# Automatic Text Simplification For Handling Intellectual Property (The Case of Multiple Patent Claims)

**Svetlana Sheremetyeva**
National Research South Ural State University, pr. Lenina 76, 454080 Chelyabinsk, Russia
LanA Consulting ApS, Moellekrog 4, Vejby, 3210, Copenhagen, Denmark
lanaconsult@mail.dk

## Abstract

Handling intellectual property involves the cognitive process of understanding the innovation described in the body of patent claims. In this paper we present an on-going project on a multi-level text simplification to assist experts in this complex task. Two levels of simplification procedure are described. The macro-level simplification results in the visualization of the hierarchy of multiple claims. The micro-level simplification includes visualization of the claim terminology, decomposition of the claim complex structure into a set of simple sentences and building a graph explicitly showing the interrelations of the invention elements. The methodology is implemented in an experimental text simplifying computer system. The motivation underlying this research is to develop tools that could increase the overall productivity of human users and machines in processing patent applications.

## 1 Introduction

In today's highly-competitive marketplace much of industrial companies' true worth relates to intellectual property protected by patents. However, a great deal of patents is not used to raise standards across industries as much as they could. In US alone more than 95% of all active patents are not licensed to a single third party and do not earn the first dollar of licensing revenue. Part of the problem is that patents can be difficult to understand and value as they are written in dense, arcane legal language that only a technical expert can read (http://patentproperties.com/patentinnovations.html).

Moreover, even patent experts, whose task is to conduct analysis of patent documents, e.g., for novelty, scope of protection or value can spend quite a time and effort to clearly understand a crucial part of a patent document, claims. The patent claim is the only part of a patent that defines the scope of inventor's rights. Linguistically the claim is the most difficult information carrier. Patent law demands the claim to be written as a single albeit very complex and long sentence, no matter that it might run for a page or so. Figure 1 shows a short fragment of a claim, just to illustrate what is said above.

*Claim 1. A grinding tool for profile strips of wood or the like, comprising a plurality of grinding segments arranged in at least two rows; at least two base bodies, each associated with one of said rows of said grinding elements, said base bodies being movable relative to one another, said grinding segments of one of said rows being offset relative to said grinding segments of the other of said rows so that said rows of said grinding segments are insertable into one another over at least a part of a respective length thereof;.....and clamping means including two clamping elements associated with and located at each side of a respective one of said base bodies so as to engage said grinding segments, said two clamping elements including an inner clamping element which is basket-shaped and has a plurality of webs which are spaced from one another by respective angular distances and lie under said grinding segment receivers, and another clamping element which has a plurality of intermediate spaces into which said webs of said inner basket-shaped clamping element are insertable.*

Figure 1. A fragment of Claim1 of the US patent 4,777,771. This patent has 24 claims.

The limited space of this paper does not allow us enclosing in the current description a real life patent claim section, but an interested reader can consult any patent bank site.





*Proceedings of the Workshop on Automatic Text Simplification: Methods and Applications in the Multilingual Society*, pages 41–52,
Dublin, Ireland, August 24th 2014.

This problem of patent expertise is further complicated by the fact that a patent document, as a rule, contains not just one but a large number of claims that should be read and interpreted as a whole. Anybody who has seen patent claims at least once will find it unnecessary to calculate claim readability indices to get persuaded that the claim text is extremely low readable. Traditional readability formulas normally take into account the number of words per sentence or/and the number of "hard", be it long or low frequency, words per sentence (Kincaid, Fishburne, Rogers, & Chissom, 1975; Brown, 1998; Greenfield, 2004). Both the first and the second ratio will be equal to the number of words in a claim sentence where practically all words are "hard" terms, some of them used for the first time. The same goes for the claim syntactic structure.

Patent experts attending to their examination tasks normally perform simplification of a claim text manually. Evidently, there is a great need for tools that could automate this process. The need has already attracted attention of R&D groups working in the field of text processing. Given the linguistic complexity of the claim it is not surprising that practically all reports related to the patent/claim simplification research describe on-going projects rather than completed studies or development (see Section 2 for references). In this paper we attempt to complement existing achievements by presenting our research in the area and suggest text simplification techniques to facilitate understanding/readability of both, the whole section of multiple claims in a patent document, and an individual claim.

The specificity of our approach is primarily motivated and conditioned by the fact that in patent examination patent experts cannot afford analyzing a simplified claim text where the content has been changed during the simplification procedure. Not a single word in the claim could be changed or omitted. Even the use of synonyms, let alone the omission of claim structural elements (pruning), can change the scope of the invention and result in patent infringement and, hence, court cases. All these put our work out of the mainstream in the text simplification research. However it meets the definition of text simplification as a process of making the text more comprehensible for a targeted audience. It should be also noted that though this study is primarily addressed to patent experts, our simplification solutions might be useful for both laypeople and machines meant to automatically process patents, e.g., information retrieval or machine translation systems.

The rest of the paper is organized as follows. Section 2 is devoted to related work. Section 3 discusses challenges in the field of claim simplification. Section 4 describes our approach to claim simplification on a macro-level that addresses the whole body of multiple patent claims. In sections 4 and 5 we suggest some solutions to the simplification of a single claim, which we call a micro-level simplification. Further in Section 6 we present evaluation results and summarize our on-going research in Conclusions.

