Dan2eng: Wide-Coverage Danish-English Machine Translation

Eckhard Bick

Institute of Language and Communication
University of Southern Denmark
Campusvej 55, DK 5230 Odense M, Denmark
eckhard.bick@mail.dk

Abstract

The paper presents and evaluates a wide coverage, rule-governed machine translation system for Danish-English. Analysis and polysemy resolution are based on Constraint Grammar dependency trees. In its 85.000 lexeme lexicon, Dan2eng uses context-sensitive lexical transfer rules linking dependencies to semantic prototype conditions, syntactic function, definiteness etc. Dependency is further exploited instead of constituent bracketing to support syntactic movement rules. A robust derivational and compound analysis, as well as a separate NER module permit the handling of unrestricted text from a wide range of genres. The system averaged TER scores of 7 (BLEU 0.55-0.6) on student tasks, but performance varied widely against raw and edited Europarl references, respectively.

1 Introduction

Machine translation (MT) has seen alternating phases of enthusiastic public funding and vehement rejection of the feasibility of the concept, and on the way subscribed to a number of different technological paradigms such as the naïve word-for-word approach, generative syntax, neural networks, artificial intelligence, memory-based translation, probabilistic and corpus-based MT. A distinction can also be made as to applicative ambition - while full MT promises translation proper, with or without human revision, *computer assisted translation (CAT)* only aims at supporting the human translator through automatic dictionary look-ups, term banks and storage of previously translated sentences.

Dan2eng, the topic of this paper, adheres to the former camp in that it targets unrestricted text and uses handcrafted linguistic rules rather than probabilistic methods. The system was developed over a 2-year period on top of an existing rule-based Constraint Grammar and dependency parser for Danish (Bick 2003).

2 System architecture

The central idea is to reduce machine translation to good source language (SL) analysis, i.e. to address the largest possible share of MT tasks (polysemy, syntactic movement etc.) by drawing on categorial information and structure provided by SL analysis, such as dependency relations, syntactic function categories and selection restrictions for semantic prototypes. Semantics is thus expressed, with a Halliday'an expression, as *ever more fine-grained syntax.* In this vein, contextual-structural rules are used at 5 different levels:

(a) A Danish Constraint Grammar (DanGram) with rules for morphological and part of speech (PoS) disambiguation, as well as mapping and disambiguation of syntactic functions (~6000 rules)

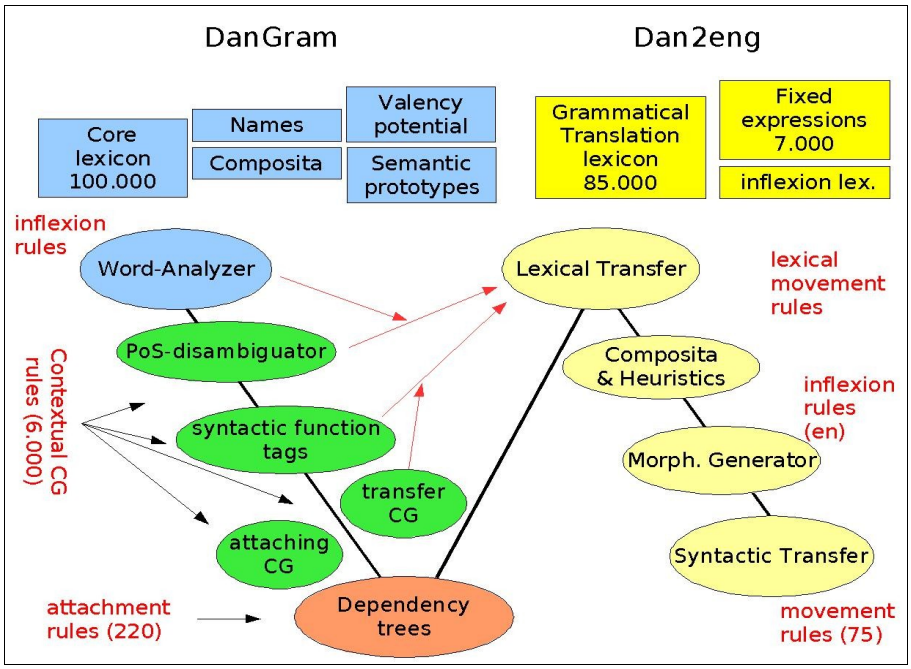
(b) Dependency rules establishing syntactic-semantic links between words or multi-word expressions (MWEs), ~ 220 rules

