Embedding MT for Generating Patent Claims in English from a Multilingual Interface

Svetlana SHEREMETYEVA
LanA Consulting ApS
Mynstersvej 7A, 2
Copenhagen, Denmark, DK-1827
lanaconsult@mail.dk

Abstract

In this paper, we present a methodology for the development of interactive domain-tuned patent tools for generating patent claims in English from non-English interfaces. The methodology is based on a merger of an interactive English-to-English patent claim generator, AutoPat[1] and any external MT engine that might be appropriate for a certain language. The translation procedure is reduced to translation words and phrases rather than a complex claim sentence. The approach has been successfully used in The J-E patent system[2], a patent claim generator in English from a Japanese-only interface, and in Dan-Pat[3], a similar tool for the Danish-English pair of languages. The two systems use different MT engines but feature similar overall architecture. The methodology is portable to other languages and MT engines.

1 Introduction

Translating patents is an important task for international trade and industry. In a patent text (or "disclosure", using official terminology), the crucial part is the patent claim, which is the actual subject of legal protection.

Translation of patent claims is a difficult and time-consuming task even for experts. There is currently no MT system that could produce high quality translations of patents, let alone patent claims, as grammar and terminology in patent claims are highly specific and need to be taken into consideration (Bourbeau and Kittredge, 1988).

Patent claims are characterized by overwhelming sentence depth, length, and the abundance of technical terms. Claims must be formulated as specified by the German Patent Office at the turn of the previous century and commonly accepted in the U.S., Russia, Denmark and other countries. The requirements are similar in Japan. The claim must describe essential features of invention in the obligatory form of a single extended nominal sentence with a well-specified conceptual, syntactic and stylistic structure. Few researchers, however, focus on the linguistic specificities of patent style (vs. technical style) (Shnimory et al., 2002; Gnasa and Woch, 2002; Fujii and Ishikawa, 2002).

Most of the research in the patent domain is devoted to information retrieval (e.g., Fujii and Ichikawa, cf; Chen et al., 2003). One of the few patent-specific research in MT has been done by (Sheremetyeva and Nirenburg, 1999).

Our approach aims to bypass major problems in MT caused by complex syntax of input text. It is based on the principles established in (Sheremetyeva and Nirenburg, cf) for the hybrid Russian-English MT system for patents and draws heavily on patent claim language restrictions.

The methodology involves a merger of an interactive English-to-English patent claim generator, AutoPat, and an external MT engine appropriate for a certain language. (Prieto-Diaz, 1993) and (Thomas and Nejmeh, 1992) provide for guidelines for reuse strategies and integration during software development process.

An MT system is adjusted to have the AutoPat's data format, and linked to AutoPat by means a Dynamic Link Library (DLL). The advantage of our methodology is that the translation procedure is reduced to translation of words and phrases rather than a complex claim sentence.

The approach has been used in the J-E patent system, a patent claim generator in English from a Japanese-only interface, and in Dan-Pat, a Danish-English generation tool. The two systems use different MT engines (PC-Transfer Honyaku-Studio[4] and

[1] *AutoPat,* LanA Consulting, Denmark, Copenhagen.

[2] *The J-E patent system* ,Cross Language KK, Tokyo, Japan and LanA Consulting, Denmark, Copenhagen.

[3] *Dan-Pat*, LanA Consulting, Denmark, Copenhagen.

[4] PC-Transer Honyaku-Studio, Cross Language KK, Tokyo, Japan.,-an English/Japanese MT system for patent claims.

APTrans[5], respectively) but have a similar overall architecture (Figure 1). The methodology is portable to other languages and MT engines.

In what follows we first give an overview of a generic generation system in English from a non-English interface with embedded MT. We then concentrate on the way the parent AutoPat and MT applications are reused in a hybrid multilingual application. We conclude with implementation issues. We mainly illustrate our approach on the example of Dan-Pat which includes AutoPat and embedded APTrans MT software. Details of The J-E patent system, combining AutoPat and PC-Transfer MT engine can be found in (Neumann, 2005).

2 Tool Overview

An overall architecture of the hybrid tool for generating Patent claims in English from a non-English interface is given in Figure 1.

The tool includes modules for SL knowledge elicitation reusing AutoPat knowledge elicitation procedure[6] and data structure, an MT engine, a Dynamic Link Library (DLL), which exports some of the AutoPat interface functions, and Generator.

DLL converts predicate structures with “bare” English strings output by an embedded MT system into the format the AutoPat Generator “understands”. In fact, DLL includes that part of the AutoPat interface which analyzers human input before the generation, - it tags and disambiguates tags against the AutoPat tagging lexicon, recognizers the same nouns (both in plural and singular) and marks them as being coreferential if necessary, and allows the user to add/delete new predicates to the predicate dictionary, automatically assigning default entries. DLL also performs automatic English grammar and content check. The modularity of the tool provides for speeding-up the development process by reusing many components from parent applications.

