually verified, explicit, and expressed in a conclusive and conceptive style. All of these traits closely resemble the requirements of KBS. Additionally, in scientific articles in specific professional fields, these statements are often written in a relatively fixed style that makes them easier to extract.

The abstract and conclusion sections of most scientific articles contain useful field knowledge. Labeling the summary statements in abstracts and conclusions, then, may greatly improve the effectiveness and accuracy of knowledge acquisition. These statements tend to have significant causal relationships, which makes them easier to represent using automatic production rules and easier to translate into field knowledge that can be used by KBS. The implementation of this idea is based on the following proven scientific practices:

- 1) There already exist many NLU and text mining

techniques that can be used to extract knowledge en masse from scientific articles. The issue for knowledge engineers is how to formalize the explicit knowledge in these articles and the difficulty is how to achieve automatic conversion. Existing data mining and analysis techniques, such as text retrieval, text association analysis, and so on, provide effective algorithms to implement the transformation [7].

2) The material processed in this paper focuses on a specific scientific field. In addition, the statements in abstracts and conclusions are often short, conclusive, and designed to focus on a single topic. Therefore, priori knowledge can be used as a guide during the text mining process to improve effectiveness of KA. In the experiments described in this paper, the range of both field knowledge and topic has been limited in order to improve accuracy.

3) The production rule is a mature method of knowledge representation with strong expression ability. It is easily combined with the Drools engine mechanism that is used as inference device in the KBS proposed in this paper.

The method described in this paper takes texts that contain abundant field knowledge, such as professional papers, as input. After labeling, casual statements are then transformed into production rules that can be executed by machine. Transformation between text knowledge and production rules can be achieved automatically through necessary and effective manual interventions, which improve the degree of typical KA automation and realize computer-aided KA.

2. Related Work

As the data show, knowledge acquisition techniques are optimized toward three main goals:

1) To improve the efficiency of knowledge acquisition, which means acquiring knowledge from field experts effectively or adopting semi-automatic knowledge acquisition methods.

2) To extend the scope of knowledge that can be accessed and improve the automation of knowledge acquisition.

3) To simplify the process of conversion from acquired knowledge to production rules that can be executed by machines.

As far as improving KA efficiency is concerned, recent studies have mainly focused on two issues. First, there has been the development of methods and assistant tools that shorten the communication cycle with field experts and guarantee the accuracy of acquired knowledge. These include the famous Repertory Grid method of delimiting and identifying field objects, the integrated

model of KA, the MRM method, and the KADs model-based method, a comprehensive methodology for KA from multiple knowledge sources [8-12]. Second, there has been the development of techniques that improve KA automation and shorten the acquisition cycle [13-16]. These include pattern recognition, machine learning, and text mining techniques, such as the automatic KA method based on inductive learning, the incremental approach to discovering knowledge from text, and knowledge discovery [17-19]. All of these methods have their own inevitable disadvantages. KA methods that use pattern reorganization or machine learning focus primarily on implicit rules contained in mass data and are suitable only for processing data text [20]. The rules obtained are still untested and have to be verified by field experts. KA methods based on text mining are often designed to process large amounts of text and the knowledge obtained is generally inadequate. Additionally, they are always designed to investigate objects and object hierarchies [21,22]. There is great gap between the knowledge rules and production rules used in KBS.

Compared to traditional KA methods, the method suggested in this paper processes causal statements focused on one specific field. The algorithm proposed in this paper, natural language comprehension for rule extraction (NLCRE), is designed to obtain IF-THEN rules from scientific articles by labeling the causal statements in those, extracting antecedence and consequence using causal templates, and generating rules automatically using a syntax parsing tree.

3. Architecture of Knowledge Based System Based on NLCRE

Using the NLCRE algorithm, we developed a new knowledge-based system with a knowledge base that can expand incrementally. The overall architecture of the KBS is shown in **Figure 1**. The KBS architecture is comprised of four modules:

1) Causal-statement-finding module (white)

The function of the causal-statement-finding module is to extract causal statements from the abstracts and conclusions of scientific articles.

2) Production-rule-generation module (gray)

This module is responsible for generating rules from causal statements. Pre-definition templates are constructed based on the characters of causal statements. The antecedence and consequence portions of production rules can be obtained by using templates matching algorithms. Then production rules are then generated by the part-of-speech-tagging and syntax-parsing tree based on natural language comprehension. After being filtered and refined manually, these production rules are added to the

