Applying Information Extraction for Patent Structure Analysis

Masayuki Okamoto
Toshiba Corporation
Kawasaki, Japan
m-okmt@acm.org

Zifei Shan
Lattice Data, Inc.
Menlo Park, CA, USA
zifei.shan@lattice.io

Ryohei Orihara
Toshiba Corporation
Kawasaki, Japan
ryohei.orihara@toshiba.co.jp

ABSTRACT

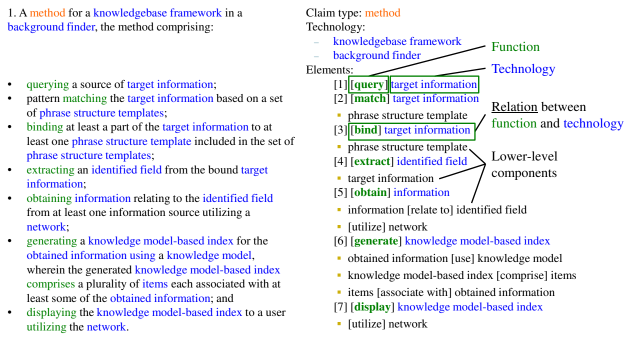
Patent engineers are spending significant time analyzing patent claim structures to grasp the range of technology covered or to compare similar patents in the same patent family. Though patent claims are the most important section in a patent, it is hard for a human to examine them. In this paper, we propose an information-extraction-based technique to grasp the patent claim structure. We confirmed that our approach is promising through empirical evaluation of entity mention extraction and the relation extraction method. We also built a preliminary interface to visualize patent structures, compare patents, and search similar patents.

CCS CONCEPTS

• **Information systems → Information extraction**;

KEYWORDS

Information extraction, patent analysis, visualization



1. A method for a knowledgebase framework in a background finder, the method comprising:

- querying a source of target information;
- pattern matching the target information based on a set of phrase structure templates;
- binding at least a part of the target information to at least one phrase structure template included in the set of phrase structure templates;
- extracting an identified field from the bound target information;
- obtaining information relating to the identified field from at least one information source utilizing a network;
- generating a knowledge model-based index for the obtained information using a knowledge model, wherein the generated knowledge model-based index comprises a plurality of items each associated with at least some of the obtained information; and
- displaying the knowledge model-based index to a user utilizing the network.

(a) Annotated claim text

Claim type: method
Technology:
– knowledgebase framework
– background finder
Elements:
[1] [**query**] target information
[2] [**match**] target information
▪ phrase structure template
[3] [**bind**] target information
▪ phrase structure template
[4] [**extract**] identified field
▪ target information
[5] [**obtain**] information
▪ information [relate to] identified field
▪ [utilize] network
[6] [**generate**] knowledge model-based index
▪ obtained information [use] knowledge model
▪ knowledge model-based index [comprise] items
▪ items [associate with] obtained information
[7] [**display**] knowledge model-based index
▪ [utilize] network

(b) Claim structure

Figure 1: Desired output.

1 INTRODUCTION

Patent engineers are spending significant time analyzing claim sections of patent documents to grasp the scope of legal protection of the inventions or the differences between similar claims. A typical case is that of evaluating a patent family. A patent family is a set of either patent applications or publications in multiple countries to protect a single invention by a common inventor(s) that is then patented in more than one country[1]. When patent engineers analyze the scope of legal protection of a claim, they often analyze claims of other patents in the same family because the scope of a patent may be wider than that of the original claim or may contain a different combination of elements.

Though patent claims are the most important section in a patent, it is hard for a human to examine them. The claim sections of patent documents are typical low-readability technical texts. Considerable research has been done to improve claim readability by modification-based approaches, e.g., simplification, paraphrasing, and summarization, and clarifying-presentation-based approaches [3]. In this paper, we propose an information-extraction-based technique to grasp the patent claim structure for a special user group of patent engineers. Our idea is claim structure extraction with a relation extraction technique. Figure 1 shows an example of the extraction of the structure from a claim text. Applying machine learning-based information extraction techniques instead of using only syntactic parsing is useful for reducing the cost of extracting important terms from patent claims and organizing them. We use DeepDive [12, 15], which has been successfully applied to document analysis tasks [4, 7, 9, 19], as an information extraction platform in the present work.