(c) Lexical transfer rules, selecting translation equivalents according to grammatical category, dependency and other structural context (17.000 rules)

(d) Generation rules for inflexion, verb chains, compositа etc. (~ 700 rules)

(e) Syntactic transformation (movement) rules to establish English word order, handle subclauses, negation, questions etc. (75 rules)

In this, CG rules are not restricted to step (a), but are used at all levels to add or alter grammatical tags to be used by - or to trigger - other rule types (b-e) not themselves formulated in the CG formalism. For instance, the original DanGram CG only adds dependency direction markers to its function tags (e.g. @<SUBJ for a subject to the right of its verb), underspecifying attachment distance and coordination, which have to be addressed by an additional CG layer prior to (b).



Figur 1: Dan2Eng system architecture

3 Lexical transfer

3.1 Transfer ambiguity

A common lay person misconception is imagining all MT as simply a "list look-up process", where source and target language (TL) words match each other 1:1. Ironically this is exactly the technique many publically available systems for smaller languages subscribe to. Apart from an untoward word order effect, word-by-word translation cannot handle the semantic many-to-many relations of natural language word translations. Thus, in Danish-English MT, it does not hurt that both *søjle* and *spalte* can be translated as *column* (many-to-one), but it *is* problematic that *spalte* can mean either *column* or *crack* (one-to-many).

Dansk	***Engelsk***
søjle	column
spalte	to split
	crack
narko	narcotics

Figure 2: many-to-many translation matches

The list translation problem can, of course, be solved by choosing the most "prototypical" or most frequent translation among several possible ones, but this does not resolve the underlying polysemy issue. Two improvement strategies were used in Dan2eng, the first one-dimensional, the other two-dimensional. The former works by using lexemes rather than tokens, exploiting the fact that the underlying parser, DanGram, introduces a lexeme distinction if a token occurs with different PoS (e.g. *spalte [noun] = column/crack, spalte [verb] = to split*), different inflexion paradigms or different phonemic transscription. Both PoS and inflexion are used in the translation of *så:*

- så V PAST (past tense verb) - 'saw'
- så V INF (infinitive) - 'sow'
- så ADV (adverb) - 'so'
- så KS (subordinator) - 'so_that'
- så KC (coordinator) - 'thus'

The 2-dimensional strategy replaces the translation list with a translation matrix, subdividing each lexeme into different senses. But while the 1-dimensional *grammatical* subdivision inherits disambiguation almost "for free" from the grammatical CG-annotation, it is necessary in the 2-dimensional polysemy resolution to introduce *sense distinctors* - a kind of lexical mini-rules that combine context conditions (global distinctors) with e.g. affix- or definiteness conditions (local distinctors).

In practice, of course, lexicographers do not agree as to how many senses a given Danish word should be subdivided into. While large dictionaries often list, for instance, metaphorical or genre-specific usage as a separate sense, such fine-grainedness is not necessarily desirable in a direct (interlingua-free) MT system. After all, both metaphoric usage and genre transfer are productive and may work the same way in the target language. Dan2eng therefore regards transfer as a *distinction* task, not a *defining* task, employing not as much meanings *per se*, but rather *translation equivalents*. On the other hand, the 2-dimensional distinctor approach may also cover the cases where differences in translation are not due to meaning differences, but rather syntactic or idiosyncratic TL conventions. The following is an example of the 2-dimensional distinctor model for the word *meget:*

meget_ADV :a_lot; S=(>A) :very; D=(>A) :much
meget_DET ...
meget_INDP ...

Apart from the 1-dimensional distinction (determiner, independent pronoun or adverb), we have shown the second distinction dimension for the latter: As an adverb, *meget* is to be translated as *very*, if the word itself (S) functions as adverbial modifier (@>A), while *much* is used if the words heads another adverbial modifier as dependent (D). If no distinctor can be instantiated, the translation *a lot* is chosen.