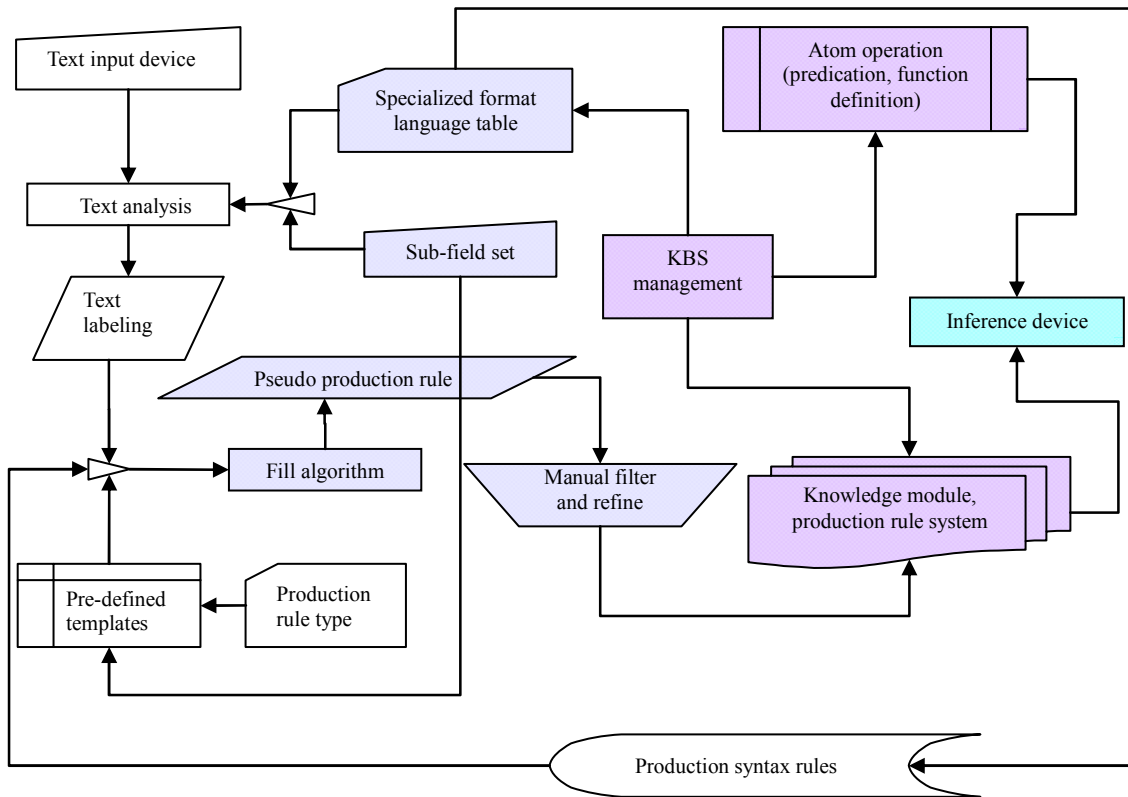


Figure 1. Architecture of Knowledge-Based System Based on NLCRE.

KBS.

3) Field knowledge management module (purple)

Production rules generated from the scientific articles or acquired from field experts are stored and managed in the field KBS. Knowledge management includes knowledge update, knowledge addition, knowledge deletion, and other processes. In order to eliminate redundancy and conflict, these operations must be verified before they are performed.

4) Inference device (green)

Knowledge reasoning must be considered when designing a KBS. In this paper, the Drools java rules engine is the main tool employed toward knowledge reasoning. Drools is an open-source business rule engine and an enhanced java language implementation system based on the RETE algorithm [23].

In addition to traditional functions such as knowledge storage, management, and reasoning, the KBS presented in this paper can also make use of production rules directly extracted from great numbers of scientific papers and alter reasoning results accordingly. Because the production rules can be obtained automatically, the degree of KBS automation is improved and knowledge acquisition time is reduced relative to traditional methods of knowledge acquisition. In this way, computer-assisted knowledge acquisition processes can be realized.

4. NLCRE Production Rule Generation Algorithm

The presentation of field knowledge contained in the natural language of scientific papers is quite different from the requirements of knowledge representation in KBS. Translating natural language to formal language requires an essential change translates labeled text into production rules. This change benefits the process of forming clear knowledge hierarchies and structures in knowledge systems. Furthermore, it improves the extensibility of knowledge and provides an effective means of increasing the efficiency of reasoning.

The NLCRE algorithm can be divided into four stages, as shown in **Figure 2**. These stages will now be described in detail.

4.1. Stage 1: Labeling of Causal Statements

The aim of this stage is to mine casual statements from great numbers of field-specific articles. This paper mainly processes the abstracts and conclusions of scientific papers because they contain the most useful field knowledge.

Text extraction of casual statements takes advantage of NLU and data mining technologies and uses existing