3 Reusing AutoPat and MT engines

The core element is *AutoPat,* - an English patent generation system for English users. The current version of AutoPat is a significally updated and extended version of the English-to-English system for generating patent claims on apparatuses as described in (Sheremetyeva, cf). It now covers two invention subject matters, - apparatuses and methods for 3 technical domains: machines, information technology and electronics and is equipped with spelling, grammar and content checkers. Two innovative features achieve this:

- The user rebuilds the structure of an invention by grouping the parts of the invention in a hierarchical tree.
- Information about processes and functions is input by filling semantic case slots of pre-defined predicate-templates with the names of the invention parts during a computer interview.

The SL (non-English) interfaces look basically like a localized version of the AutoPat English interface gathering all necessary information about the invention in a complex, refined interface modeled along the claim structure. Figure 3 shows a screenshot of the Dan-Pat system, where the knowledge is supplied in the Danish language.

In the course of the knowledge elicitation procedure the tool builds a SL content representation which is converted into an English content representation, the SL words and phrases input by the user, are translated by a certain embedded MT engine. MT techniques of such engines may be different. For example, we use the in-house built APTrans system for the Danish-English translation and, Cross-Language[7] uses the PC-Transfer Honyaku-Studio system for the Japanese-English translation.

As immediate feedback, the tool creates and displays one short sentence both in SL and in English, so that the user can check whether the system has correctly “understood” him. At the same time the English content representation is input into the AutoPat generator, which produces the final claim text in English.

3.1 Multilingual Representation of Patent Sublanguage Knowledge

A successful writer of patent claims needs two distinct types of expert knowledge: knowledge of patent claims as legal documents and knowledge about the invention technical field.

The technical knowledge is mainly conveyed by domain-tuned terminology. The legal knowledge essentially manifests itself in the constraints and preferences concerning claim syntax. The claim must describe essential features of the invention in the obligatory form of a single extended nominal sentence with a well specified conceptual, syntactic and stylistic/rhetorical structure which frequently includes long predicate phrases.

[5] *APTrans*, LanA Consulting, Denmark, Copenhagen, - an MT system for translating claims between English and other European lanuages, currently under development (Sheremetyeva, 2003a).

[6] See detailed screenshots of the AutoPat knowledge elicitation procedure and generation at www.lanaconsult.com

[7] Cross Language KK, Tokyo, Japan

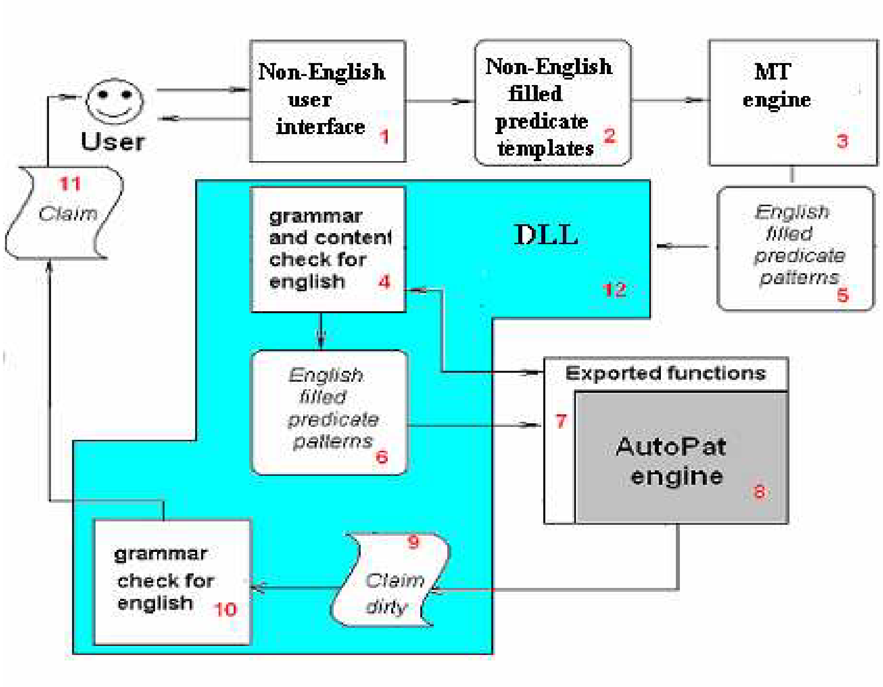

User
Non-English user interface 1
Non-English filled predicate templates 2
MT engine 3
11 Claim
grammar and content check for english 4
DLL
12
English filled predicate patterns 5
English filled predicate patterns 6
Exported functions
7
AutoPat engine 8
9 Claim dirty
grammar check for english 10

Figure 1. An overall architecture of the tool for generating patent claims in English from a non-English interface.

