Controlled Authoring In A Hybrid Russian-English Machine Translation System

Svetlana Sheremetyeva
National Research South Ural State University / pr.Lenina 74, 454080
Chelyabinsk, Russia
LanA Consulting ApS/ Moellekrog 4, Vejby, 3210, Copenhagen, Denmark
lanaconsult@mail.dk

Abstract

In this paper we describe the design and deployment of a controlled authoring module in REPAT, a hybrid Russian-English machine translation system for patent claims. Controlled authoring is an interactive procedure that is interwoven with hybrid parsing and simplifies the automatic stage of analysis. Implemented in a pre-editing tool the controlled authoring module can be stand-alone and pipelined to any foreign MT system. Although applied to the Russian-English language pair in the patent domain, the approach described is not specific for the Russian language and can be applied for other languages, domains and types of machine translation application.

1 Introduction

MT systems have become an inherent part of translation activities in spite of general understanding that it is impossible to get high quality machine translation (MT) without human judgment (Koehn, 2009). In addition to lexical ambiguity, among the linguistic phenomena that lower translatability indicators (Underwood and Jongejan, 2001) is the syntactic complexity of a source text, of which the patent claim whose sentence can run for a page or so is an ultimate example.

A wide range of activities can be found in the area of developing different techniques to "help" an MT engine cope with the ambiguity and complexity of the natural language. Recent work investigated the inclusion of interactive computer-human communication at each step of the translation process by, e.g., showing the user various "paths" among all translations of a sentence (Koehn, cf.), or keyboard-driving the user to select the best translation (Macklovitch, 2006). One of the latest publications reports on Patent statistical machine translation (SMT) from English to French where the user drives the segmentation of the input text (Pouliquen et.al, 2011). Another trend to cope with the source text complexity is to rewrite a source text into a controlled language (CL) to ensure that the MT input conforms to the desired vocabulary and grammar constraints. When a controlled language is introduced, the number of parses per sentence can be reduced dramatically compared to the case when a general lexicon and grammar are used to parse specialized domain texts.

Controlled language software is developed with different levels of automation and normally involves interactive authoring (Nyberg et al., 2003). The users (authors) have to be taught the CL guidelines in order to accurately use an appropriate lexicon and grammar during authoring. In line with these studies is the research on developing pre-editing rules, e.g., textual patterns that reformulate the source text in order to improve the source text translatability and MT output. Such rules implemented in a software formalism are applied for controlled language authoring (Bredenkamp et al. 2000; Rayner et al. 2012).

This paper focuses on the design, deployment and utilization of a controlled language in the implementation of the hybrid REPAT environment for machine translation of patent

Proceedings of the 3rd Workshop on Hybrid Approaches to Translation (HyTra) @ EACL 2014, pages 15–20,
Gothenburg, Sweden, April 27, 2014.

claims from Russian into English. In selecting Russian as a source language we were motivated by two major considerations. Firstly, Russia has a huge pool of patents which are unavailable for non-Russian speakers without turning to expensive translation services. The situation is of great disadvantage for international technical knowledge assimilation, dissemination, protection of inventor's rights and patenting of new inventions. Secondly, in an attempt to find ways that could lower efforts in developing MT systems involving inflecting languages, for which statistical techniques normally fail (Sharoff, 2004), we were challenged to develop a hybrid technique for parsing morphologically rich languages on the example of such a highly inflecting language as Russian.

In what follows we first give an overview of the REPAT machine translation environment and then focuse on the components of the system which are responsible for controlled authoring of the source texts with complex syntactic structure, such as patent claims. These components raise the translatability of patent claims and, second, improve their readability in both source and target languages, which for patent claims is of great importance. It is well known that an extremely complex syntactic structure of the patent claim is a problematic issue for understanding (readability) even in a source language (Shinmori et al., 2003), let alone in translation.