3.2 Local distinctors

Apart from the above mentioned PoS distinctions, also other grammatical features may be used as local (1-dimensional) distinctors.

(a) Numerus: *slægt_N ... S=(S) :family; S=(P) :generation* [S = singular, P = plural]
(b) Genus: *rod_N ...* S=(UTR) :root; S=(NEU) :mess [UTR = common gender, NEU = neuter gender]
(c) tempus: *måtte_V ... S=(PR) :must; S=(IMPF) :have_to* [PR =present tense, IMPF =preterite]

Local features may, of course, also be instantiated through non-local relations - e.g. np-agreement in the face of a gender-ambiguous prenominal and a gender-unambiguous head.

A local tag with a strong contextual note is the syntactic function tag (subject, object etc.), which in CG annotation is assigned to a token, but reflects deep sentence analysis. Locally, such a tag can be used to, for instance, distinguish between "noun-like" or "adjectival" meaning of participle. Here, prenominal function (@>N) and predicative function (@SC) are read as adjectival, while subject (@SUBJ) and object readings (@ACC) will trigger a non-like reading:

(d) boligsøgende_PCP2 @>N
--> boligsøgende_ADJ :house-hunting
(e) boligsøgende_PCP2 (@SUBJ|@ACC)
--> boligsøgende_N :house-hunter

3.3 Contextual distinctors

Though local distinctors are useful and easy to implement, it is the dependency-based contextual distinctors that constitute the backbone of the system and have the largest development potential.
The multi-semous Danish verb *at regne (rain, calculate, consider, expect, convert ...)*, may serve as an example. Rather than ignore the less-frequent readings, or let an "AI"-module try to *understand* these meanings through world models or frames, Dan2eng choses a pragmatic middle path where *distinctors* are used to chose translation equivalents solely on the grounds of structurally deep SL analysis. Thus, the translation *rain (a)* is chosen if a daughter/dependent (D) is found with the function of *situative/formal subject (@S-SUBJ)*, while most other readings ask for a human subject. The default translation for the latter case is *calculate (f)*, but the presence of other dependents (objects or particles) can elicitate other translations. *regne med (c-e)*, for instance, is interpreted as *include*, if *med* is marked as an adverb, while the preposition *med* will trigger the translation *cont on* in connection with human granddaughter dependents (GD=<H>), and the translation *expect* in all other cases. Note that the translation *include* could also be singled out through the condition of a direct object (D=@ACC), but not on its own, barring confusion with (b), *regne for ('consider')*, which also governs a direct object.

regne_V[1]
(a) D=(@S-SUBJ) :rain;
(b) D=(<H> @ACC) D=("for" PRP)_nil :consider;
(c) D=("med" PRP)_on GD=(<H>) :count;
(d) D=("med" PRP)_nil :expect;
(e) D=(@ACC) D=("med" ADV)_nil :include;
(f) D=(<H> @SUBJ) D?=("på" PRP)_nil :calculate;

It has to be stressed that the use of grammatical relations in the choice of translation equivalents is methodologically very different from a *translation memory* approach or corpus-based machine learning approach, where words or sequences of words are matched in bilingual corpora of existing translations.

First, in such methods - or at least in their naïve, lexicon-free form - it may be difficult or impossible to generalize over semantic types (e.g. <H> for 'human') or syntactic functions, making the approach vulnerable to the "sparse data" problem[2]. Second, contiguous collocations (n-grams) are less robust then function-based dependency relations which also allow discontinuity, with interfering material such as modifiers or subclauses. Finally. a category based method has the advantage over a token based one of being robust with respect to inflexion and lexical variation.

3.4 Polylexicals

Dan2eng recognizes and translates different types of multi word expressions (MWEs). A small part (e.g. *i_stedet_for - instead_*of), treated like simplex words, is inherited from the parser reflecting its need for syntactically manageable "tokens".