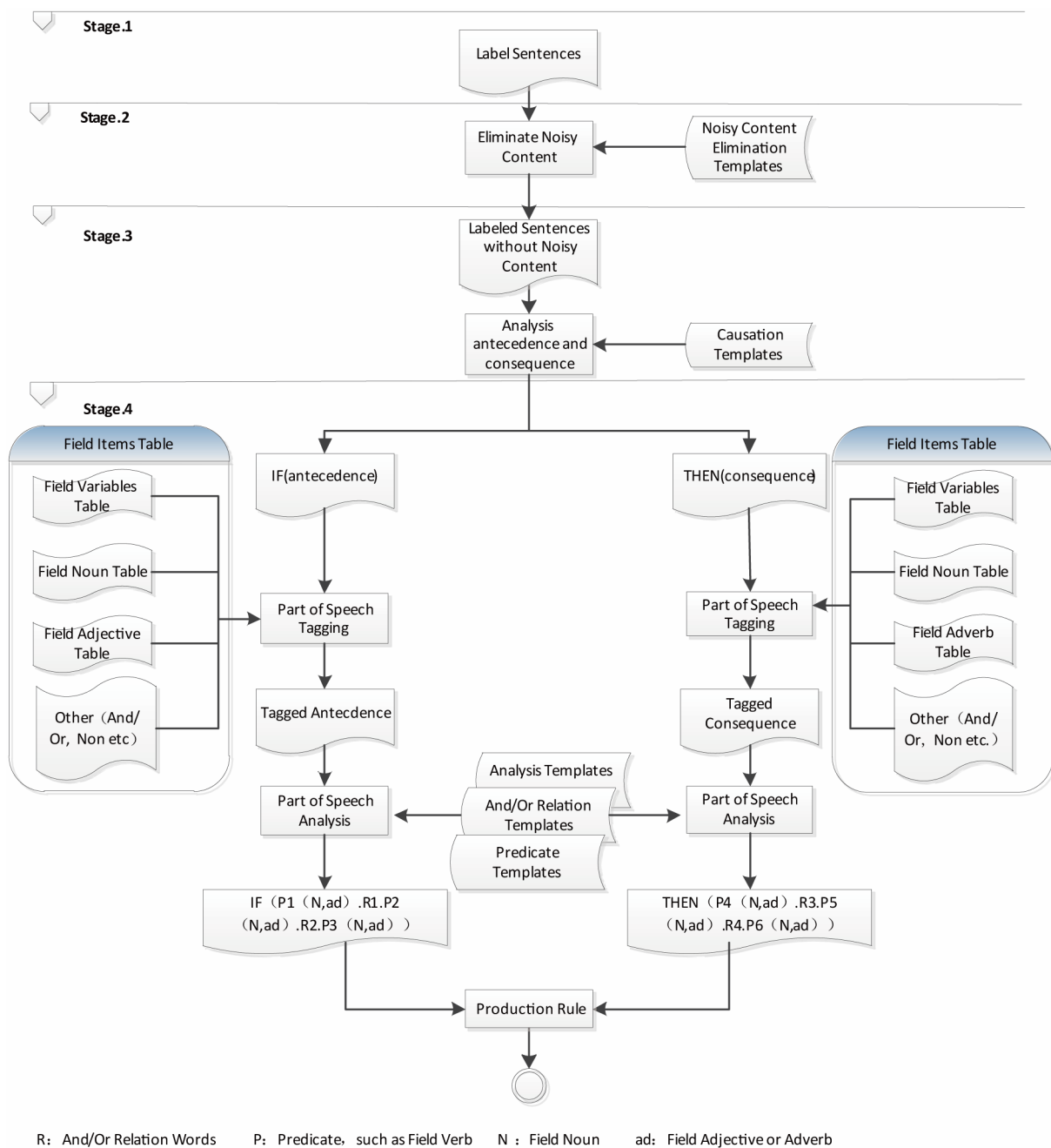


Figure 2. Production rule conversion.

field knowledge for guidance. This stage comprises the following steps:

1) Extract abstracts and conclusions from the scientific articles in question, then label and extract their conclusive sentences.

2) Label field glossaries such as “NEPE propellant,” “metal powder,” “oxidants,” “energy,” and so on. Current effective labeling techniques can be adopted in this step. These include Tregex, which is used to search and operate tree data structure.

3) Label field operation terms, such as “increase,” “decrease,” and “add.” These operation terms will take the glossaries defined in the previous step as operation objects.

4) Identify causal words and use predefined causal templates with the terms and glossaries to extract causal statements from the conclusive sentences.

Causal statements can then be extracted from scientific articles and these sentences can be translated into production rules.

4.2. Stage 2: Elimination of Noise

After stage 1, some noisy words that do not contain knowledge will inevitably be included in the causal statements. These noisy words affect the accuracy of the production rules and should be eliminated before converting the field causal statements to production rules. To this end, noise elimination templates must be constructed according to field and language characters. Because the labeled contents are often conclusive sentences that always have fixed patterns and locations, it is practical to locate most of the similar noisy words with high occurrences and separate them from the labeled causal statements using fuzzy matching algorithms. **Table 1** shows key words often labeled as noisy content.

4.3. Stage 3: Extraction of Antecedence and Consequence Phrases

Once the noisy words have been eliminated, it is crucial to convert the remaining labeled content into rough production rules describable as “IF (antecedence) THEN (consequence)” according to causal templates. This ensures the accuracy of the whole conversion, so it is important to construct effective causal templates with high accuracy. In field-specific scientific papers, causal relationships are usually presented in fixed patterns. Conversion templates meant to translate causal statements to production rules can be constructed by summarizing common presentation patterns. Detailed information is provided in Appendix.

In the Chinese language, some words are used to indicate casual relations explicitly, such as “for,” “because,” “so,” and so on. For sentences that have obvious casual relationships, it is quite easy to identify antecedents and consequents fill in “IF (antecedent) THEN (consequent)” rules. For example, the causal statement “B is the result of A” can be converted to “IF (A) THEN (B).” However, sometimes causal statements are more complicated. Intricate “and” and “or” relationships may exist among many conditions and results. Conditions and results may even intersect. Conversion templates must be designed to deal with this complicated reality. They must also be

Table 1. Key noise content words.

Common noise words	preliminary view, studies suggest that, studies show that, experimental results show that, results show that, studies discover that, preliminary discover, through this experiment, through above analysis that we can see , ... shows, discover ... by computing, discover ... by experiments, discover ... by studies, computing results show, experiments show, discover ... by analyzing ..., as experimental results indicate, in summary, etc.
--------------------	--

tested and refined according to experiment results. This improves the effectiveness of the conversion from causal statements to production rules.

After this stage, causal statements described in natural language are formatted into “IF (antecedent) THEN (consequent)” structures. The antecedents and consequents are refined and formatted to predicate logic to generate production rules correctly. The semantics of the production rules must be consistent with the causal statements. Integrity must be preserved during conversion.