2 REPAT environment overview

The REPAT system takes a Russian patent claim as input and produces translations at two major levels, the level of terminology (not just any chunks), and the text level. Full translation of a patent claim is output in two formats, - in the form of one sentence meeting all legal requirements to the claim text, and as a better readable set of simple sentences in the target language. In Figure 3 an example of the REPAT output is shown for a fragment of a Russian claim given below:

Стеклоподъемник автомобиля содержащий электропривод и направляющую с ползуном, отличающися тем, что в ползуне выполнены два гнезда, образованные пластиной и выемками во вкладыше, в которых расположены параллельно друг другу две цилиндрические витые пружины для компенсации вытяжки каната...

The system also improves the readability of a source claim by decomposing it into a set of simple sentences that can be useful for a posteditor to better understand the input and thus control the quality of claim translation. The REPAT translation environment includes hybrid modules for source language analysis, controlled authoring, terminology management, knowledge development and rule-based modules for transfer and target text generation. All modules work on controlled language which is built into the system. The overall architecture of the system is shown in Figure 1. The workflow includes these main steps:

Source claim shallow analysis based on hybrid techniques. It serves two purposes : a) the on-the-fly translation of terminology; this can be used by a non-SL speaker for digest, and b) the preparation of a raw document for authoring in case a full claim translation is needed; the input is made interactive and the nominal and predicate terms are highlighted, the predicate terminology is linked to the knowledge base.

Terminology update. The document is checked against the system bilingual lexicon and unknown words are flagged. If needed the lexicon can be updated.

Authoring. The document is authored to conform the controlled lexicon and grammar. Unknown words are either avoided or flagged. The source claim syntactic structure is simplified. The simplification also serves the purpose of improving the readability of a source language claim.

Document processing and translation. This includes document parsing into a formal content representation, generation of a source claim in a controlled language, crosslinguistic transfer and generation of the target text. The full translation is output in two controlled syntax formats, a) as one complex sentence meeting all legal requirements to the claim text, and d) as a better readable set of simple sentences that might meet the needs of the user in case the translation is needed to assimilate technical knowledge rather than to be included in a patent document. The simplified syntactic presentation of translation can be useful for further automatic claim processing, e.g., when translation into other languages is needed.