These requirements apply to the description of all types of inventions (devices, substances, methods, etc.). Figure 2 illustrates a fragment of a US main claim for methods. A claim for apparatuses is shown in Figure 4.

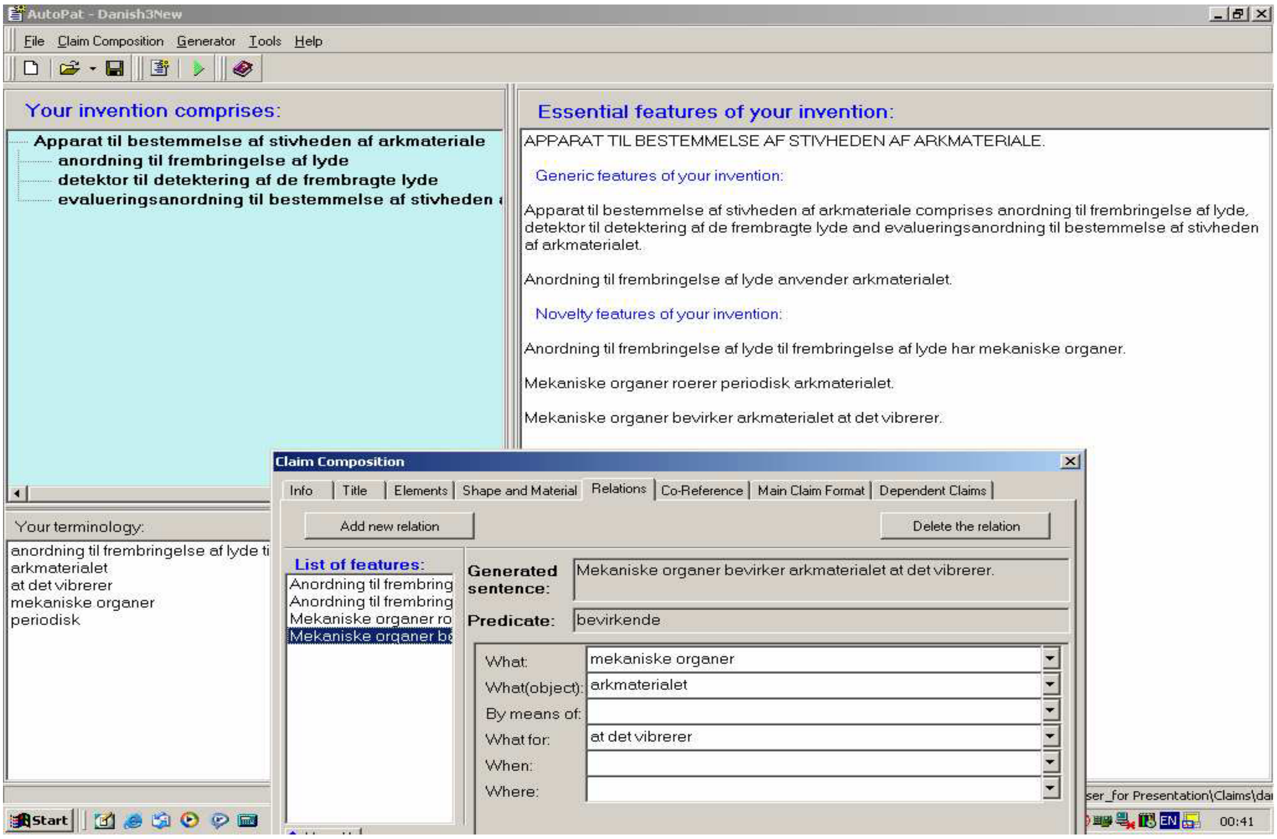
> *A method of* ***forming*** *synthetic hydrogen defect free diamond film on a substrate surface* ***having*** *an initial coating* ***selected*** *from the group consisting of fullerene and diamond, comprising the steps of:*
>
> *(a)* ***supplying*** *a source of fullerenes with each fullerene molecule* ***consisting of*** *carbon-carbon bonds;*
>
> *(c)* ***contacting*** *said energized fullerene ions* ***consisting of*** *carbon-carbon bonds with said coating substrate surface...*

Figure 2. A fragment of a patent claim text for methods. Predicative words which are heads of individual phrases describing essential features of the invention are bold faced.