2 BACKGROUND

2.1 Patent Claim Structure

In general, patent documents consist of several sections, e.g., summary of the invention, title, abstract, background, brief description of the drawings, and claims. A patent contains one or more independent claims that describe the essential features of the invention[2]. Additionally, a patent may contain dependent claims that impose further limitations and restrictions on other dependent or independent claims. Each claim has to be defined in a single sentence [8]. The claim section defines the boundaries of the legal protection of the invention by describing complex technical issues and using specific legal terminology [10]. Each claim consists of preamble, transition, and body. The preamble is an introductory part defining and clarifying the subject matter of the scope of a claim. The transition is the part showing the condition of listed features with regard to the subject matter in the preamble. The body is the part listing the features of the invention[3]. Each invention element in the body

[1] http://www.epo.org/searching-for-patents/helpful-resources/first-time-here/patent-families.html

[2] https://www.epo.org/law-practice/legal-texts/guidelines.html

[3] http://www.wipo.int/edocs/mdocs/africa/en/wipo_pat_hre_15/wipo_pat_hre_15_t_11.pdf

SIGIR'17, August 7–11, 2017, Shinjuku, Tokyo, Japan

978-1-4503-5022-8/17/08…$15.00
DOI: http://dx.doi.org/10.1145/3077136.3080698

text is separated by a semicolon [11]. From the viewpoint of evaluating the value of a patent, it is important to understand the claim structure quickly. Thus, technologies improving the readability of the claim structure are required.

2.2 Related Work

There have been several activities to improve patent claim readability. There are two approaches: the modification approach and the presentation clarification approach. The idea of the modification approach is to change the claim sentence into a more readable form using parsing methodology [13, 14], rule-based methods [17], and paraphrasing and summarization methods [1]. The advantage of this approach is that the claim text is simplified so that patent engineers can understand it in less time. However, there is a risk of changing the meaning of the text [3]. The second approach is presentation clarification. This approach does not change the claim text itself but changes the presentation of the claim to improve the readability. There are several strategies, e.g., aligning claim phrases with relevant text from the description section [16], and segmentation [3]. Segmentation consists of two levels. First, an entire claim is segmented to the components of preamble, transitional phrase and body, using a rule-based approach. Second, a conditional random field is trained to segment the components into clauses.

Our approach combines the advantages of the two approaches described above. In advance, key terms and phrases in claims are annotated by information extraction techniques. When the original claim is being read, highlighted text is shown to improve the readability of the text. If the aim is to grasp the claim structure, a simplified view showing only extracted terms and phrases is shown.

3 CLAIM STRUCTURE EXTRACTION

In this section, we propose an information extraction-based patent claim structure extraction method. Our approach is threefold: extract entity and relation mentions, build claim structure, and implement a patent analysis interface.

3.1 Extraction Target Schema

To analyze patent claim structure, we extract the following information:

Claim type There are many types of claim such as system, method, and apparatus. Usually the claim type is indicated by the first noun(s) of the preamble.

Technology Keywords (important noun phrases) in a claim.

Function How the technologies are used. Basically verbs connected to the noun phrases.

Relation Relation between technology and function. There are two kinds of relations: subject (technology-function) and object (function-technology).

An example of relation is as follows:

- *Using* said first performance data to *compute* a mathematical function ('performance data' is the subject for '*compute*')
- A method of a computer system *predicting* a performance shortage ('performance shortage' is the object for '*predict*')

Subclaim A claim can refer to another claim to extend it.

Example: 2. The system of claim 1 further comprises means ...

Examples except subclaim are shown in Figure 1(b).