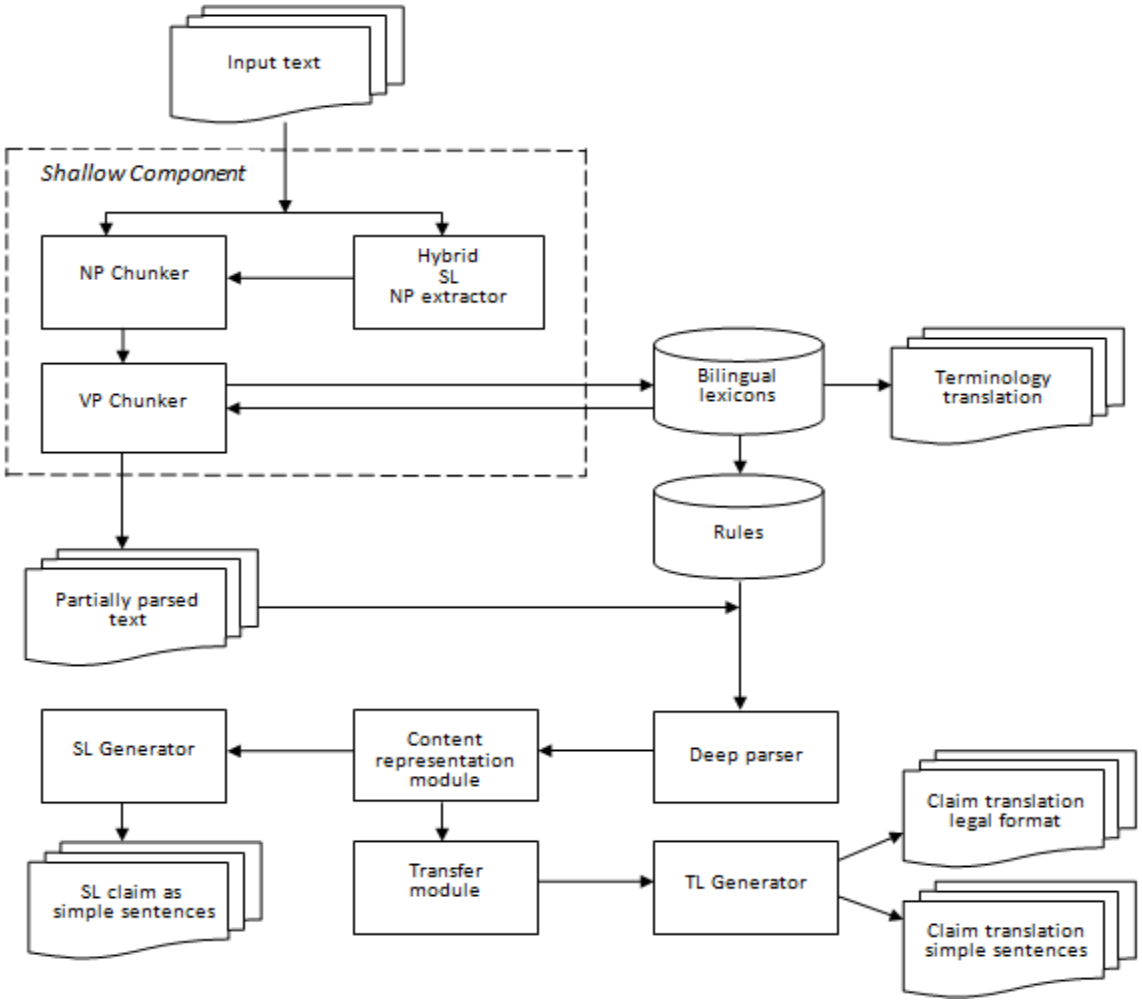

Figure 1. An overall architecture of the hybrid REPAT system.

3 Controlled language

The system controlled language specifies constraints on the lexicon and constraints on the complexity of sentences. It draws heavily on the patent claim sublanguage on devices in automobile industry, and in addition to the universal phenomena affecting translatability (Underwood and Jongejan, cf.) it addresses the REPAT engine-specific constraints.

Constraints of the REPAT controlled language are mainly coded in the corpus-based system lexicon, where ambiguous terms, that unavoidably emerge in any doimain are split in different lexemes, each having only one domain meaning. Where possible ambiguous lexemes are put in the lexicon as components of longer terms/phrases with one meaning. To disambiguate the residue of ambiguous terms we have created a method for disambiguation of lexical items that supports interactive disambiguation by the user through the system user interface.

Grammar restrictions on the structure of sentences are set by an implicitly controlled grammar which is associated with a controlled set of predicate/argument patterns in the system lexicon rather than with syntactic sentence-level constraints. The patterns code domain-based information on the most frequent co-occurrences of predicates in finite forms 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 controlled language restrictions are imposed on the source text semi-automatically. The system prompts the user to make correct authoring decisions by providing structural templates from the system knowledge base and by raising the users' awareness about the linguistic phenomena that can increase the

potential problems in machine translation. For example, the users are encouraged to repeat a preposition or a noun in conjoined constructions, limit the use of pronouns and conjunctions, put participles specifying a noun in postposition, etc.

4 Analyzer and authoring engine

Authoring engine is interwoven with the system hybrid analyzer. The analyzer performs two tasks in the REPAT system. It analyzers the input text into a formal internal representation and provides environment for authoring. In particular, the analyzer performs the following authoring-related steps:

Segmentation and lexicalization. The input text is chunked into noun phrases (NPs) predicate phrases (VPs) and other types of lexical units. Every chunk is lexicalized by associating it with a known lexicon entry.

The source NPs are chunked based on the dynamic knowledge automatically produced by a stand-alone hybrid extractor, the core of the REPAT shallow parsing component. It was ported to the Russian language following the methodology of NP extraction for English described in (Sheremetyeva 2009). The extraction methodology combines statistical techniques, heuristics and a shallow linguistic knowledge. The extractor does not rely on a preconstructed corpus, works on small texts, does not miss low frequency units and can reliably extract all NPs from an input text. The extraction results do not deteriorate when the extraction methodology is applied to inflecting languages (Russian in our case).