The major difference between these claim texts is that in apparatuses the invention components are physical objects-elements (*devices, rotating shaft*, etc.) that are described with nominal terminology, while in methods the invention components are processes, - method steps, which require predicate templates. The description of every method step can further include descriptions of element relations as in apparatuses. It is also characteristic of a "method" claim that the title always realizes a template of the predicate meaning "process". To deal with these specifities different AutoPat interfaces for apparatuses and methods were implemented, each tuned to its own part of static knowledge (lexicon, disambiguation rules, and analysis and generation algorithms). Both kinds of knowledge for the English language are hard-coded in the AutoPat English lexicons.

The essential part of the system knowledge is a *deep multilingual lexicon of predicates.* Predicates are lexical units that describe properties or relations between invention elements (bold-faced in Figure 2). For correct translation it is crucial that every SL lexicon of predicates had the same data format as the English lexicon. The English predicate lexicon is thus used as a seed lexicon in any of multilingual tools. The SL predicate lexicons are built as translation equivalents of the original English predicates and have the AutoPat data format. A predicate entry includes a predicate/argument template, defining a semantic class ("connection", "location", etc.), a set of case-roles ("agent", "place", "mode", etc.) and linear patterns coding possible word orders in a predicate phrase (Sheremetyeva, 2005).

Translation of case-role fillers is based on the *SL-English technical dictionaries,* inherent part of embedded parent MT systems. For example, PC-Transfer Honyaku-Studio, the Japanese to English MT software, has more than 2 million entries providing exact terminology and translation.



Your invention comprises:

- Apparat til bestemmelse af stivheden af arkmateriale
  - anordning til frembringelse af lyde
  - detektor til detektering af de frembragte lyde
  - evalueringsanordning til bestemmelse af stivheden a

Essential features of your invention:

APPARAT TIL BESTEMMELSE AF STIVHEDEN AF ARKMATERIALE.

Generic features of your invention:

Apparat til bestemmelse af stivheden af arkmateriale comprises anordning til frembringelse af lyde, detektor til detektering af de frembragte lyde and evalueringsanordning til bestemmelse af stivheden af arkmaterialet.

Anordning til frembringelse af lyde anvender arkmaterialet.

Novelty features of your invention:

Anordning til frembringelse af lyde til frembringelse af lyde har mekaniske organer.

Mekaniske organer roerer periodisk arkmaterialet.

Mekaniske organer bevirker arkmaterialet at det vibrerer.

Your terminology:

anordning til frembringelse af lyde ti
arkmaterialet
at det vibrerer
mekaniske organer
periodisk

Claim Composition

Info | Title | Elements | Shape and Material | Relations | Co-Reference | Main Claim Format | Dependent Claims
Add new relation
Delete the relation

List of features:
Anordning til frembring
Anordning til frembring
Mekaniske organer ro
Mekaniske organer be

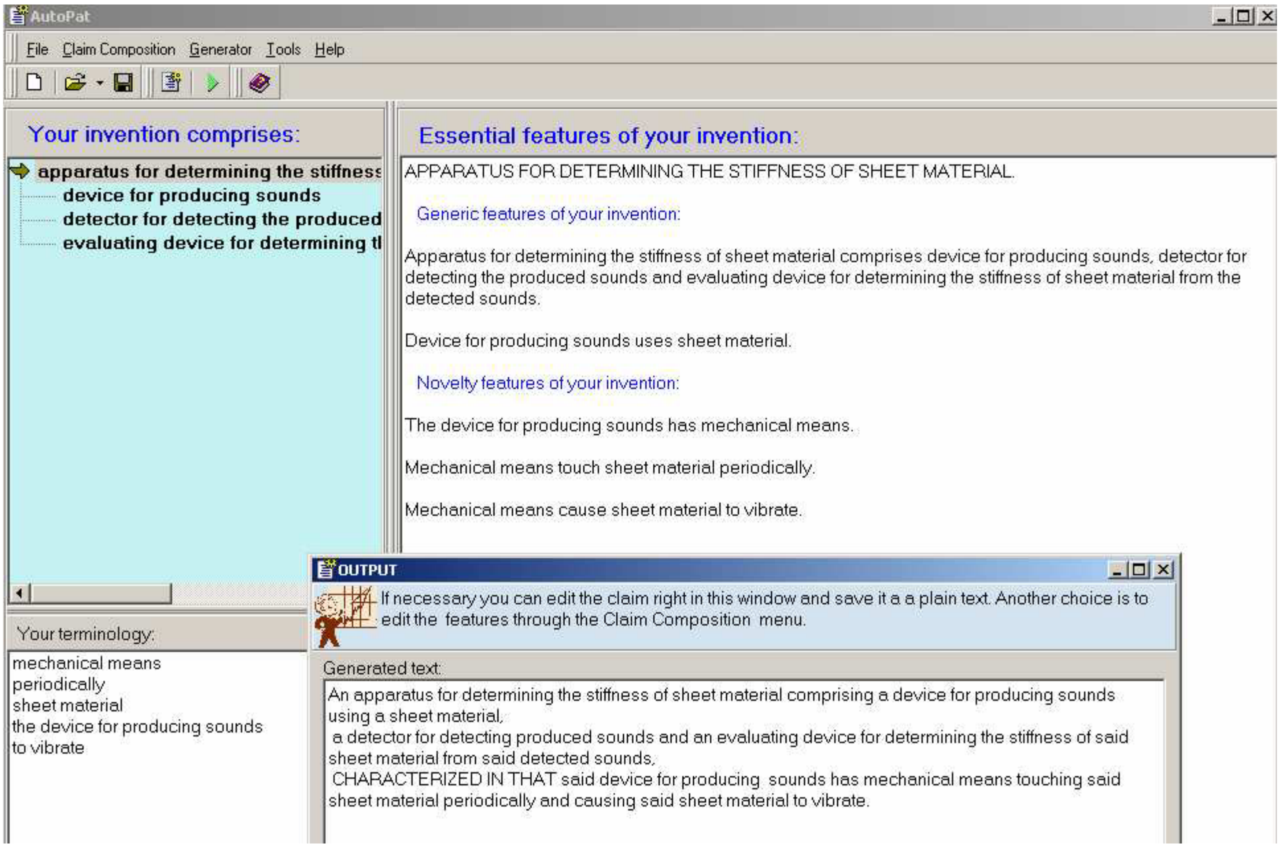
Generated sentence: Mekaniske organer bevirker arkmaterialet at det vibrerer.