3.2 Information Extraction Platform

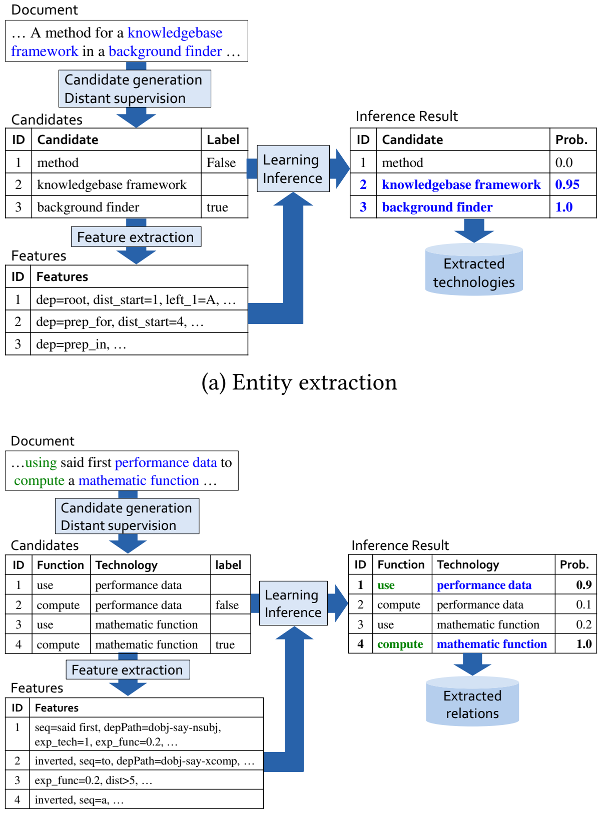
We are using a machine learning-based information extraction approach rather than only a traditional natural language processing (NLP) approach for the following reasons: NLP parsers and taggers are noisy; simply extracting noun phrases is inaccurate; and it is hard to generalize matching dictionaries to unseen terms.

To extract the relations from papers, we use an information extraction platform, DeepDive [12, 15]. DeepDive uses Markov logic network-based inference [2] and distant supervision-based labeling [6] to extract relations from unstructured text. To use DeepDive, the user designs a pipeline of the relation extraction including extractors and inference modules. The output is a set of extracted relations in which estimated probability is assigned for the corresponding relation. We use DeepDive as the information extraction platform because it handles noisy input with the probabilistic inference, it integrates NLP with domain-specific knowledge, and it allows systematic error analysis. Also DeepDive worked in several domains such as extracting knowledge about biodiversity from papers [9]. The system has better precision and recall than human efforts. A DeepDive-based system also won the Knowledge Base Population competition [18].

3.3 Pipeline

This section describes the pipeline to extract information shown in Section 3.1. Our patent analysis pipeline consists of entity mention extraction and relation mention extraction. Entity mention extraction consists of candidate generation, feature extraction, distant supervision, learning, and inference. Relation mention extraction consists of steps similar to those of entity mention extraction, built on extracted technology / function mentions. As the preprocess to get part-of-speech (POS) tags, named entity recognition (NER) tags, and dependency paths, we ran Stanford CoreNLP [5]. Figure 2 shows the summary of our application pipeline.

3.3.1 Entity mention extraction. In this section, 'technology' term generation is exemplified as shown in Figure 2(a). Firstly, the candidate generation step extracts the longest phrases that satisfy the following conditions: each word has POS tag of NN* / JJ / VBN / VBG; last word has POS tag of NN*; not all words are stopwords; first word is not an ordinal number (e.g., 'first' and 'second'); length of each word is greater than 1. Secondly, the feature extraction step generates the following features: features generated by DeepDive generic feature library (e.g., bag-of-words and POS tags of mentions or relations), distance to the start of claim / sentence, dependency label (e.g., 'nn' and 'amod'). Thirdly, the distant supervision step labels each candidate phrase with the rules shown in Table 1. Finally, the learning and inference step uses the inference rules (constraints) between two technology phrase candidates t1 and t2: (1) t1 equals t2 if t1 and t2 have the same words, (2) t1 equals t2 if t1's words start with t2's, and (3) t1 equals t2 if t1's words end with t2's. For other types of entities, a similar step is used.