The NPs are chunked by matching the extractor output (lists the source claim NPs in their text form) against the claim text. Here the language rich inflection properties turn to be an advantage: the NP chunking procedure proves to be very robust with practically no ambiguity. NPs excluded, the rest of the claim lexica is chunked by the lexicon look-up practically without (ambiguity) problems. The analyzer thus trigs highlighting of the nominal and verbal terminology, flags unknown words and provides means for lexical disambiguation. All lexicalized chunks are tagged with supertags coding sets of typed features as found in the morphological zones of the lexicon.

Automatic and Interactive Disambiguation. Ambiguity of lexical units are resolved, either via a) automatic selection of the most likely meaning, using a set of disambiguation heuristics, or b) interactive clarification with the user. Syntactic ambiguity is to be resolved by human-computer interaction with strong computer support in the form of predicate templates to be filled with claim segments.

Content representation. A formal internal representation of the source claim content is built in the following two steps:

Construction of the underspecified internal representations resulting from the authoring procedure of calling and filling predicate templates by the user. A predicate template is a visualization of a corresponding predicate case-role pattern in the system lexicon. The main slot in the template corresponds to the predicate, while other slots represent case-roles. By supplying fillers into the slots of predicate templates the user in fact puts syntactic borders between the argument phrases and determines the dependency relations between the predicates and their arguments.

Automatic completion of tagging and recursive chunking by the deep parser component that works over the set of the disambiguating features of the underspecified content representation. The final parse, a set of tagged predicate/argument structures, is then submitted into a) the source language generator that outputs a source claim in a more readable format of simple sentences, and b) to the transfer module and then to the target language generator, that outputs translations in two formats.

5 Authoring Interface

A screenshot of the REPAT authoring interface is shown in Figure 2. In the left pane it shows an interactive source claim with nominal and predicate terminology highlighted in different colours. Unknown words, if any, will be flagged. The user is encouraged not to use such words and remove the flag. In case the user considers them necessary, the flag stays (the terms are passed to the developer for lexicon update). The highlighted terminology improves the input readability and helps the user quicker and better understand the input content and structure. To simplify the input structure the user clicks on a predicate and gets a pop-up template whose slots are to be filled out with texts strings. Predicate templates are generated based on the case-role patterns in the system lexicon.