## 2 Related work

Research on automatic text simplification aims at developing techniques and tools that could make texts more comprehensible for certain types of targeted audience/readers. The mainstream of text simplification is developing methodologies and tools for general types of texts that address people with special needs, such as poor literacy readers (Aluisio et al. 2010), readers with mild cognitive impairment (Dell'Orletta et al., 2011), elderly people (Bott et al., 2012), language learners of different levels (Crossley and McNamara, 2008) or just "regular" readers (Graesser et al., 2004). Text simplification is most often performed on the sentence level. Simplifying texts to provide more comprehensible input to a targeted audience the developers generally work within two approaches: an intuitive approach and a structural approach. An intuitive approach relies mainly on the developers' intuition and experience (Allen, 2009) that leads to using less lexical diversity, less sophisticated words, less syntactic complexity, and greater cohesion. A structural approach depends on the use of structure and word lists that are predefined by the intelligence level, as typically found in targeted readers. The latter is defined by readability formulas. Traditional readability formulas are simple algorithms that measure text readability based on sentence length and word length. Later research on readability suggests formulas that reflect the psycholinguistic and cognitive processes of reading (Crossley et al.2011).

At the linguistic level, simplified texts are largely modified to control the complexity of the lexicon and the syntax. Automated text simplification tools are trying to achieve this purpose by combining linguistic and statistical techniques and penalize writers for polysyllabic words and long, complex sentences. (Siddharthan, 2002) describe the implementation of the three stages - analysis, transforma-

tion and regeneration, system that lay particular emphasis on the discourse level aspects of syntactic simplification. Some works on text simplification use parallel corpora of original and simplified sentences (Petersen & Ostendorf, 2007). There are works where text simplification is treated as a "translation task within a RBMT (Takao and Sumita. 2003). In (Specia, 2010) text simplification is developed in the Statistical Machine Translation framework, given a parallel corpus of original and simplified texts, aligned at the sentence level. In (Poornima et al.2011) a rule based technique is proposed to simplify the complex sentences based on connectives like relative pronouns, coordinating and subordinating conjunctions. Sentence simplification is expressed as the list of sub-sentences that are portions of the original sentence. (Bott, et al., 2012) describe a hybrid automatic text simplification system which combines a rule based core module with a statistical support module that controls the application of rules in the wrong contexts.

The approaches to patent claim simplification can be roughly put into two groups. Studies of the first group try to adapt to the patent domain general text simplification techniques and involve lexical and/or structural substitution, pruning, paraphrasing, etc. For example, in (Shinmori et al., 2003) the discourse structure of the patent claim is built by means of a rule-based technique; each discourse segment is then paraphrased. In (Mille and Wanner, 2008) the claim sentence (by means of lexical and punctuation clues) is segmented into clausal units, that are then compressed into a summary. The simplification methods proposed by this group of researches to some extent change the original content of the claim that might not always be desirable, especially for patent experts.

Another group of studies focuses on segmenting, reformatting or highlighting certain parts of the patent claim without changing the content of the original. For example, in one of the earlier works a rule-based technique was developed for decomposing the complex sentence of a claim into a set of simple sentences while preserving the initial content (Sheremetyeva, 2003). Most recently (Shinmori et al., 2012) suggested aligning claim phrases with explanatory text from the description section, while (Ferraro et al., 2014) proposed an approach that involves highlighting the claim segments borders and reformatting the original text so as to emphasis segments with the identified border marker. This approach does not involve any syntactic restructuring, just visualization of claim segments.

In general, due to the linguistic complexity of patent claims all research on automatic claim simplification make extensive use of rule-based methods possibly augmented with statistical techniques. Text segmentation is performed on two levels. First the claim in segmented into 3 information-relevant parts, the preamble, transition and body and then the claim body is further segmented into smaller parts, often clausal structures.

To the best of our knowledge practically all publications on claim simplification consider individual claims, while in real life most patents contain multiple interrelated claims of different types and a patent reader has to understand the whole range of information in the claim section. The cited studies address laypeople that are not trained to read patent claims**.** However, there is also a great demand for claim readability tools among patent experts who have to perform thorough and tedious work on claim analysis for different examination tasks on a daily basis. When accessing the prototype systems or methodologies, the developers normally evaluate the correctness of their own intuitive understanding how a simplified claim should look. No studies on end-user requirements or user-centered evaluation have been reported so far. In our work among others we have tried to address the above issues.

Our research includes the following steps:

- Extraction of expert knowledge about their needs and procedure of claim analysis
- Acquisition of linguistic knowledge about the patent claim sublanguage
- Developing a prototype claim simplification system that meets expert expectations.

## 3 Challenges in claim simplification

In preparing for this research we have investigated professional instructions (Pressman. 2006; Radack, 1995) on how to read patent claims and conducted extensive interviews with patent experts of several companies in the US and Europe handling intellectual property[1]. The recommendations are as follows. The first step towards understanding a claim is to identify its information parts, preamble, transition and the body. Another recommendation is to identify and mark the elements of the invention spelled

[1] The confidentiality policy of these companies does not allow us discosing them in this paper.

out in the body of the claim. Element markup is useful not only for proper understanding of the claim but also because claims have to be supported by the description. Any terms used in claims must be found in the description. Hence, there is a demand to automate patent terminology extraction that could underlie terminology markup, e.g., by highlighting.

In real practice the examiners manually decompose the claim in a tree with noun terminology and predicates (verbs, adjectives and prepositions) on separate indented lines to clearly see the invention elements and their interrelations. Hence there is a need to automate the construction of such element-relation diagrams for every particular claim. The experts we have interviewed were also very enthusiastic about a tool that could decompose a complex claim sentence into a set of simple sentences-features of the invention, provided the content of the claim is preserved. It is evident that building such a tool is a much more demanding task than any other as it clearly cannot rely on statistical methods only but also requires extensive linguistic knowledge and rule-based techniques.

Most of patents contain a large number of claims that can claim experts have to interpret related to each other. There are two basic types of claims: the *independent claims*, which stand on their own, and the *dependent claims*, which depend on one or several claims and should be interpreted in conjunction with their parents. Any dependent claim which refers to more than one other claim is a *multiply dependent claim* that should also be visualized in a simplifying tool.