ID	Candidate	Label
1	method	False
2	knowledgebase framework	
3	background finder	true

ID	Features
1	dep=root, dist_start=1, left_1=A, …
2	dep=prep_for, dist_start=4, …
3	dep=prep_in, …

ID	Candidate	Prob.
1	method	0.0
2	**knowledgebase framework**	**0.95**
3	**background finder**	**1.0**

(a) Entity extraction

ID	Function	Technology	label
1	use	performance data	
2	compute	performance data	false
3	use	mathematic function	
4	compute	mathematic function	true

ID	Features
1	seq=said first, depPath=dobj-say-nsubj, exp_tech=1, exp_func=0.2, …
2	inverted, seq=to, depPath=dobj-say-xcomp, …
3	exp_func=0.2, dist>5, …
4	inverted, seq=a, …

ID	Function	Technology	Prob.
1	**use**	**performance data**	**0.9**
2	compute	performance data	0.1
3	use	mathematic function	0.2
4	**compute**	**mathematic function**	**1.0**

(b) Relation extraction

Figure 2: Summary of information extraction pipeline.

Table 1: Distant supervision rules for technology mention extraction (P: positive, N: negative).

No.	Label	Condition
1	P	In a domain-specific positive KB (e.g., freebase “software-genre”)
2	P	Matches the pattern ‘said · · · performed/./,/;/and/to/comprising/VBZ’
3	P	Matches patterns (the · · · in, one · · · of, one · · · and, for · · · of, · · ·)
4	P	Matches patterns (‘set of [*] ,|;|.’, ‘a [*] .*ing’, ‘a/an/the [*] ,|;|.’, ‘by the [*] of’, ‘* [*] consist’)
5	P	In the second level hierarchy (in a claim element) and the words before are in a/the/empty
6	N	Has intersection with a claim type
7	N	In a domain-independent non-technology dictionary (139 terms)
8	N	Matches patterns (‘^first’, ‘^second’, ‘^\w+ of’, ‘^sub .’, ‘=’, ‘\bEQU\b’, ‘#’)
9	N	Mention is three letters or less
10	N	Matches the pattern ‘in [order] to’
11	N	Some special handling for mentions starting with VBG
12	N	If any word has a person/location/organization NER tag

Table 2: Preliminary evaluation result.

Schema	Precision	Recall
Claim type	99%	99%
Technology	82%	77%
Function	77%	65%
Subject relation	68%	69%
Object relation	91%	39%
Subclaim reference	100%	100%

3.3.2 Relation mention extraction. In this section, technology-function (subject) relation is exemplified as shown in Figure 2(b). The whole step is similar to the entity mention extraction. For the candidate generation, we use technology-function pairs in the same sentence (claim). For the feature extraction step, we use DeepDive's generic feature library and expectation of technology / function calculated in the entity mention extraction step. For the distant supervision, we use linguistic patterns with dependency path, word sequence, and so on. For the function-technology (object) relation, the same step is used.

4 EMPIRICAL EVALUATION OF INFORMATION EXTRACTION

We evaluated hundreds of mentions. As a dataset, we used 12,972 granted U.S. patents in the domain of “Data processing: artificial intelligence”. One of the authors annotated the test data: 708 technology mentions, 311 function mentions, 126 claim type mentions, 102 subclaim references, 62 technology (subject)-function relations, and 113 function-technology (object) relations. The summary of mention extraction and relation extraction for this dataset is shown as Table 2. Although our experiment is limited in scale and the inter-annotator agreement is the next step, this work clarified the possibility of making applications to extract claim structure.