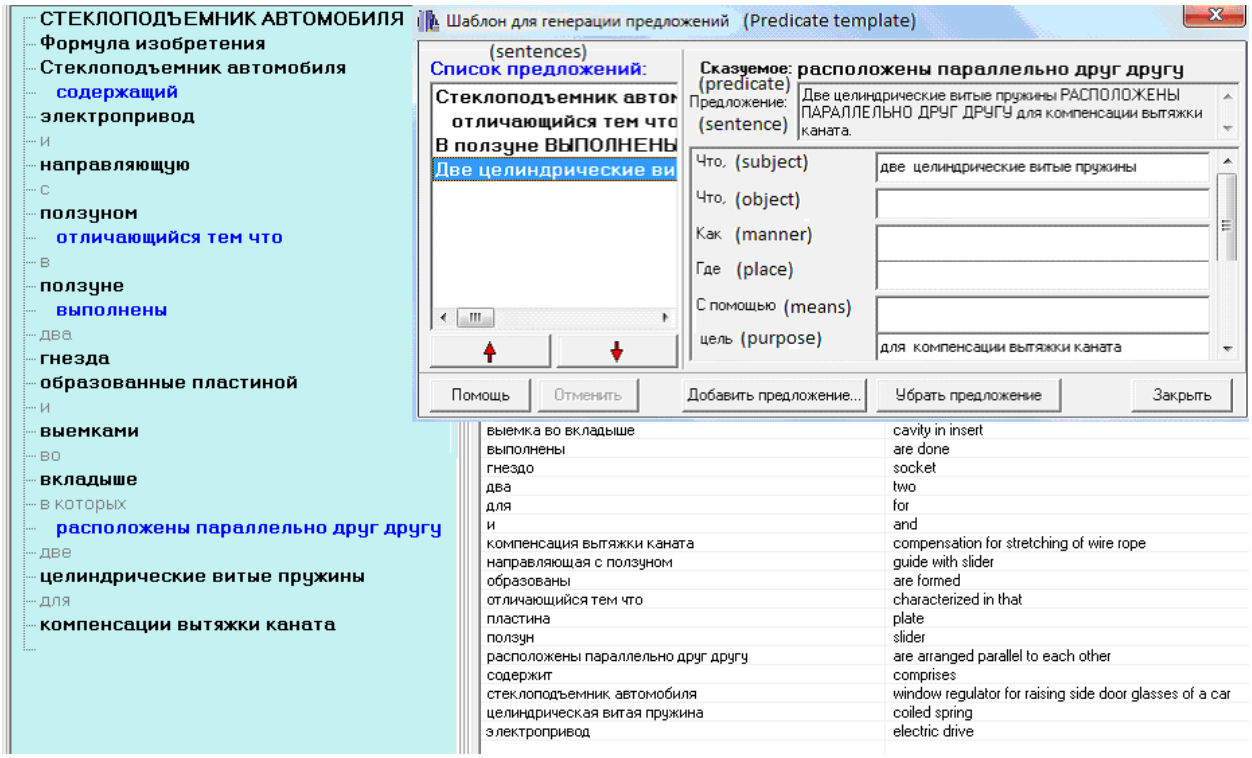

Figure 2. A screenshot of the user interface showing the authoring set up for a fragment of the Russian claim given in Section 2. The source text with visualized terms is shown in the left pane. In the middle is the template for the Russian predicate *является (is)*. The English translations for the terminology are shown in the bottom of the right pane.

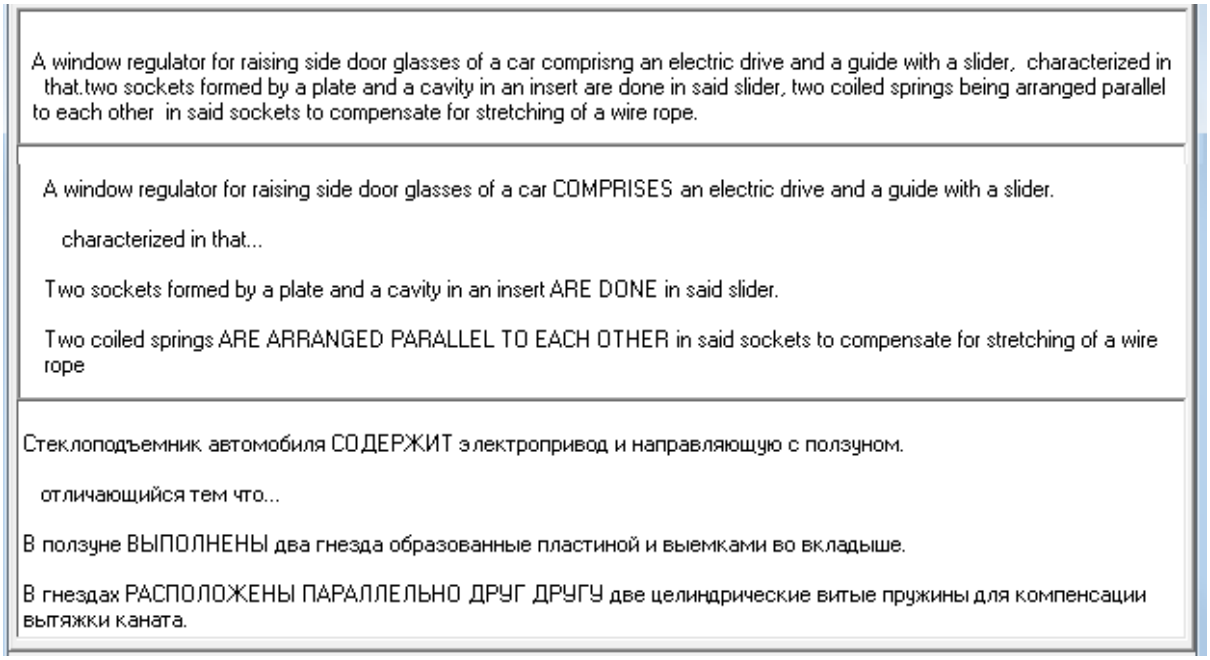

Figure 3. The two translation variants of the patent claim fragment given in Section 2. On the top the claim translation into English in the legal format of one nominal sentence is shown. In the middle the "better readable" claim translation in the form of simple sentences is displayed. In the bottom the authored Russian input text is given.