Predicate: bevirkende

What: mekaniske organer
What(object): arkmaterialet
By means of:
What for: at det vibrerer
When:
Where:

ser_for Presentation\Claims\da

Figure 3. A screenshot of the Danish interface at the final stage of knowledge elicitation.



Your invention comprises:

- apparatus for determining the stiffness
  - device for producing sounds
  - detector for detecting the produced
  - evaluating device for determining tl

Essential features of your invention:

APPARATUS FOR DETERMINING THE STIFFNESS OF SHEET MATERIAL.

Generic features of your invention:

Apparatus for determining the stiffness of sheet material comprises device for producing sounds, detector for detecting the produced sounds and evaluating device for determining the stiffness of sheet material from the detected sounds.

Device for producing sounds uses sheet material.

Novelty features of your invention:

The device for producing sounds has mechanical means.

Mechanical means touch sheet material periodically.

Mechanical means cause sheet material to vibrate.

Your terminology:

mechanical means
periodically
sheet material
the device for producing sounds
to vibrate

OUTPUT

If necessary you can edit the claim right in this window and save it a a plain text. Another choice is to edit the features through the Claim Composition menu.

Generated text:

An apparatus for determining the stiffness of sheet material comprising a device for producing sounds using a sheet material,
a detector for detecting produced sounds and an evaluating device for determining the stiffness of said sheet material from said detected sounds,
CHARACTERIZED IN THAT said device for producing sounds has mechanical means touching said sheet material periodically and causing said sheet material to vibrate.

Figure 4. A screenshot of the last page of the tool at the final stage of patent claim generation in English from the Danish input shown in Figure 4.

To provide for the correct input required by the English Generator the interlingual tool reuses the AutoPat *English tagging lexicon* (see the section Workflow*)*. It has more than 200.000 entries where words are listed with such features as POS, number, inflection type and semantic class (physical object, substance, etc.) coded in tags. At present we use 43 tags that are combinations of 1 to 4 features out of a set of semantic, morphological and syntactic features for 14 parts of speech.

The claim content representation language is a shallow interlingua composed of predicate structures as shown in Figure 5, where *predicate-class* is as specified in the predicate lexicon, *predicate* is a string corresponding to one of predicates from the predicate lexicon, *status* is the semantic status of a case-role, such as agent, theme, place, instrument, etc., *value* is a SL string which fills a case-role, *tag* is a label which conveys both morphological information and semantic knowledge as specified in the tagging lexicon (see examples in Figures 7 and 8).

```
text::={ template){template}*
template::={predicate-class    predicate
            ((case-role)(case-role)*}⁸
case-role::= (status value)
value::= {word tag}*
```

Figure 5. A generic structure of claim content representation

4 Workflow

The workflow in multilingual claim generation tools with any embedded MT software essentially includes the steps of:

Technical knowledge elicitation with the following procedures:

- *elicit-type* prompts the user to select the type of patent (apparatus or method)
- *elicit-title* deals with specification of the title of the invention;
- *elicit-parts/method steps* elicits an hierarchy of major components of the invention;
- *elicit-relations* establishes spatial and other relations among invention parts
  - o The top-level procedure here is *retrieving a predicate template****.*** Following the user selection in a SL predicate menu the system displays an interlingual predicate template.
  - o The next procedure is *filling a predicate template* by the user with SL words and phrases**.** It creates SL filled predicate structures, records phrase borders and status of case-roles.
- *elicit-format* prompts the user to select a European or US format.
- *mark-co-references* uses human help in determining the referential indices in the texts.
- *elicit-dependent-claims* establishes references between dependent and the main claims or between dependent claims.