Based on the extracted expert demands and analyzing procedures we suggest two levels of patent claim simplification that should necessarily preserve the claim section content:

- the macro-level simplification resulting in the visualization of the hierarchy of claims explicitly showing their interdependence (type: dependent/independent, parents and children)
- the micro-level simplification of one claim that includes
    - visualization of the claim terminology
    - decomposition of a claim complex structure into a set of simple sentences
    - building a diagram explicitly showing the interrelations of invention elements.

The micro-level claim simplification is extremely challenging as cannot but require the NLP techniques and elaborate and extensive linguistic resources that for our purpose do not exist so far.

## 4 Macro-level simplification

The macro-level simplification improves the readability of the whole section of multiple claims in a patent document. For this purpose we have developed a patent macro-analyzer that takes as input a whole patent document and outputs the hierarchy of claims with a lot of accompanying information relevant for patent examination. In particular, the macro-analyzer automatically performs the following successive steps:

- Segmentation of the claim section from the rest of the input patent document
- Segmentation of individual claims from the body of the claim section
- Identification of the type of every segmented claim as independent or dependent
- Identification of all children (one or multiple) for every individual claim
- Identification of all parents (one or multiple) for every dependent claim
- Construction of an hierarchical tree of claims

The macro-analyzer is rule-based and uses the knowledge extracted from a 9mio wordform corpus of US and European patents[2] in the English language. The knowledge for macro-simplification is very shallow and it includes:

*Clues signaling on the start of the Claims section* such as location (the claim section of a patent comes after the description at the end of the patent document) and a list of delimiting expressions, such as *"We claim», » I claim», » claim", "what we claim is",* etc.

*Clues signaling on the start of every individual claim* that include numbering, formatting, punctuation and a list of delimiting expressions. The claims are set forth as separately numbered paragraphs in

[2] This is justified by the similarity of structures of different national patents due to the similarity of writing rules imposed by Patent Law throughout the world.

a single-sentence format. Each claim begins with a capital letter and with a number. The first claim of an issued patent is always numbered "1," with each claim thereafter following in an ascending sequence of Arabic numerals (1, 2, and 3) from broad claims to narrow claims.

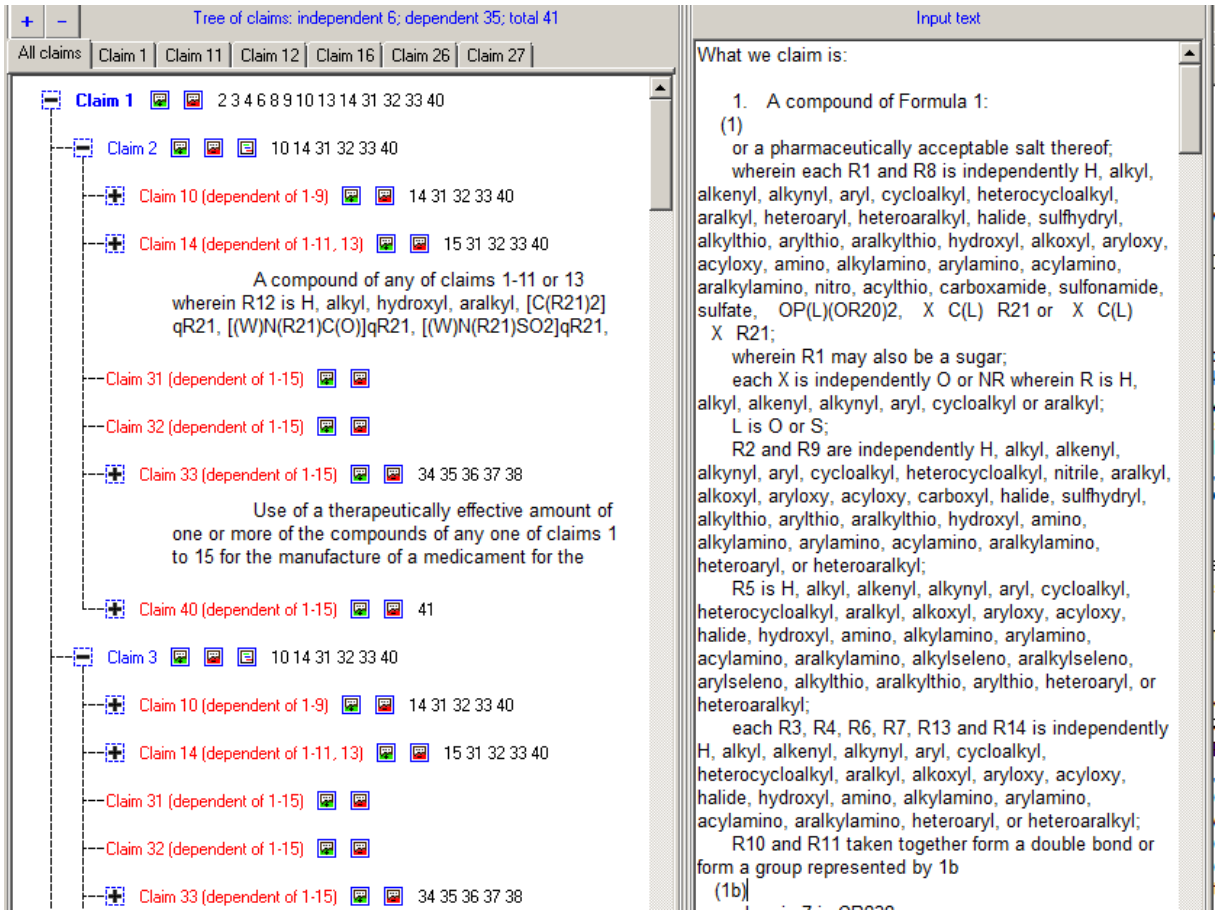


Figure.2. A screenshot of the tree of claims fragment visualized in the user interface. The number of dependent and independent claims is shown on the top. The right pane is an interactive window which displays the input patent text; this text can be scrolled and/or edited right in there. The left pane shows a tree with claims as nodes. Clicks on the coloured square buttons next to claim nodes allow displaying/hiding the claim text The numbers on the right of a claim node list claims dependent on the claim in question. The tree of claims is collapsible and expendable in different ways. The "+" and "-"are the usual "expand" and "collapse" tree buttons. The coloured square buttons on the right allow getting truncated sub-trees of the main claim tree.