---

[1] The full list of distinctors for this verb consists of in all 13 items, among them several prepositional complements not shown in the example *(regne efter, blandt, fra, om, sammen, ud, fejl ...)*

[2] For the language pair Danish-English, parallel corpora of any size are hard to find. Even the 25 million word Europarl corpus (www.iccs.inf.ed.ac.uk/~koehn/publication/europarl) is small for lexical purposes and does not have the genre coverage necessary for building a full system.

The second category is made up of names, recognized by DanGram as MWE chains and classified semantically as *person*, *place*, *event*, *organisation* etc. To treat names as single units facilitates the matching of selection restrictions, such as +HUM for the subject slot of a cognitive verb.

The third group of MWEs is the closest Dan2eng gets to a memory-based translation[3] list:

- terms (complex nouns): *aflåst sideleje - recovery position*
- pp's, np's or adjp's with a non-compositional translation: *af gammel vane (pp) - from habit, bleg om næbbet (adjp) - green about the gills*
- fixed expressions: *bordet fanger - a bargain is a bargain*

When a expression from the list is matched in the text, it is allowed to override other strategies, avoiding analytical translations in these cases. Even terms and idiomatic expressions *can*, however, be treated by rules in the ordinary translation lexicon. Though more cumbersome, this alternative is to be preferred for expressions with inflexional or lexical variation. Thus, the idiomatic *skøn sild* (a) inflects in number and allows other attributive variation *(skøn, dejlig, smuk).* In the same fashion, the verb *male* (b) allows variation in tense and finity *(male, malet, malede).* In both cases, lexical rules are used rather than listing all variants:

(a) skønne sild sild_N :herring;
D=("(dejlig | skøn | smuk)") :girl
(b) male byen rød male_V :paint; ....
D=("by" DEF @ACC)_nil;
D=("rød" @OC)_nil :have some serious fun

3.5 Compounds and names

Along with other morphological information, the DanGram parser also provides a compound and derivation analysis, in the form of a secondary tag, as in the following example, containing no affixes, but 2 noun-roots (the first marked N:)... and a ligature-s:

oversættelsesudvalg
<N:oversættelse~s+udvalg> N UTR S IDF NOM

Since compounding is a productive process in Danish, an analysis like the above is often not to be found in the parser's lexicon, but rather a byproduct of active derivational analysis.

The transfer module of Dan2eng draws on the compound analysis, if no complete entry can be found in the bilingual lexicon. In this case, the word is translated stepwise, using the individual part's status as e.g. affix, first or last root lexeme, as well as its internal "PoS". Thus, the translation of a part-lexeme (a) may be different from its translation as an independent word (b)

(a) oversættelse~s+udvalg --> translation committee
(b) FN-styrke --> UN force (with a 2.root-distinctor, not *UN-strength*)

An evaluation of 313 running compounds from the Danish Europarl corpus (http://people.csail.mit.edu/koehn/publications/europarl/) showed that only 7% of compound translations resulted in odd lexeme combinations[4], plus 2% of dubious cases and 2% of generation inflexion errors. Only 2 failures were due to PoS mistaggings, and in 1 case, no translation was suggested.

In spite of a robust markup of even multi-part names, it can be difficult to determine a translation of names not registered i the lexicon. The current strategy is to exploit the semantic name-tag provided by DanGram, and distinguish between person names and brand on the one hand (a-b) and institution and event names on the other hand (c-d), translating the latter part-by-part, but not the former.

(a) *Georg Jensen* - **George Johnsson
(b) *Den Danske Bank* - *The Danish Bank
(c) *Det Danske Sprog- og Litteraturselskab* - the Danish Society of Language and Literature
(d) *Rådet for Større Færdelssikkerhed* - the Council for Greater Traffic Safety

4 Structural transfer

Dan2eng's other transfer component is the structural transfer from Danish to English syntax. Such transfer is necessary where no TL word can be found that can fill the same syntactic slot as its SL counterpart. A simple example are Danish s-passives, which do not have an English inflexional equivalent, but have to be rendered with an auxiliary construction *(be+...ed).* Since its SL origin is inflexional, the phenomenon is handled in the morphological generation module that turns the finite verb into a participle and adds the auxiliary.

Private biler sælges ikke uden moms
- Private cars aren't sold without VAT

The example contains another type of structural change - the negation particle *ikke/not* had to be moved to a within the newly created verb chain.