4.4. Stage 4: Generation of Production Rules

The antecedent and consequent of each production rule must be separated from the causal statement. However, they are still represented in natural language and cannot be used to form production rules. This kind of knowledge, then, must be converted into first-order predicate expressions that have clear semantics and are suitable for use in production rules. In this stage the parts of speech (POS) of the antecedent and consequent are tagged. Next, POS tagging results are analyzed according to a parsing tree (**Figure 3**) and a bottom-up merge is carried out. At the same time, the order of the words and phrases in each antecedent and consequent are changed to eventually form predicate verbs with structures such as “predicate (object, value).”

The POS sequences of the antecedents and consequents are produced through this POS-tagging. Then verb-object structures contained in the antecedents and consequents can be found out. Finally, production rules are generated by translating the verb-object structures into first-order predicate expressions. Generally, predicate (object, value) structure is the basic element of production rules. The process of transforming antecedents and consequents into production rules can be focused on generating predicate (object, value) structures.

In this paper, a parsing tree, which is summarized from scientific articles from one specific field, is used as guide to analyze POS sequences and transforming them into predicate structure elements. The conversion process is made up of the following steps:

1) Mark the POSes of the antecedents and consequents and represent these in speech pairs (word, speech tag). For example, an antecedent that literally says “increase content of metal fuel AL” and may be represented with a speech pair list including (increase, V), (metal fuel, DV), (AL, DC), (of, “of”), (content, DV). There are eight POS types in this system, as shown in **Table 2**.

2) Merge the appropriate speech pairs into noun phrases (NPs) according to the parsing tree (**Figure 3**) and mark the speech tag as NP. For example, according to

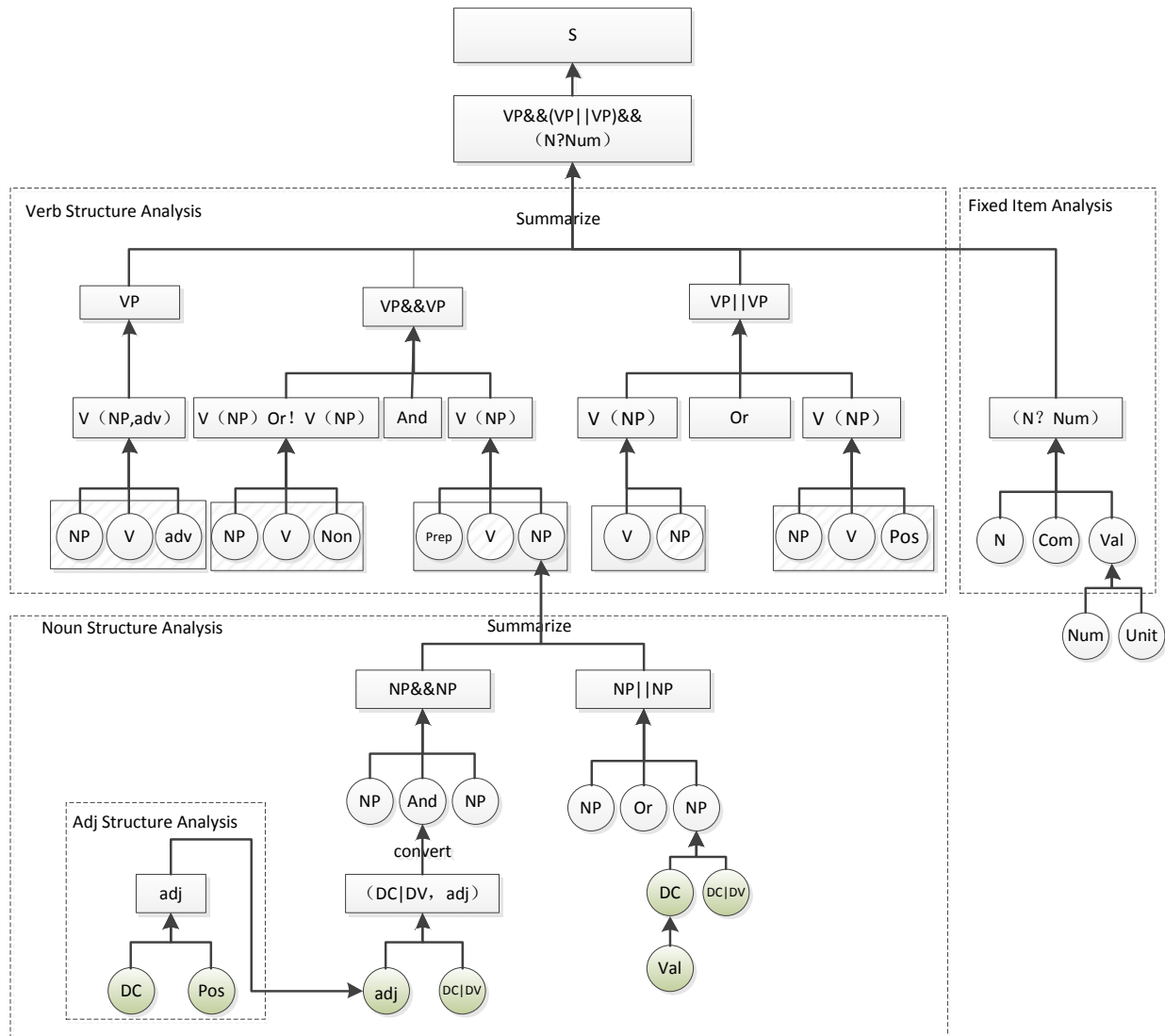


Figure 3. Speech pair list parsing tree.

the “DC.DV->NP” branch of the parsing tree, “metal fuel” and “AL” should be combined and tagged “NP” to form a new speech pair: “(metal fuel, NP).”