*Clues signaling on the dependent claim* that include a list of reference expressions. The text of a dependent claim always starts with a number (this clue is common to all types of claims) and a specific reference expression of the type "2. The machine of Claim 1,…" . The wording of a multiply dependent claim reference expression could be, for example, "5. A gadget according to claims 3 or 4, further comprising...". Multiply dependent claims may depend on other claims which do not necessarily follow one another. For example, dependent claims can be referenced as "14. A compound of any of claims 1-9 or 13,…." There may be also reference expressions like "17. An invention as in previous claims…". Though variable, the number of dependent claim reference expressions is still limited, so that they can be rather exhaustively acquired and explicitly listed in the analyzer knowledge base.

*Clues signaling on the parents of dependent claims that* are in fact contained in the *dependent claims* reference expressions. The sets of parents of different dependent claims can be different, the same or intersect. That does not always let build a single root tree of claims for a patent. In complicated cases the macro-analysis can result in a forest of root trees of claims.

*Clues signaling on the children of the claims* that do not need to be acquired, the analyzer calculates them from the dependent claims reference expressions.

The type of knowledge required for macro-analysis of the claim section and high structural similarity of national patents imposed by Patent law make the analysis algorithm practically language-independent. The only thing which is required to port the macro-analyzer from English into any other language is to change the lexicon of reference expressions. Such lexicons should certainly be acquired for every particular language by corpus analysis, which is pretty straight forward.

The macro-analyzer is implemented as a module of an end-user tool that visualizes the results of macro-analysis in the form of a tree structure as shown in Figure 2. The visualized tree is highlighted in a way that facilitates the understanding of multiple claim interrelations and allows grasping a lot of claim-related information "at a glance" thus improving the readability of the claim section. The independent claims in the tree nodes are highlighted in blue, while dependent claims are presented in red. Lists of children are displayed in black to the right of their parent claim nodes, the parents of a multiply dependent claims are shown in red on the left of multiply-dependent claim nodes. The nodes corresponding to multiply-dependent claims are highlighted in red. The interface program does supplementary math and displays a total number of claims, as well as the number of independent and dependent claims, correspondingly, and displays them in the status bar. The independent claims are bookmarked.

The user can navigate the claim tree, which can collapse/expand in different combinations to display the subtrees of independent claims, claim children, parents, or ascenders. There are special buttons next to each claim node that allow to partially or fully display claim texts. The input text of a whole patent is displayed on the right interactive pane of the interface. These functionalities allow interactively aligning claims with certain parts of the description for consistency check or editing. The macro-analyzer for the English language is currently available as a standalone tool.

## 5 Micro-level claim simplification

### 5.1 The knowledge

Micro-level simplification at each of its stages is done by means of a specific combination of rule-based and statistical techniques and relies on linguistic knowledge of different depth. This knowledge is structured following the methodology described in (Sheremetyeva, 1999; Sheremetyeva, 2003) and is mostly coded in the system lexicon as well as in analysis and generation rules. Different modules of the micro-level simplification component use specific parts and types of linguistic knowledge included in the lexicon and their own specific sets of rules.

The word list for the lexicon was automatically acquired from a 9 million-word corpus of a US and European patents available to us from our previous projects and patent web sites. A semi-automatic supertagging procedure was used to label these lexemes with their supertags. A supertag codes morphological information (such as POS and inflection type) and semantic information, an ontological concept, defining a word membership in a certain semantic class (such as object, process, substance, etc.). For example, the supertag Nf shows that a word is a noun in singular (N), means a process (f), and does not end in –ing. This supertag will be assigned, for example, to such words as `activation` or `alignment`. At present we use 23 supertags that are combinations of 1 to 4 features out of a set of 19 semantic, morphological and syntactic features for 14 parts of speech. For example, the feature structure of noun supertags is as follows: Tag [ POS[Noun [object [plural, singular] process [-ing, other[plural, singular]] substance [plural, singular] other [plural, singular]]]]].

The "depth" of supertags is specific for every part of speech and codes only that amount of the knowledge that is believed to be sufficient for our analysis procedure. The units of the system lexicon are described with a different level of depth. A deep (information-rich) description is only assigned to predicates. Other types of lexemes are only assigned morphological information.

Predicates in our system are words, which are used to describe interrelations between the elements of the invention. They are mainly verbs, but can also be adjectives or prepositions. A predicate entry covers both the lexical, and, crucially for our system, the syntactic and semantic knowledge. The morphological knowledge includes partial paradigms of explicitly listed predicate wordforms as found in the patent corpora. Syntactic and semantic knowledge relevant for our task is included in the CASE_ROLEs and PATTERNs fields of predicate entries. The CASE_ROLEs field lists a set of the

corpus-based predicate case-roles such as agent, theme, place, instrument, etc. The PATTERNs code domain-based information on the most frequent co-occurrences of predicates with their case-roles, as well as their linear order in the claim text. For example, the pattern (1 x 3 x 2) corresponds to such clam fragment as `1:boards x:are 3:rotatably x:mounted 2:on the pillars.`

The processing algorithms and rules for every stage of micro-simplification will be described in the corresponding sections below.