5 APPLICATION

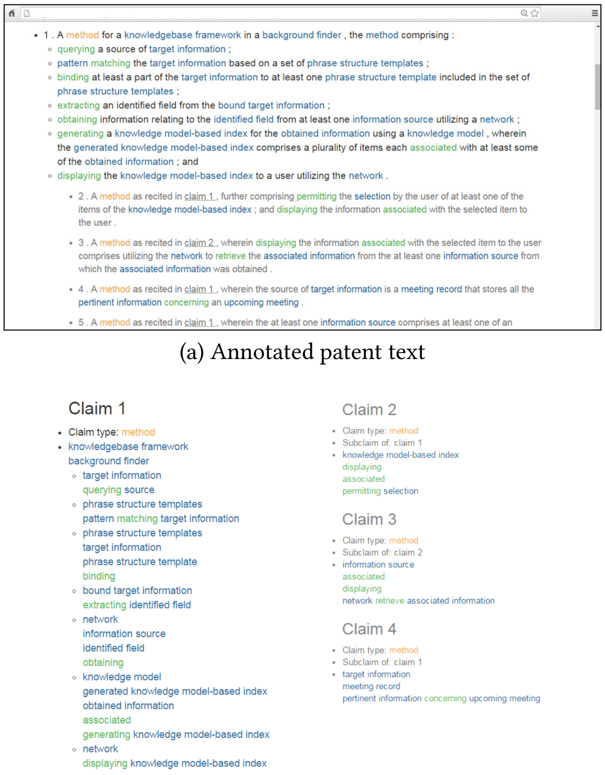
We built a preliminary web-based interface to visualize patent structures, compare patents, and search patents. Functions are as follows: view annotated claim text, view claim structure, compare two patents, and search similar patents for a given (query) patent based on extracted components. The following subsections show the corresponding use cases.

5.1 Viewing annotated claim

Extracted entities and relations can be used to build claim structures. These claim structures are useful for analyzing patents. Figure 3 shows the visualized patent claim. Figure 3(a) shows an annotated patent claim and Figure 3(b) shows the patent structure. Using these views, patent engineers can grasp both the key feature of a patent claim and the structure of the claim.

5.2 Comparing two patent claims

When patent engineers want to compare patent claims precisely, they can use the patent comparison view. Figure 4 shows an example of comparing two patents. Each claim is analyzed in the way described in the previous subsection. Grey terms highlight



- 1 . A method for a knowledgebase framework in a background finder , the method comprising :
  - querying a source of target information ;
  - pattern matching the target information based on a set of phrase structure templates ;
  - binding at least a part of the target information to at least one phrase structure template included in the set of phrase structure templates ;
  - extracting an identified field from the bound target information ;
  - obtaining information relating to the identified field from at least one information source utilizing a network ;
  - generating a knowledge model-based index for the obtained information using a knowledge model , wherein the generated knowledge model-based index comprises a plurality of items each associated with at least some of the obtained information ; and
  - displaying the knowledge model-based index to a user utilizing the network .
    - 2 . A method as recited in claim 1 , further comprising permitting the selection by the user of at least one of the items of the knowledge model-based index ; and displaying the information associated with the selected item to the user .
    - 3 . A method as recited in claim 2 , wherein displaying the information associated with the selected item to the user comprises utilizing the network to retrieve the associated information from the at least one information source from which the associated information was obtained .
    - 4 . A method as recited in claim 1 , wherein the source of target information is a meeting record that stores all the pertinent information concerning an upcoming meeting .
    - 5 . A method as recited in claim 1 , wherein the at least one information source comprises at least one of an

(a) Annotated patent text

Claim 1

- Claim type: method
- knowledgebase framework
  background finder
  - target information
    querying source
  - phrase structure templates
    pattern matching target information
  - phrase structure templates
    target information
    phrase structure template
    binding
  - bound target information
    extracting identified field
  - network
    information source
    identified field
    obtaining
  - knowledge model
    generated knowledge model-based index
    obtained information
    associated
    generating knowledge model-based index
  - network
    displaying knowledge model-based index