3) Combine verbs with NPs and other words according to the parsing tree to form predicate elements with “predicate (object, value)” structure. For example, the speech pairs (increase, V), (metal fuel AL, NP), and (content, V) must be merged together to form a predicate element such as increase (AL, content). In this step, some fixed phrases must also be converted into formal formats according to the corresponding part of the parsing tree (Figure 3).

4) Merge the predicate elements bottom-up according to the parsing tree and change the relative positions of the predicate elements. When the merging process comes to the highest level, rules will be generated.

4.5. Implementation of Production Rule Generation via Algorithm

In the process of obtaining rules from scientific literature, the first three steps can be implemented by using existing mature template matching algorithms. At the end of the third stage, the antecedence and consequence of each rule has been extracted from the causal statements and structured as “IF (antecedence) THEN (consequence).” However, the antecedence and consequence are still represented in the form of natural language. This section will focus on describing the algorithm used to convert them into predicate expressions according to the parsing tree shown in Figure 3.

1) This step can be implemented by querying field variable table, field constant table, etc. (Table 2). After

Table 2. POS in the KBS.

POS Type	Corresponding Causal Statement Element
Field Variable: DV	Variables, such as “metal power,” etc.
Field Constant: DC	Constant, such as AL, etc.
Field Verb: V	Verb, such as “increase,” “improve,” “raise,” etc.
Value: Value	Constant, such as “temperature is 20°C,” etc.
Degree Word: Adj (adjective)	Degree words, such as “mass of,” etc.
Field Adverb: Ad (adverb)	Properties, such as “significantly,” “effectively,” etc.
And/or Relationship Words: And/or	Logical words, such as “at the same time,” “and,” etc.
Negative Words: Non	Negative words, such as “not,” “cannot,” etc.
Preposition: Prep	Generally this is the Chinese word “liao,” following a verb.
Comparing Words: Com	Such as “more than,” “equal to,” “lower than,” “reach,” etc.
Suffix: Pos	Generally this is the Chinese word “de,” following a DC or DV.
Number: Num	Number, such as 18, 32, etc.
Unit Noun: Unit	Unit nouns, such as mm.s-1, %, etc.

this step, speechPairs structured as (word, speech tag) are created. The speechPairs of each antecedent and consequent are stored in the speechPairList.

2) Extract predicate expressions by merging structured speechPairs according to the parsing tree: The merging process must take place in a bottom-up and circular style following the instructions in Stage 4. When deriving NP and predicate elements, conflict resolution may be necessary if multiple routes exist. The merging algorithm is described as follows:

```
//Set merging completion mark:
end = false;
//Keep looping until POS merging is completed and
production rule is generated
//"." indicates adjoining words
while(!end){
//Combine Noun structure circularly
while(multiAdjacent(<word,"DV">Or<
word,"DC">||< word A,"DC">."de"Or< word
A,"DV">."de"||< word A,"DC">.< word B,"adj">||<
word A,"NP">AndOr< word B,"NP">){
Routes route = searchAllMatch-
TreeRoute(NonStructureLevel);
if(!single(route))
route = resolveConfliction(route);
//Summarize current word according to final
route path
word C = compile(words sequence, route);
define(word C, "NP");
```

```
update(speechPairList);
}
```

```
//Change single speech pair with structure < word,"DC">
or < word,"DV"> into < word,"NP">;
if(single(< word A,"DC">)|| single(< word
A,"DV">)){
define(word A, "NP");
update(speechPairList);
}
```

```
//Merge Verb structure words into predicate element cir-
cularly
while(<word A,"adv">.< word B,"V">||< word
A,"NP">.< word B," non">.< word C," V">||< word
A,"NP">.< word B," de">.< word C," V">){
Routes route = searchAllMatch-
TreeRoute(VerbStructureLevel);
if(!single(route))
route = resolveConfliction(route);
// Summarize current word according to final
route path
word C = combine(words sequence, route);
define(word C, "VP");
update(speechPairList);
}
```

```
//Convert fixed phrases into formal formats cir-
cularly
```

```

while(<word A,"Num">.< word B,"Unit">){
    word C = combine(word A, word B);
    define(word C, "Val");
    update(speechPairList);
}

if(antecedence or consequence has been converted to
predicate elements or formal fixed phrases){
    PredicateExpression pe = compose(speechPair,
andOrRelation);
end = true;
}
}
//Generate production rule.
convert(speechPairList, pe);
Production rules obtained through the abovementioned
process can be used in knowledge reasoning immediately
after conflicts have been eliminated. It can then be stored
in corresponding knowledge managing modules.
    
```

5. Experiments

This section describes two instances of the algorithm used for deriving rules from scientific articles under simulated use.

5.1. Example 1

Stage 1: A causal statement is extracted from the paper "Effect of RDX Particle Size on Properties of CMDB

Propellant" [24]. "As experiments have proved, adding moderate amounts of AP and decreasing the granularity and granularity gradation of high-energy composite propellants can improve flammability."

Stage 2: After eliminating the noise in the statement, the result turns to be "through adding moderate AP, decreasing its granularity and granularity gradation in high-energy composite propellant, the flammability will be improved."

Stage 3: The antecedent and consequent are derived by matching casual relation templates (Through A, B: IF (A) THEN (B)) and represented as IF (adding moderate AP, decreasing its granularity and granularity gradation in high-energy composite propellant) THEN (the flammability will be improved).