### 5.2 Terminology visualization

The readability of patent claims increases if the reader can spot the terminology at a glance. It is important not only in the process of claim examination for novelty but also for a quick check of whether the claim text complies the writing rules prescribed by the Patent law. Claims have to be supported by the patent description, which means that any terms used in the claims must be found in the description. To facilitate these tasks we simplify the claim text by automatically highlighting its nominal terms with the subsequent highlighting of these terms in the patent description. In case a certain claim term is not found in the description a warning message is given. This task is performed based on the results of a shallow analysis performed by a hybrid NP extractor and NP and predicate term chunkers which in succession run on the same claim text.

To extract (and then highlight) nominal terminology we use the NP extractor described in (Sheremetyeva, 2009). The extraction methodology combines statistical techniques, heuristics and a very shallow linguistic knowledge extracted from the main system lexicon (see Section 5.1). The NP extractor knowledge base consists of a number of unilingual lexicons, - sort of extended lists of stop words forbidden in particular (first, middle or last) positions in a typed lexical unit (NP in our case). These lists of stopwords are automatically extracted from the morphological zones of the entries of relevant parts-of-speech.

The NP extraction procedure starts with n-gram calculation and then removes those n-grams that cannot be NPs from the list of all calculated n-grams. This is done by successive matching the components of calculated n-grams against the stop lexicons. The NP extraction itself thus neither requires such demanding NLP procedures, as tagging, morphological normalization, POS pattern match, etc., nor does it rely on statistical counts (statistical counts are only used to sort out keywords which is not needed in our case). The advantages of this extractor are in that it does not rely on a preconstructed corpus, works well on small texts, does not miss low frequency units and can reliably extract *all* NPs from an input text. The noun phrases thus extracted are of 1 to 4 components due to the limitations of the extractor that uses a 4-gram model. A small adaptation of the extractor has been made to have it better suite the current task. First, we excluded a lemmatizer from the original extraction algorithm and kept all extracted NPs in their textual forms and, second, we updated the tool knowledge so as to allow NPs being extracted form a claim text with articles and determiners (“said”, this”, etc;) if present. It was done to avoid the ambiguity in the subsequent NP chunking in the claim text.

The chunker users the knowledge dynamically produced by the extractor (lists of all NPs with determiners in their text form as found in the claim text in question). The NPs are chunked in the claim text by matching the extractor output against the claim text. The predicate terminology is chunked by the main lexicon predicate entries look-up practically without (ambiguity) problems. The chucked nominal and predicate terminology is visualized in the user interface by highlighting them in the claim text (see Figure 3, left pane). The same dynamic knowledge is used to check for the claim noun and predicate terminology in the text of the description. In case of a failure a warning message about inconsistency is displayed.

### 5.3 One-sentence-to-many decomposition

Decomposition of one syntactically complex claim sentence into a set of simple sentences is done in two takes. First the claim is segmented into the preamble, transition and body text, and then the preamble and claim body are further segmented into simple sentences.

The first segmentation is pretty straight forward and is performed based on the knowledge about transition expressions explicitly listed in the system knowledge base. The list of corpus-based transition expressions covers both the US and European rules for writing claims. In the US claims the transitions basically used are: "comprising", "which comprises," "consisting of," and "consisting essen-

tially of." Modern claims follow a format whereby the preamble is separated from the transitional term by a comma, while the transitional term is separated from the body by a colon.

Under the European Patent Convention a claim can be written according to the so-called "two-part form" where the claim text is divided into a generic part that contains old knowledge and a difference part that contains novel features of the invention. The delimiting expressions are "characterized in that" or "characterized by". If the European format is used, what is called the "preamble" is different from the meaning of «preamble" under the U.S. patent law. In an independent claim in Europe, the preamble is everything which precedes the delimiting expression. The preamble in Europe is sometimes also called "pre-characterizing portion". It can contain a text of a certain length and syntactic complexity. The preamble can therefore require decomposition (simplification) as well.