Claim 2

- Claim type: method
- Subclaim of: claim 1
- knowledge model-based index
  displaying
  associated
  permitting selection

Claim 3

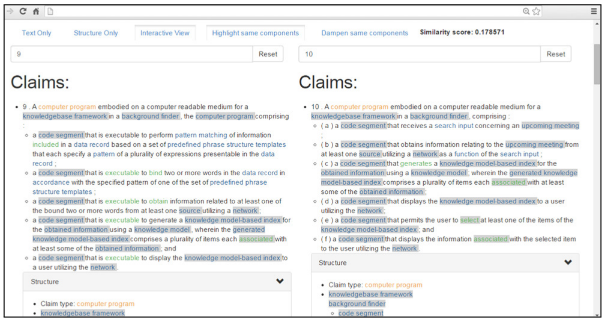
- Claim type: method
- Subclaim of: claim 2
- information source
  associated
  displaying
  network retrieve associated information

Claim 4

- Claim type: method
- Subclaim of: claim 1
- target information
  meeting record
  pertinent information concerning upcoming meeting

(b) Patent structure

Figure 3: Annotated patent claims.



Similarity score: 0.178571

9 Reset

Claims:

- 9 . A computer program embodied on a computer readable medium for a knowledgebase framework in a background finder , the computer program comprising :
  - a code segment that is executable to perform pattern matching of information included in a data record based on a set of predefined phrase structure templates that each specify a pattern of a plurality of expressions presentable in the data record ;
  - a code segment that is executable to bind two or more words in the data record in accordance with the specified pattern of one of the set of predefined phrase structure templates ;
  - a code segment that is executable to obtain information related to at least one of the bound two or more words from at least one source utilizing a network ;
  - a code segment that is executable to generate a knowledge model-based index for the obtained information using a knowledge model , wherein the generated knowledge model-based index comprises a plurality of items each associated with at least some of the obtained information ; and
  - a code segment that is executable to display the knowledge model-based index to a user utilizing the network .

Structure

- Claim type: computer program
- knowledgebase framework

10 Reset

Claims:

- 10 . A computer program embodied on a computer readable medium for a knowledgebase framework in a background finder , comprising :
  - (a) a code segment that receives a search input concerning an upcoming meeting ;
  - (b) a code segment that obtains information relating to the upcoming meeting from at least one source utilizing a network as a function of the search input ;
  - (c) a code segment that generates a knowledge model-based index for the obtained information using a knowledge model ; wherein the generated knowledge model-based index comprises a plurality of items each associated with at least some of the obtained information ;
  - (d) a code segment that displays the knowledge model-based index to a user utilizing the network ;
  - (e) a code segment that permits the user to select at least one of the items of the knowledge model-based index ; and
  - (f) a code segment that displays the information associated with the selected item to the user utilizing the network .

Structure

- Claim type: computer program
- knowledgebase framework
  background finder
  - code segment

Figure 4: Comparing two patents. Grey terms highlight the same extracted components in two patents.

the same extracted components in the two patents. We can grasp how a patent claim covers another patent claim. Additionally the similarity score is shown at the top of the page. This score is also used for the claim search function.

5.3 Searching similar patent claims

We also implemented the preliminary version of a similar claim search function. It calculates the similarity score using simple cosine similarity of extracted entities and relations.

6 CONCLUSIONS

We proposed a claim structure analysis method that uses an information extraction technique. Applying machine learning-based information extraction techniques instead of using only syntactic parsing is useful for reducing the cost of extracting important terms from patent claims and organizing them. We also built a preliminary interface to visualize patent structures, compare patents, and search similar patents.

Our future work includes: compare our method against other baseline methods, improve extraction quality, improve patent search quality, improve comparison interface with alignment functionality, and generalize to other domains.