Stage 4: In this stage, the predicate expressions are derived from the antecedent and consequent according to the parsing tree. This can be implemented following the instructions in Section 4.4.

1) By querying the field variable table (e.g., high-energy composite propellant), field constant table (e.g., AP), etc. (Table 2), the structured speechPairs of the antecedent and consequent are produced as follows: IF({in [.prep]} { high-energy composite propellant [.DV]} {add [.V]} {moderate[.adj]} {AP[.DC]} {,[.And]} {decrease [.V]} its{granularity[.DV]} {and[.And]} {make[.V]} {granularity[.DV]} {gradation[.DV]}) THEN ({improve [.V]} {flammability[.DV]}).

2) The process of merging speechPairs based on the parsing tree is shown in Figure 4:

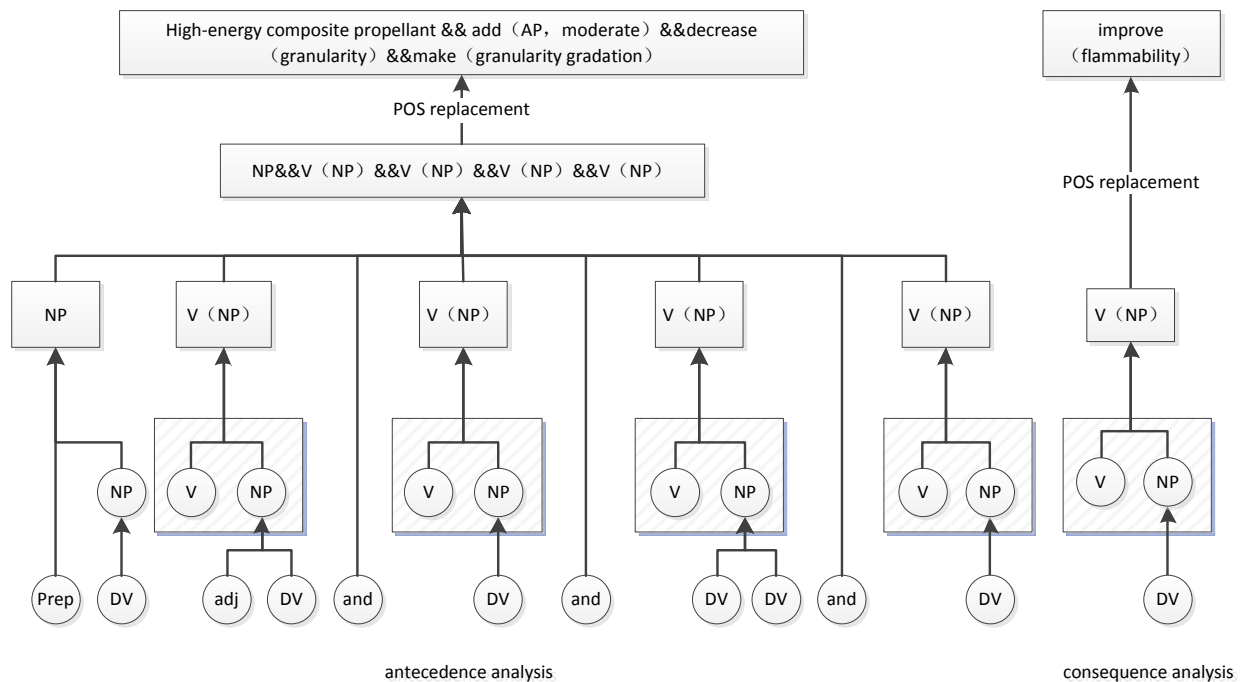


Figure 4. Merging process of example 1.

3) The production rule is finally derived from the causal statement as follows: IF (high-energy composite propellant &&add (AP, moderate) && decrease (granularity) &&make (granularity gradation)) THEN (improve (flammability)).

5.2. Example 2

Stage 1: Another example of a statement with a causal relationship was extracted from the paper “Influence of Ammonium Perchlorate and Aluminum Powder on the Combustion Characteristics of AP-CMDB Propellant” [25]. “If average size of RDX decreases from 92.02 μm to 17.35 μm, pressure index will increase 7.3%.”

Stage 2: The above statement does not contain any noise, so it is not changed during this stage.

Stage 3: By matching casual relation template (If A, B: IF (A) THEN (B)), the antecedent and consequent can be extracted from the statement and represented as follows: IF (average size of RDX decreases from 92.02 μm to 17.35 μm) THEN (pressure index will increase 7.3%).

Stage 4: Based on the above result, the production rule can be derived as follows:

1) After querying field variable table (e.g., average size), field constant table (e.g., RDX), field verb table (e.g., decrease, increase) etc. (shown in **Table 2**), the

structured speechPairs will be produced. The result is: IF({RDX[DC]} {average size[DV]} {92.02[Num]} {μm [Unit]} {decrease[V]} {17.35[Num]} {μm[Unit]}) THEN ({pressure index[DV]} {increase[V]} {7.3[Num]} {% [Unit]}).

2) According to the parsing tree, the merging process of the speechPairs takes place as shown in **Figure 5**.

3) Upon the completion of these processes, the production rule becomes “IF (decrease (RDX average size, 92.02 μm, 17.35 μm)) THEN (increase (pressure index, 7.3%)).”