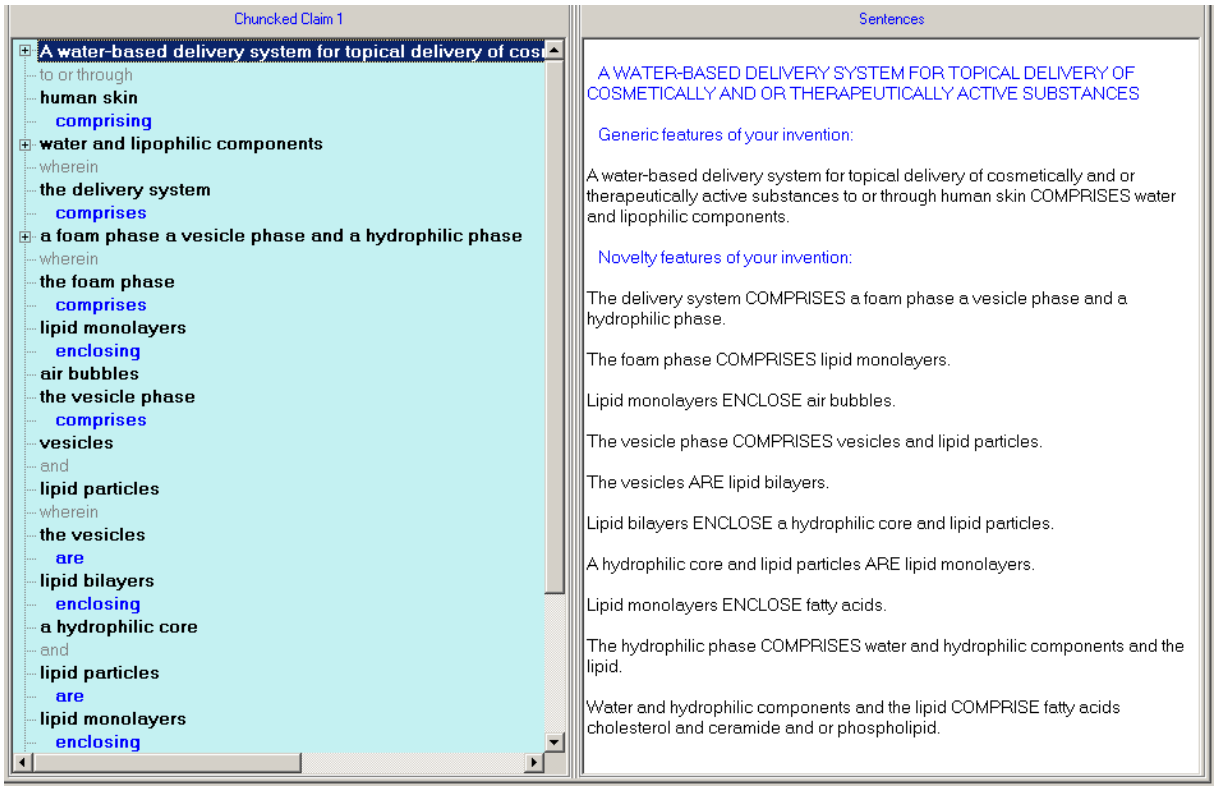


Figure 3. A screenshot of "Decomposition" page of the user interface. The left pane shows the input claim text with highlighted terminology. Predicates are in blue, the nominal terminology is boldfaced. The right pane visualizes a simplified claim text in the form of simple sentences. The content of the texts in both panes is the same.

Decomposition of the generic/preamble and difference/body parts of the claim text demands much more sophisticated techniques than those used at previous levels of simplification. It is performed by the deep analyzer that in full uses the knowledge of the lexicon described in Section 5.1.

The deep analyzer includes a disambiguating supertagger, typed phrase chunker based on PSG rules and DPG-based predicate/argument dependency identifier. It superficially performs the NLP analysis procedure as described in (Sheremetyeva 2003). However, the original procedure of the NLP claim analysis presented in the cited paper was significantly modified and simplified by introducing the shallow analyzer (see section 5.2) at the pre-deep-NLP analysis stage. This made the analysis procedure more robust and less computationally demanding.

The workflow of the current analyzing procedure is as follows. A raw claim is first pre-processed by the shallow analyzer that extracts and chunks claim nominal phrases and predicates as presented in Section 5.2.

The claim, thus partially parsed and tagged is then input into the preexisting deep analyzer, which completes super tagging, recursive chunking and defines predicate/argument dependencies. The output

of the analyzer is a shallow interlingual representation where the content of every nascent simple sentence is represented by a separate predicate/argument structure (proposition) in the form

*proposition::={label predicate-class predicate ((case-role)(case-role))*}*
*case-role::= (rank status value)*
*value::= phrase{(phrase(word supertag)*)}**

The final parse, a set of fully tagged predicate/argument structures, is then submitted into the generator that transforms every predicate/argument structure into a simple sentence. The generator determines the order of sentences, the order of words in the nascent sentences taking care of morphological forms and agreement. The order of the sentences follows the order of predicates in the claim. The order of the words in a sentence is defined by the knowledge in the PATTERNs zones of the predicate entries of the lexicon. Morphological synthesis and agreement are rule-based. The generic part and novelty parts of the claim are generated separately. The micro-level of simplification is illustrated in Figure 3.

### 5.4 Text-to-diagram simplification

Simplification of a claim text into a diagram is performed based of the internal claim representation as shown in Section 5.3. We here used the automatic text planner of the claim generator that was developed as a module of a patent MT system (Sheremetyeva, 2007).