REFERENCES

[1] Nadjet Bouayad-Agha, Gerard Casamayor, Gabriela Ferraro, Simon Mille, Vanesa Vidal, and Leo Wanner. 2009. Improving the Comprehension of Legal Documentation: The Case of Patent Claims. In *Proceedings of International Conference on Artificial Intelligence and Law (ICAIL 2009)*. 78–87.

[2] Pedro Domingos and Daniel Lowd. 2009. *Markov Logic: An Interface Layer for Artificial Intelligence.* Morgan and Claypool Publishers.

[3] Gabriela Ferraro, Hanna Suominen, and Jaume Nualart. 2014. Segmentation of patent claims for improving their readability. In *Proceedings of Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR 2014)*. 66–73.

[4] Emily K. Mallory, Ce Zhang, Christopher Ré, and Russ B. Altman. 2015. Large-scale extraction of gene interactions from full-text literature using DeepDive. *Bioinformatics* (2015).

[5] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of ACL 2014*. 55–60.

[6] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of ACL 2009*. 1003–1011.

[7] Masayuki Okamoto, Yuichi Miyamura, Ayana Yamamoto, Shuichi Toriyama, and Kentaro Takagi. 2016. Automatic Property Visualization for Material Survey Support. In *Proceedings of International Symposium on Semiconductor Manufacturing (ISSM 2016)*.

[8] Peter Parapatics and Michael Dittenbach. 2009. Patent Claim Decomposition for Improved Information Extraction. In *Proceedings of International Workshop on Patent Information Retrieval (PaIR 2009)*. 33–36.

[9] Shanan E. Peters, Ce Zhang, Miron Livny, and Christopher Ré. 2014. A Machine Reading System for Assembling Synthetic Paleontological Databases. *PLoS ONE* 9, 12 (2014).

[10] David Pressman. 2006. *Patent It Yourself.* Nolo, Berkeley, CA.

[11] David V. Radack. 1995. Reading and Understanding Patent Claims. *JOM* 47, 11 (1995), 69.

[12] Christopher Ré, Amir Abbas Sadeghian, Zifei Shan, Jaeho Shin, Feiran Wang, Sen Wu, and Ce Zhang. 2014. Feature Engineering for Knowledge Base Construction. *IEEE Data Engineering Bulletin* 37, 3 (2014), 26–40.

[13] Svetlana Sheremetyeva. 2003. Natural language analysis of patent claims. In *Proceedings of ACL Workshop on Patent Corpus Processing (PATENT 2003)*. 66–73.

[14] Svetlana Sheremetyeva. 2014. Automatic Text Simplification For Handling Intellectual Property. In *Proceedings of the Workshop on Automatic Text Simplification: Methods and Applications in the Multilingual Society*. 41–52.

[15] Jaeho Shin, Sen Wu, Feiran Wang, Christopher De Sa, Ce Zhang, and Christopher Ré. 2015. Incremental knowledge base construction using DeepDive. *Proceedings of the VLDB Endowment* 8, 11 (2015), 1310–1321.

[16] Akihiro Shinmori and Manabu Okumura. 2004. Aligning Patent Claims with Detailed Descriptions for Readability. In *Working Notes of NTCIR-4*.

[17] Akihiro Shinmori, Manabu Okumura, Yuzo Marukawa, and Makoto Iwayama. 2003. Patent claim processing for readability: structure analysis and term explanation. In *Proceedings of ACL Workshop on Patent Corpus Processing (PATENT 2003)*. 56–65.

[18] Mihai Surdeanu and Heng Ji. 2014. Overview of the English Slot Filling Track at the TAC2014 Knowledge Base Population Evaluation. In *Proceedings of TAC-KBP 2014 Workshop*.

[19] Ayana Yamamoto, Yuichi Miyamura, Kouta Nakata, and Masayuki Okamoto. 2017. Company Relation Extraction from Web News Articles for Analyzing Industry Structure. In *Proceedings of IEEE International Conference on Semantic Computing (ICSC 2017)*. 89–92.