6. Conclusions and Possibilities for Future Research

This paper is based on a well-noted fact: in specific professional fields, experts often use relatively fixed patterns to describe their findings when they publish scientific articles. Moreover, the abstracts and conclusions, the essence of these articles, contain abundant field knowledge. Labeling the summarizing those statements in abstracts and conclusions that have apparent causal relationships, transforming these relationships into formats that can be used in KBSes, and then using an automatic process to represent this knowledge as production rules may greatly improve the effectiveness and accuracy

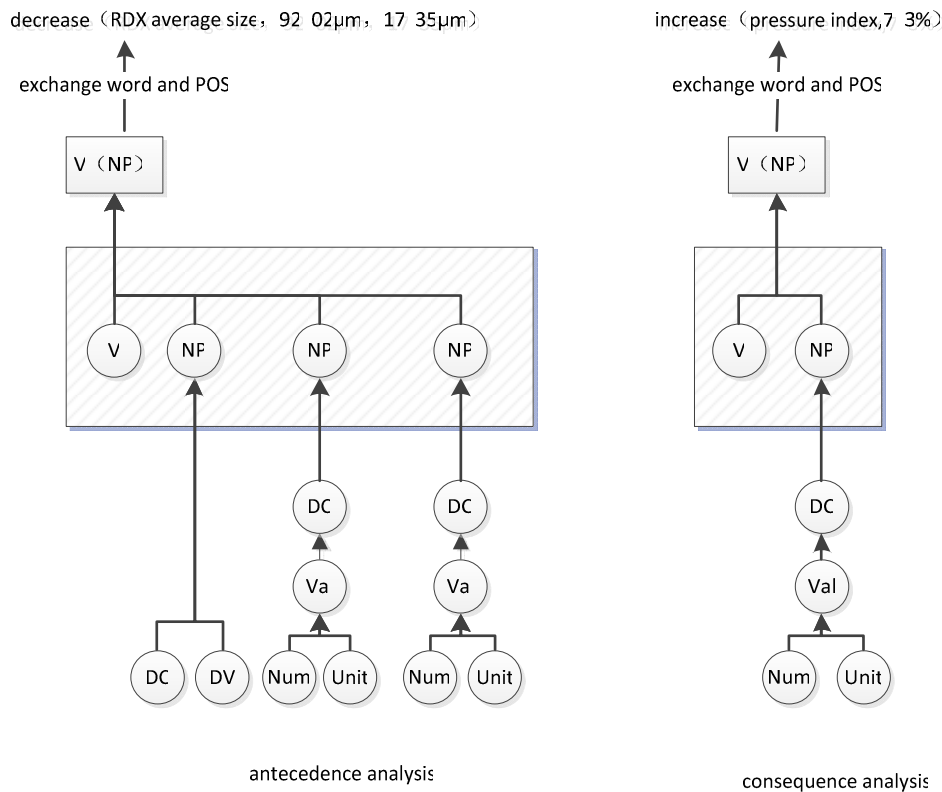


Figure 5. Merging process of example 2.

of knowledge acquisition.

In this paper a pattern-text-mining-based method of acquiring production rules from field-specific papers is proposed. This paper takes texts that contain field knowledge as input. First, the summarizing statements in the abstracts and conclusions of the articles are processed using mature technologies, such as text mining, and causal statements are extracted from them. Then production rules are derived by POS analysis and parsing tree processing.

Two experiments showed that the proposed method can be used effectively to obtain production rules from summarizing statements. In future studies, algorithms used for POS analysis can be investigated to enhance the algorithm's self-learning ability. At the same time, the process of converting causal statements to production rules can be refined to further improve the efficiency of field knowledge acquisition.

7. Acknowledgements

This work was supported by the 973 Program (Project No. 2010CB327900).