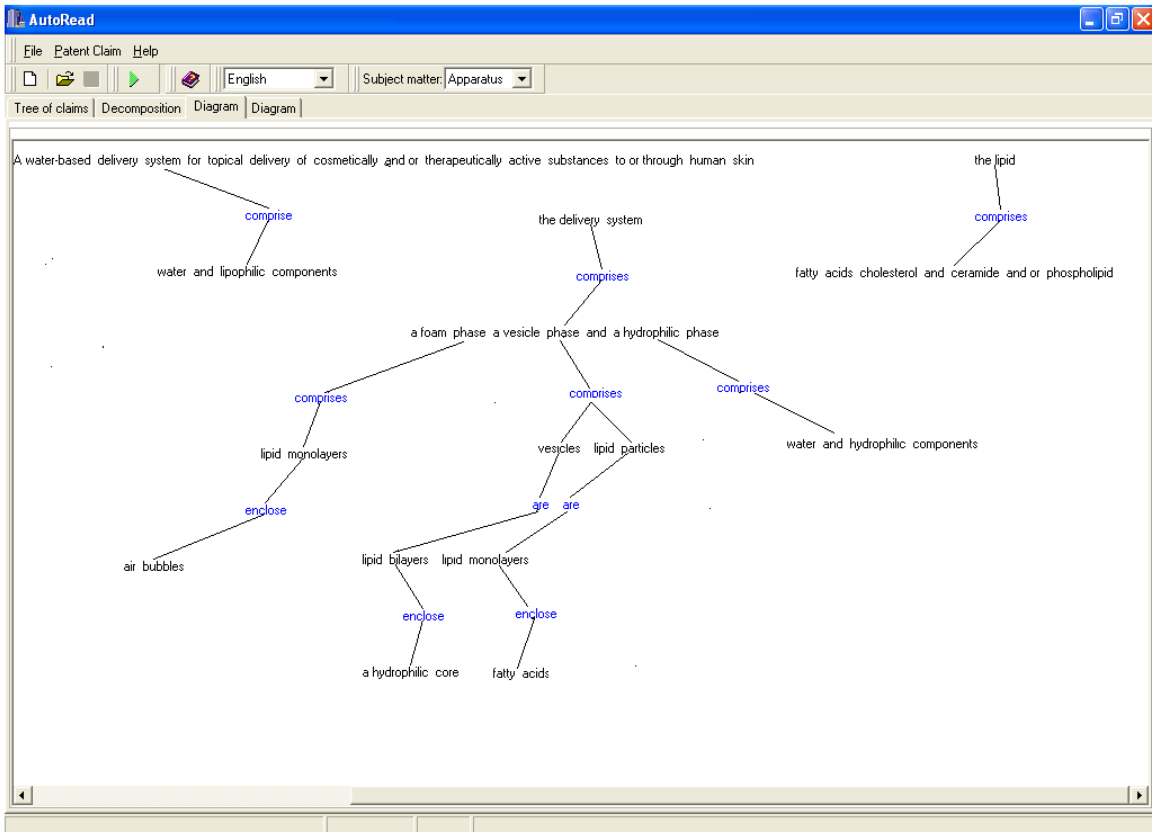


Figure 4. This screenshot of the "Diagram" page of the user interface which displays a conceptual schema of the invention underlying the claim text.

The planner runs over the output of the deep analyzer in the form of a set of separate predicate/argument structures and unifies separate predicate-argument structures into a hierarchical structure in the form of a single root tree or a forest of trees. The planning stage is guided by the constraints on the patent claim sublanguage. The unified trees of predicate structures are visualized for the reader in the form of a diagram with explicitly listed invention elements and their relations as in Figure 4.

## 6 Evaluation

Given that no reliable evaluation metrics exist so far for text simplification we performed a preliminary qualitative evaluation of our methodology based on human judgment (as in all cited works on claim simplification). Some of the researchers admit avoiding qualitative evaluation due to the lack of resources that would have made it possible (Mille and Wanner, 2008). The number of patents the authors use to evaluate their methodologies might seem quite limited, e.g., (Mille and Wanner, 2008) report evaluation results based on 30 patents; in (Bouayad-Agha et al.) the test corpus consisted of 29 patents; (Ferraro et al. 2014) inspected 38 patent documents, but again, the reason is the immense complexity and length of the patent claims.

There is no need to use readability formulas to prove the higher comprehensibility of the output of our macro- and micro level simplifiers as compared to the original claim section texts. These formulas are not applicable to the macro-level simplification. As for the micro-level simplification, the terminology of the original and simplified claims is kept unchanged and it is evident that simple and short sentences are "simpler" than long and complex ones.

We evaluate our methodology with a view to preserving the claim content and grammaticality as bad syntax can change the content of the claim with all the legal consequences. We asked human annotators (5 linguist students and 3 patent experts) to grade the simplification results according to these two criteria. The architecture of our system allows evaluating each component independently.

*The quality evaluation method of nominal and predicate terminology* extraction/highlighting consisted in comparing our results with a gold reference list. The gold lists of multi-component nominal terms and predicate terms were built manually by linguist students from the patent corpus of 72000 words for which it was feasible to create a gold standard. The number of multi-component NPs does not include the number of those NPs that only appear inside longer nominal phrases. The evaluation results of the extraction are in Table 1.

Table 1. Results of the extraction of nominal and predicate terminology

| | Multicomponent NPs | Predicates |
|---|---|---|
| Total number of gold terms | 1425 | 1272 |
| Total extracted phrases | 1476 | 1186 |
| Correct terms | 1394 | 1154 |
| Missed terms | 67 | 54 |
| Incorrect phrases | 24 | - |

Most of the missed NPs are longer than 4 words; they are missed because we limited ourselves to a 4-gram extraction model. The problem can be fixed by widening the extraction window which might increase the computation time. As for predicates, no incorrect terms were extracted because they were only searched against the predicate entries in the system lexicon in the "residue" of the claim text after NP extraction. Extraction mistakes can be corrected by updating the knowledge of the NP extractor.

*The macro-level simplification (construction of the hierarchical trees of claims)* was tested on 25 patents (each having from 7 to 98 claims of different kind). The performance at this level of simplification was practically perfect (i.e., for detecting the beginning and end of the claim section in a patent, the accuracy percentage is 100 and the trees of claims for every patent were also 100% correct. The result is explained by that the very shallow and closed knowledge required for this simplification procedure was completely covered in the lexicon.

*Decomposition of a long claim sentence* is undergoing extensive testing, further extension and knowledge update. It was feasible to test the methodology on the material of the first (most representative) claims of 25 patents containing from 5 to 10 predicates (meaning that claims should be decomposed into from 5 to 10 simple sentences, correspondingly). The total number of the resulting simple sentences is 147 out of which 93 sentences were correct. The problems are mainly due to the insufficient coverage of the rules identifying predicate/argument relations of syntactic chunks as output by the deep parser. However, these problems can be solved by the knowledge extension and brush-up. Already in their present state this simplifying component shows promising performance.

*Building diagrams* is performed by the planning component of a fully operational generator (see section 5.3). It is completely conditioned by the parser and correlates with the claim decomposition. Once the decomposition into simple sentences is correct, the diagram is correct as well.

## 7 Conclusions

In this paper, we have presented a methodology for the simplification of both the whole section of patent claims and individual claims. The simplification improves the readability of patent clams by the following: building a hierarchy of multiple claims with relevant accompanying information; highlighting the claim/patent nominal and predicate terminology; decomposing long and complex sentences of individual claims into a set of simple sentences preserving the content of the claim; building claim diagrams graphically visualizing interrelations of the invention elements.