Terminology management. Though the user is encouraged to choose words from the SL menus, he may type a free text. The user can create a new multilingual default entry. Phrases typed in by the user are displayed in the "Your terminology". (Figures 3, 4). Lexical units displayed on the screen can be transferred to a new text area on mouse click and, if necessary be edited.

Building SL claim content representation as a set SL filled predicate structures. Figure 7 shows one of such structures (built on the quantum of user's input shown in Figure 3 and initially written as in Figure 6) for the predicate "bevirkende" ("causing")

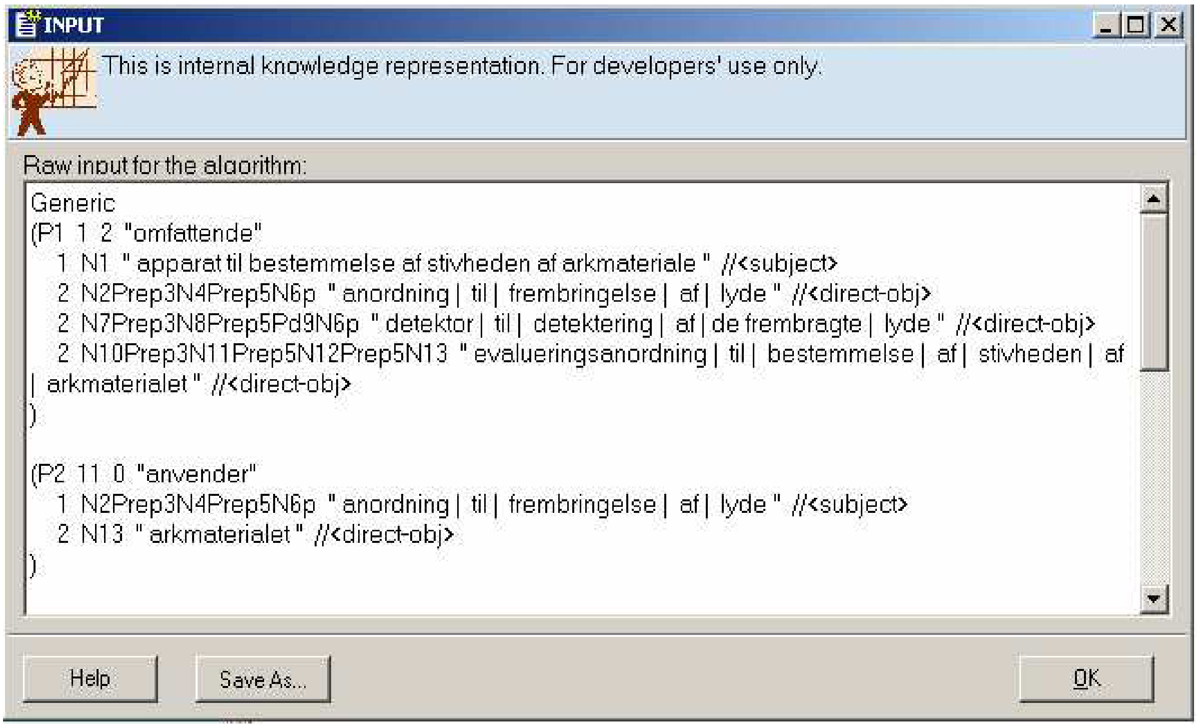
```
(P5  12  6  "bevirkende"
    1  " mekaniske | organer "  //<subject>
    2  " arkmaterialet "  //<direct-obj>
    4  "at | det vibrerer "  //<purpose>)
```

Figure 6. The initial internal representation of a quantum of user's input in Danish.

Building simple sentences in SL based on filled predicate sentences for the user to control his input. This is done either directly by a SL AutoPat Generator as in Dan-Pat (Figure 3), or as a result of a translation procedure as in The J-E patent system with the PC-Transfer engine.

Translation of predicates and case-roles is done entirely by an embedded MT system. In Dan-Pat we translate every Danish string (predicate lexemes and case-role fillers) by the APTrans MT system which first performs analysis (Sheremetyeva, 2003a), then a rule-based transfer into the English language. Translation technique in The J-E patent system is different, see (Neumann, 2005) for details.