The main slot of the template is automatically filled with a predicate in a finite form, not withstanding in which form the predicate was used in the text. Other predicate slots are referenced to particular case-roles whose semantic statuses are explained to the user by the questions next to the predicate slots. The user can either drag-and-drop appropriate segments from the interactive claim text or simply type the text in the slots. During the process of filling the template the system shows translations of the lexica used in the bottom of the right pane. In case a unit put in the slot is not found in the lexicon, it is flagged. The user is encouraged to either avoid using a problematic unit or substitute it with a synonym known to the system. Once the template is filled, the system automatically generates a grammatically correct simple sentence in the source language and displays it for control. In addition to constraining the complexity of the sentence structure predicate templates also put certain constraints on the phrase level. As templates are meant for simple sentences only, coordination of verbal phrases (predicates) that may be ambiguous is avoided. Prepositions or particles attached to the verb are put to the main (predicate) template slot that resolves a possible attachment ambiguity.

The authoring procedure completed, the underspecified content representation built by the analyzer "behind the scenes" is passed to the other modules of the REPAT for translation. The authored claim in the source language can also be saved and input in any foreign MT system.

Conclusions

We presented an authoring environment integrated in the hybrid PATMT system for translating patent claims. The efficiency of the system is conditioned by the controlled language framework. The controlled language data are created based on the domain-specific analysis of the patent corpus on devices in automobile industry. The constraints of the controlled language are embedded into the system knowledge base and included into a comprehensive, self-paced training material.

The authoring environment is interwoven with hybrid analysis components specially developed for inflecting languages. Rich morphology turns out to be an advantage in our approach. A great variety of morphological forms significantly lowers ambiguity in source text chunking and lexicalization.

The system is implemented in the programming language C++ for the Windows operational environment.

References

Bredenkamp, A., Crysmann, B., and Petrea, M. 2000. Looking for Errors: A Declarative Formalism for Resource-Adaptive Language Checking. *Proceedings of LREC 2000*. Athens, Greece.

Koehn Philipp. 2009. A process study of computer-aided translation, *Philipp Koehn, Machine Translation Journal,* 2009, volume 23, number 4.

Macklovitch, Elliott. 2006. TransType2: The last word. *In proceedings of LREC06*, Genoa, May.

Nyberg E., T Mitamura, D. Svoboda, J. Ko, K. Baker, J. Micher 2003. An Integrated system for Source language Checking, Analysis and Terminology management. *Proceedings of Machine Translation Summit IX,* September. New-Orleans.USA

Pouliquen Bruno, Christophe Mazenc Aldo Iorio. 2011. Tapta: A user-driven translation system for patent documents based on domain-aware Statistical Machine. *Proceedings of the EAMT Conference.* Leuven, Belgium, May.

Rayner, M., Bouillon, P., and Haddow, B. 2012. Using Source-Language Transformations to Address Register Mismatches in SMT. In *Proceedings of the Conference of the Association for Machine Translation in the Americas (AMTA),* October, San Diego, USA.

Sharoff, S. 2004. What is at stake: a case study of Russian expressions starting with a preposition. In: *Proceedings of the Second ACL Workshop on Multiword Expressions Integrating Processing*.

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

Shinmori A., Okumura M., Marukawa Y. Iwayama M. 2003. Patent Claim Processing for Readability - Structure Analysis and Term Explanation, *Workshop on Patent Corpus Processing. conjunction with ACL 2003*, Sapporo. Japan, July.

Underwood N.L. and Jongejan B. 2001. Translatability Checker: A Tool to Help Decide Whether to Use MT. *Proceedings of MT Summit VIII,* Santiago de Compostela, Spain.