[3] This list is accessible to user editing, and can be augmented by true translation memory, as well as user-provided term banks

[4] This is one of the areas where "statistical smoothing" module (chapter 7) comes into play, using a monolingual database to corroborate or change choices made between translation alternatives for compound parts

This second type of transformations (movement rules) is controlled by a separate topological grammar module, run *after* lexical transfer and morphological generation. This grammar, too, exploits DanGram's dependency annotation, allowing a rule to specify whether a word is to be moved with or without its complete set of dependents. Thus, noun "mothers", for instance, can be made to carry their attributes or relative clauses with them[5].

One of the major differences between Danish and English syntax is VS - SV alternation in the presence of a non-subject front field:

I dag @ADVL *drikker @FMV vi @SUBJ vin @ACC* - ***Today*** *we drink wine*

The relevant transformation rule first lists the involved "constituents" (dependency heads), then maps into a new (English) order using position numbers:

(@ADVL|@ACC|@FS-ADVL|@>>P),	*I_dag*
w(@FMV|@FAUX|@FS-[^Q]+),	*drikker*
w(@ICL-AUX<)?,	
w(@ADVL)?,	
(@SUBJ|@F-SUBJ|@S-SUBJ)	*vi*
-> 1, 5, 2, 3, 4	

Each constituent can be quantified (? = optional, * = none or more, + = 1 or more). A prefixed 'w' og 'g' means, respectively, moving only the word, or all group level dependents (but not subclauses). In the next example, the optional field 3 (auxiliary complement), is still empty i Danish, but will be filled in the English translation:

I dag:today @ADVL (1) _:are @FAUX (2)
sælges:sold @ICL-AUX< (3) igen:again @ADVL
(4) flere @>N biler @<SUBJ (5)

It is not possible to treat *are_sold* as one field, since it has to be possible for other, later rules to move the adverb *ikke:not* to the inside of the newly created verb chain. In a token-based annotation like DanGram's, adding and splitting tokens poses a much greater challenge than substituting nouns with - unsplittable - np's or vice versa. Other examples of token-adding transfer result from English *'do'*-negations and the interdiction of sentence-initial tempus-inflected main verbs. Thus, English needs to add an *'if'*, where Danish can construct a conditional clause through VS-inversion alone:

Kommer han ikke, har vi et problem

*--- **If** he doesn't come, we have a problem.*

In general, movement of head-dependent chunks works much the same way constituents are treated in a generative grammar, but complications may arise when a movement targets not sister-nodes, but mother- and daughter nodes, as in the case of a finite auxiliary that is to swap places with its subject-daughter. This special case is marked with a w-prefix (word-only-movement).

In addition to its movement grammar, Dan2eng also has access to lexicon-based methods of structural transfer, the '+' operator and function variables, which both can be tailored to one individual translation equivalent at a time. The '+' operator can be used to simulate number movement in the translation of *klokken 4* as *4 o'-clock:*

klokke_N (1) :bell; (2) S=(DEF (@ADVL | @N<))
P1=(NUM)_[+o'clock] :SIC-at;
(3) S=(DEF) P1=(NUM)_[+o'clock] :nil;

Function variables are e.g. used to control the movement of objects in a lexical way:

*Jeg tiltror **ham** @<DAT det værste*
*--- I think **him** @<DAT capable of the worst*
*De overhældte hende med **maling** @P<*
*---They poured **paint** @<ACC over her*

Here, a function variable (e.g. DATive object) is first inserted into a multi-word translation equivalent (a). Next, a general transformation rule moves the @DAT head (and all its dependents!) into the assigned slot (b). The generalized version works with a MOVE variable, introduced by a lexical rule (c) and later "filled" by a rule from the movement grammar module (d).