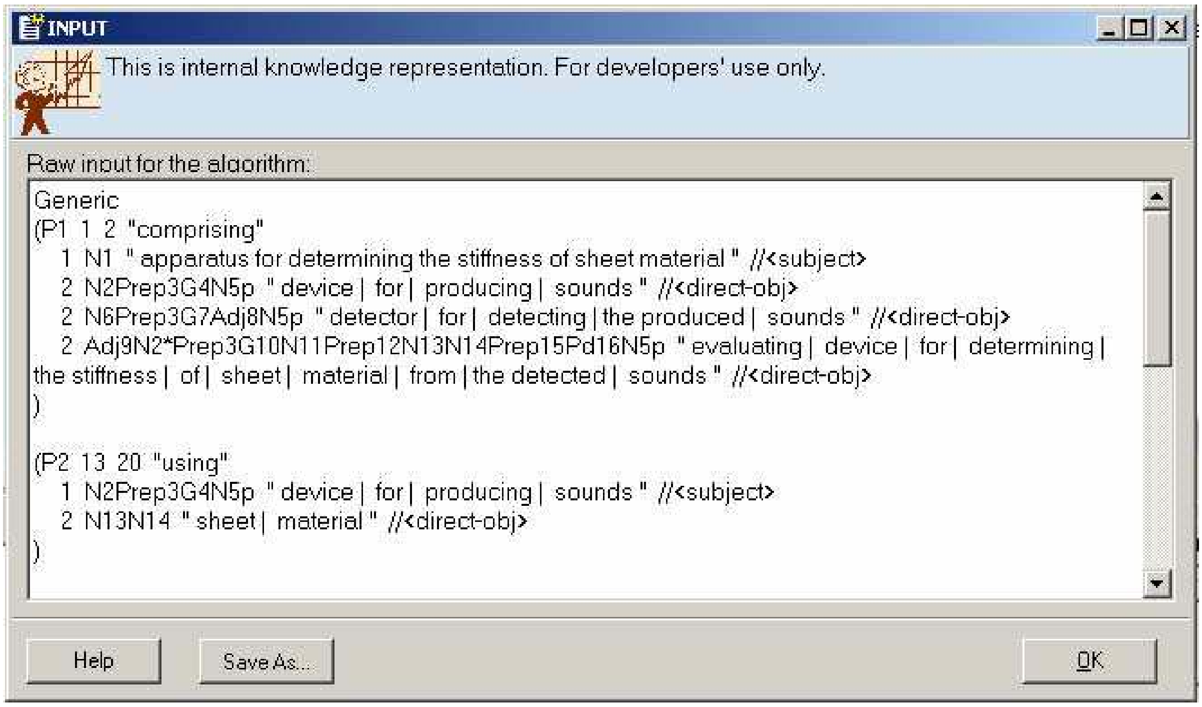
[8] template in the meaning representation is retrieved from a predicate lexicon following the user's predicate selection from a system menu. The interface presentation of such template is shown in Figure 3.


INPUT
This is internal knowledge representation. For developers' use only.
Raw input for the algorithm:
Help
Save As...
OK

```
Generic
(P1 1 2 "omfattende"
   1 N1 " apparat til bestemmelse af stivheden af arkmateriale " //<subject>
   2 N2Prep3N4Prep5N6p " anordning | til | frembringelse | af | lyde " //<direct-obj>
   2 N7Prep3N8Prep5Pd9N6p " detektor | til | detektering | af | de frembragte | lyde " //<direct-obj>
   2 N10Prep3N11Prep5N12Prep5N13 " evalueringsanordning | til | bestemmelse | af | stivheden | af
| arkmaterialet " //<direct-obj>
)

(P2 11 0 "anvender"
   1 N2Prep3N4Prep5N6p " anordning | til | frembringelse | af | lyde " //<subject>
   2 N13 " arkmaterialet " //<direct-obj>
)
```

Figure 7. A screenshot of the developer interface for content representation transfer control which shows a fragment of a set of Danish predicate templates as output by the AutoPat Danish Analyzer.


INPUT
This is internal knowledge representation. For developers' use only.
Raw input for the algorithm:
Help
Save As...
OK

```
Generic
(P1 1 2 "comprising"
   1 N1 " apparatus for determining the stiffness of sheet material " //<subject>
   2 N2Prep3G4N5p " device | for | producing | sounds " //<direct-obj>
   2 N6Prep3G7Adj8N5p " detector | for | detecting | the produced | sounds " //<direct-obj>
   2 Adj9N2*Prep3G10N11Prep12N13N14Prep15Pd16N5p " evaluating | device | for | determining |
the stiffness | of | sheet | material | from | the detected | sounds " //<direct-obj>
)

(P2 13 20 "using"
   1 N2Prep3G4N5p " device | for | producing | sounds " //<subject>
   2 N13N14 " sheet | material " //<direct-obj>
)
```

Figure 8. A screenshot of the developer interface for content representation transfer control which shows a fragment of a set of English predicate templates equivalent to the Danish templates in Figure 7, as output by the AutoTrans MT software.