Based on the methodology an experimental claim simplification tool was developed. As of today the programming shell of the tool is completed and provides for knowledge administration in all modules of the system to improve their performance. The static knowledge sources have been compiled for the domain of patents about apparatuses and chemical substances. The morphological analysis of English is fully operational and well tested. The English generator is also operational. The evaluation results suggest that our system produce much more readable output when compared to the original claims, and that the preservation of the claim content and grammaticality are positively rated by the annotators. The tool is currently undergoing an extensive extension and evaluation. However, already in it present state it provides for promising performance. The research is primarily targeted to patent experts, but can also be useful for laypeople and for automatic patent processing.

## References


Aluisio S., Specia L., Gasperin C. and Scarton C. 2010. Readability assessment for text simplification. *Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications*, pp.1–9.

Bott S., Saggion H. and Figueroa D. 2012. A Hybrid System for Spanish Text Simplification. *NAACL-HLT 2012 Workshop on Speech and Language Processing for Assistive Technologies (SLPAT),* pages 75–84, Montreal, Canada, June 7–8, 2012. c 2012 Association for Computational Linguistics

Bouayad-Agha N., Casamayor G.,Ferraro G., and Wanner L. 2009 Simplification of Patent ClaimSentences for Their Paraphrasing and Summarization. *Proceedings of the Twenty-Second International FLAIRS Conference.*

Brown, J. D. (1998). An EFL readability index. *JALT Journal*, 20, 7–36.

Crossley, S. A. & McNamara, D. S. 2008. Assessing Second Language Reading Texts at the Intermediate Level: An approximate replication of Crossley, Louwerse, McCarthy, and McNamara. *Language Teaching*, 41 (3), 409–229.

Crossley, S. A., Allen D. B. and McNamara D. S. 2011. Text readability and intuitive simplification: A comparison of readability formulas. *Reading in a Foreign Language*. April 2011, V. 23, No. 1. pp. 84–101

Dell'Orletta, F., Montemagni S., and Venturi G. 2011. READ–IT: Assessing Readability of Italian Texts with a View to Text Simplification. *Proceedings of the 2nd Workshop on Speech and Language Processing for Assistive Technologies,* Edinburgh, Scotland, UK, 2011, pp. 73-83.

Ferraro G., Suominen H., and Nualart J. 2014.Segmentation of patent claims for improving their readability *Proceedings of the 3rd Worshop on Predicting and Improving Text Readability for Target Reader Populations (PITR) @ EACL 2014*, pp. 66–73. Gothenburg, Sweden, April 26-30 2014. c 2014 Association for Computational Linguistics

Graesser, A. C., McNamara, D. D., Louwerse, M. L., and Cai, Z. 2004. Coh-Metrix: Analysis of text on cohesion and language. *Behavior Research Methods, Instruments, & Computers*, 36, 193–202.

Greenfield, J. (2004). Readability formulas for EFL. *JALT Journal*, 26, 5–24.

Mille S. and Wanner L. Multilingual Summarization in Practice: The Case of Patent Claims *12th EAMT conference,* 22-23 September 2008, Hamburg, Germany

Kincaid, J. P., Fishburne, R. P., Rogers, R. L. & Chissom, B. S. 1975. Derivation of new readability formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy enlisted personnel, *Research Branch Report 8–75, Millington, TN: Naval Technical Training,* U. S. Naval Air Station, Memphis,.

Petersen, S., and Ostendorf, M. 2007 Text simplification for language learners: a corpus analysis. *Proceedings of Workshop on Speech and Language Technology for Education*

Poornima C , Dhanalakshmi V, Anand Kumar M, and Soman K. P. 2011. Rule based Sentence Simplification for English to Tamil Machine Translation System. *International Journal of Computer Applications* (0975 – 8887) Volume 25– No.8, July 2011.

Pressman D. 2006. *Patent It Yourself*. Nolo, Berkeley, CA.

Rada D. V. 1995. Reading and understanding patent claims. *JOM*, 47(11):69–69.

Siddharthan, A. 2002. An Architecture for a Text Simplification System. *Proceedings of the Language Engineering Conference (LEC'02),* Hyderabad, India, IEEE Computer Society pp. 64.

Sheremetyeva S. 1999. A Flexible Approach To Multi-Lingual Knowledge Acquisition For NLG.. *Proceedings of the 7th European Workshop on Natural Language Generation.* Toulouse. (France) May 13-15.

Sheremetyeva S. 2003. Natural language analysis of patent claims. *Proceedings of the ACL 2003 Workshop on Patent Processing, ACL '03*, Stroudsburg, PA, USA. Association for Computational Linguistics.

Sheremetyeva S. 2007. On Portability of Resources for Quick Ramp-Up of Multilingual MT for Patent Claims. *Proceedings of the workshop on Patent Translation in conjunction with MT Summit XI*, Copenhagen, Denmark, September 10-14

Sheremetyeva S. 2009. On Extracting Multiword NP Terminology for MT. *Proceedings of the Thirteenth Conference of European Association of Machine Translation (EAMT-2009)*. Barcelona, Spain. May 14-15.

Shinmori, A., Okumura M., Marukawa Y., and Iwayama M. (2003). Patent claim processing for readability: structure analysis and term explanation. *Proceedings of the ACL-2003 Workshop on Patent Corpus Processing, volume 20 of PATENT 03*, pp. 56–65, Stroudsburg, PA, USA. Association for Computational Linguistics.

Shinmori, A., Okumura M., Marukawa Y. 2012. Aligning patent claims with the"detailed description" for readability. *Journal of Natural Language Processing*, 12(3):111–128.

Takao D. and Sumita E. 2003. "Input sentence splitting and translation", *Proceedings of the Workshop on Building and using parallel Texts, HLT-NAACL 2003*.

Zhu, Z., Bernhard, D. and Gurevych, I. A. 2010. Monolingual Tree-based Translation Model for Sentence Simplification. *Proceedings of The 23rd International Conference on Computational Linguistics (COLING)*, August 2010. Beijing, Chin.a