8. References

- [1] J. G. Gammack and R. M. Young, "Psychological Techniques for Eliciting Expert Knowledge," Cambridge University Press, London, 1985, pp. 105-112.
- [2] R. R. Hoffman, "The Problem of Extracting the Knowledge of Experts from the Perspective of Experimental Psychology," *AI Magazine*, Vol. 8, No. 2, 1987, pp. 53-67.
- [3] A. L. Kidd, "Knowledge Acquisition—An Introductory Framework," *Knowledge Acquisition for Expert Systems*, Plenum Press, New York, 1987, pp. 1-16.
- [4] J. R. Quinlan, "Fundamentals of the Knowledge Engineering Problem," In: D. Michie, Ed., *Introductory Readings in Expert Systems*, Gordon and Breach, London, 1984, pp. 33-46.
- [5] B. J. Wielinga, B. Bredeweg and J. A. Breuker, "Knowledge Acquisition for Expert Systems," In: R. T. Nossur, Ed., *Advanced Topics in Artificial Intelligence*, Springer-Verlag, Berlin, 1988, pp. 25-58.
- [6] E. A. Feigenbaum, "Expert System in the 1980s," In: A. Bond, Ed., *Infotech State of the Art Report on Machine Intelligence*, Pergamon Infotech Ltd, Maidenhead, 1981, pp. 27-52.
- [7] J. Wei, R. K. Srihari, H. H. Hay and W. Xin, "Improving Knowledge Discovery in Document Collections through Combining Text Retrieval and Link Analysis Techniques," *Seventh IEEE International Conference on Data Mining*, Omaha, 28-31 October 2007, pp. 193-202.
- [8] F. Kennerh and P. Frederick, "Knowledge Acquisition from Repertory Grids Using a Logic of Confirmation," *Knowledge Acquisition Special Issue*, Vol. 108, 1989, pp. 146-147.
- [9] C. Osvaldo, "KAMET: A Comprehensive Methodology for Knowledge Acquisition from Multiple Knowledge Sources," *Expert Systems with Applications*, Vol. 14, No. 1-2, 1998, pp. 1-16. [doi:10.1016/S0957-4174\(97\)00064-X](https://doi.org/10.1016/S0957-4174(97)00064-X)
- [10] S.-C. Lin, S.-C. Lin, S.-S. Tseng and C.-W. Teng, "Dynamic EMCUD for Knowledge Acquisition," *Expert Systems with Applications*, Vol. 34, No. 2, 2008, pp. 833-844. [doi:10.1016/j.eswa.2006.10.041](https://doi.org/10.1016/j.eswa.2006.10.041)
- [11] C. Wang, H. Lan and H. Xie, "An Integrated Model of Knowledge Acquisition: Empirical Evidences in China," *International Conference on Information Management, Innovation Management and Industrial Engineering*, Taipei, 19-21 December 2008, pp. 335-338.
- [12] B. J. Wielinga, A. T. Schreiber and J. A. Breuker, "KADS: A Modeling Approach to Knowledge Engineering," *Knowledge Acquisition*, Vol. 4, No. 1, 1992, pp. 5-53. [doi:10.1016/1042-8143\(92\)90013-Q](https://doi.org/10.1016/1042-8143(92)90013-Q)
- [13] D. Pedro, "Knowledge Discovery via Multiple Models," *Intelligent Data Analysis*, Vol. 2, No. 1-4, 1998. pp. 187-202. [doi:10.1016/S1088-467X\(98\)00023-7](https://doi.org/10.1016/S1088-467X(98)00023-7)
- [14] N. Lavrac and I. Mozetic, "Second Generation Knowledge Acquisition Methods and Their Application to Medicine," *Deep Models for Medical Knowledge Engineering*, Elsevier, New York, 1992, pp. 177-198.
- [15] S. Potter, "A Survey of Knowledge Acquisition from Natural Language," 2003. [http://www.aiai.ed.ac.uk/project/akt/work/stephenp/TMA%20of%20KA from NL. pdf](http://www.aiai.ed.ac.uk/project/akt/work/stephenp/TMA%20of%20KA%20from%20NL.pdf)
- [16] J. Xing and T. Ah-Hwee, "Mining Ontological Knowledge from Domain-Specific Text Documents," *Fifth IEEE International Conference on Data Mining*, Washington DC, 24-30 November 2005, pp. 665-668.
- [17] J. Li and D. Keith, "KDMAS: A Multi-Agent System for Knowledge Discovery via Planning," *American Association for Artificial Intelligence*, 2006, pp. 1877-1878.
- [18] Ö. Akgöbek, Y. S. Aydin, E. Öztemel and M. S. Aksoy, "A New Algorithm for Automatic Knowledge Acquisition in Inductive Learning," *Knowledge-Based Systems*, Vol. 19, No.6, 2006, pp. 388-395. [doi:10.1016/j.knosys.2006.03.001](https://doi.org/10.1016/j.knosys.2006.03.001)
- [19] R. Valencia-García, J. M. A. Ruiz-Sánchez, P. J. Vivancos-Vicente, et al., "An Incremental Approach for Discovering Medical Knowledge from Texts," *Expert Systems with Applications*, Vol. 26, No. 3, 2004, pp. 291-299. [doi:10.1016/j.eswa.2003.09.001](https://doi.org/10.1016/j.eswa.2003.09.001)
- [20] L. Xu, W. Dong, J. H. Wang and S. S. Gu, "A Method of the Knowledge Acquisition Using Rough Set Knowledge Reduction Algorithm Based on PSO," *7th World Congress on Intelligent Control and Automation*, Chongqing, 25-27 June 2008, pp. 5321-5326.
- [21] H. T. Wang, G. C. Cao and Y. Gao, "Design and Implementation of a System for Ontology-Mediated Knowledge Acquisition from Semi-Structured Text," *Chinese Journal of Computers*, Vol. 12, No. 4, 2005, pp. 2010-2018.

- [22] V. Nastase and M. Strube, "Decoding Wikipedia Categories for Knowledge Acquisition," *Proceedings of the 23rd National Conference on Artificial Intelligence*, Chicago, 13-17 July 2008, pp. 1219-1224.
- [23] W. Liu, "Introduction and Implementation of Drools-a Rule Engine Based Java," *Microcomputer Application*, No. 6, 2005, pp. 717-721.
- [24] Q. J. Jiao and J. C. Li, "Effect of RDX Particle Size on Properties of CMDB Propellant," *Chinese Journal of Energetic Materials*, Vol. 15, No. 3, pp. 220-223.
- [25] J. Z. Li, X. Z. Fan and X. G. Lu, "Influence of Ammonium Perchlorate and Aluminum Powder on the Combustion Characteristics of AP-CMDB Propellant," *Chinese Journal of Explosive & Propellants*, Vol. 4, No. 31, 2008, pp. 61-63.

Appendix: Causal Relationships in Scientific Articles

	Original sentence	Production rule
1	If A, B	IF (A) THEN (B)
2	After A, B or B, after A	IF (A) THEN (B)
3	With A, B or B, with A	IF (A) THEN (B)
4	When A, B or B, when A	IF (A) THEN (B)
5	As A, B or B, as A	IF (A) THEN (B)
6	Because A, B or B, because A	IF (A) THEN (B)
7	A is the reason for B	IF (A) THEN (B)
8	When A, after B, C	IF (A AND B) THEN (C)
9	Through A, B	IF (A) THEN (B)
10	B is the result of A	IF (A) THEN (B)