Building TL content representation is done by substituting SL case-role fillers with their English equivalents output by an MT engine, while preserving the template structure (Figure 8) .

Spelling, Grammar and Template filler check of the user input. Correctness of the user textual input is a crucial point in getting a high quality output in NLP systems (Nyberg et al., 2003).

Our tool provides for automatic completion of words, corrects grammar mistakes in automatic or interactive mode, and corrects wrong fillers of predicate templates.

Generation of patent claim in English is performed by supplying the English content representation (English predicate templates) into the AutoPat Generation engine.

In case of the representation produced by an MT engine other than APTrans (whose output was designed to be compatible with the AutoPat input format), the AutoPat takes "coarse" representation templates with untagged strings and first tags bare English case-role fillers against the AutoPat tagging lexicon and rules, and then inputs thus rewritten templates into the English Generator.

An example of a final English claim generated from the Danish interface is shown in Figure 3.

5 Conclusion

We have described the concept and development issues of a patent domain tool for generating patent claims in English from non-English interfaces. The tool is based on merging AutoPat, the English-English patent claim generator, with any external MT software. The methodology is portable to multiple languages and MT engines.

Two implementations of the multilingual generator of patent claims are currently under way:

The J-E patent system, a tool for generating patent claims in English from a Japanese-only interface, which is being developed by CrossLanguage (Japan) in cooperation with LanA Consulting (Denmark). This system is undergoing (July, 2005) extensive tests and will soon be released on the market (Neumann, 2005).

Dan-Pat, a tool for generating patent claims in English from an interface that takes human input in Danish, - a LanA Consulting project. This system is in the demo version stage with some of the modules fully implemented and well tested (AutoPat Generator, the English part of the knowledge base, developers' environment). The APTrans software is currently being tested and updated. The Danish part of knowledge (lexicons and rules) is to be acquired on a larger scale to provide for commercially acceptable coverage. Preliminary results show a reasonably small number of failures, mainly due to the incompleteness of linguistic knowledge.

We intend to extend the tool to generate claims in a variety of languages (not only in English) from multilingual interfaces taking input in user's native languages other than English and Danish.

6 Acknowledgements

Our thanks go to the Information-technology Promotion Agency, Tokyo, Japan (Grant 2004/374) who partly funded this research.

References

M. L. Bourbeau and R. Kittredge. 1988. *Project d'automation aux brevets et invevtions: traduction assistee par ordinateur (rapport final).* Ministere de la Consommation et des Corporations Canada, Direction des Systemes Automatises, Bureau de la propriete intellectuelle.

L.Chen, N.Tokida, and H.Adachi. A Patent Document Retrieval System Addressing both Semantic and Syntactic Properties. *Proceedings of the ACL Workshop on Patent Corpus Processing*. Sapporo. Japan.

K. A. Fujii and T. Ishikawa. 2002. NTCIR-3 Patent Retrieval Experiments at ULIS. *Proceedings of the Third NTRCIR Workshop.*

M. Gnasa and J. Woch. 2002. Architecture of a knowledge based interactive Information Retrieval System. *http://konvens2002.dfki.de/cd/pdf/12P-gnasa.pdf*

C.Neumann. 2005. Generating Patent Claims in English from a Japanese-Only Interface *Proceedings of the Workshop on Patent Translation in conjunction with the MT Summit X, September.* Phluket, Thailand.

R. Prieto-Diaz. 1993. Status report: software reusability. *IEEE Software 10(3).*

A. Shnimori, M. Okumura, Y. Marukawa, and M. IwaYama. 2002. Rhetorical Structure Analysis of Japanese Patent Claims Using Cue Phrases. *Proceedings of the Third NTRCIR Workshop.*

S. Sheremetyeva, and S. Nirenburg. 1999. Interactive MT As Support For Non-Native Language Authoring. *Proceedings of the MT Summit VII.* Singapore.

S. Sheremetyeva 2003. Towards Designing Natural Language Interfaces. *Proceedings of the 4th International Conference "Computational Linguistics and Intelligent Text Processing"* Mexico City, Mexico.

S. Sheremetyeva 2003a. Natural Language Analysis of Patent Claims. *Proceedings of the workshop "Patent Corpus Processing" in conjunction with 41st Annual Meeting of the Association for Computational Linguistics (ACL 2003)*, Sapporo. Japan, July 7-12.

S. Sheremetyeva 2005. "Less, Easier and Quicker" in Language Acquisition for Patent MT. *Proceedings of the Workshop on Patent Translation in conjunction with the MT Summit X, September.* Phluket, Thailand.

I. Thomas, and Nejmeh B.1992. Definitions of tool integration for environments. IEEE Software. 9(2).