(a) tiltro_V :trust;
D=(@DAT) :think=DAT=capable=of
(b) w(:DAT.*), w(:nil)?, (@DAT) -> 3,1,2
(c) overhælde_V :drench; D=("med")_nil[@.*->
@MOVE] :pour=MOVE=over
(d) w(:MOVE=.*), (@ACC), (@MOVE) -> 3,1,2

5 Applications

Currently, 4 interfaces are available for Dan2eng, all using a CGI-solution with server-side analysis, and handling unrestricted, running text at about 100 words/sec:

(a) A web interface for text field input (http://beta.visl.sdu.dk and http://visl.dk)
(b) A web page translator for URL input
(c) A browser plugin
(d) Remote access for automatic interfacing

Most current users use (a) and (b), and only a few test-contacts have received (c). The latter is, however, the most ambitious regarding preservation of formatting and dynamic text. As to (d), an early version of Dan2eng was successfully integrated into a mobile phone service (Ahlmann & Bick, 2006), retrieving Danish weather forecasts and offering them in English to tourists on the island of Funen.

[5] This use of dependency is more or less equivalent to the role of constituents in a generative grammar.

6 Evaluation

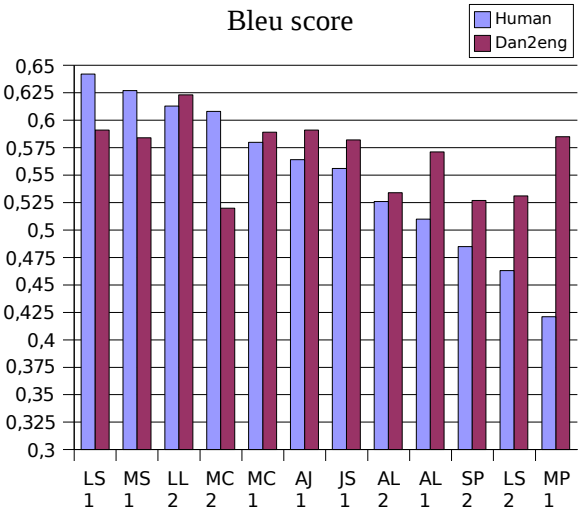
Encouraged by positive web-user feedback, we performed a first evaluation of Dan2eng on one Danish-English translation exam and two translation home tasks, all three at university level (SDU). In a best case scenario, with manually corrected output was used as a reference, the system achieved TER edit distance scores (Translation Error Rate, Snover 2006) in the single digit range:

	ins	***del***	***sub***	***shift***	***TER***
exam	32%	21%	42%	5%	**7.79**
home1	27%	23%	36%	14%	**8.91**
home2	18%	18%	55%	9%	**4.47**

Table1: Error types

It can be seen from the relative error type percentages that most edits were substitutions, while word order (shift edits) fared best, an outcome likely to reflect an essential difference between rule and statistics based systems, where the former are good at long distance structure but tend to choose prototypical word translations rather than n-gram based synonymy variation.

All available original student translations (7 for the first home task, and 5 for the second), were then evaluated for comparison, each being assigned a BLEU score (Papineni et al. 2002) using all remaining human translations as a reference set. Raw system output was measured against the same sets. Here, Dan2eng achieved rank 3 (out of 8) in task one, and rank 2 (out of 6) in task two, both in terms of absolute score (6.00 for task 1, and 5.48 for task 2), and when compared to human translations one by one.



Figur 3: Student translation comparison

Dan2eng was also tested on Europarl data (1324 sentences, 35200 words). Here, though the system produced intelligible translations without meaning loss, it often used synonyms and ad-hoc MWEs that did not match the political jargon typical of this corpus, resulting in a Bleu score of 0.2, lower than the 0.285 reported in (Koehn, 2005) for statistical MT on the same corpus. On the other hand, not much editing was needed to turn output into correct translations, without grammatical errors and with idiomatic word order. Thus, a chunk of *edited* translations (44 sentences, 1032 words), compared to its own, unedited original with a Bleu score of 0.8261 and a TER score of 11.185. It must also be stressed that all figures are for the general purpose system, and no lexical-statistical domain tuning (cf. chapter 7) was performed. Danish MT systems being a very rare commodity, no comparable published results for rule-based systems could be found at the time of writing. Thus, PaTrans (Maegaard & Hansen, 1995), a rule-based system for the patent domain, translates in the other direction (English-Danish), uses manual preprocessing and has not published BLEU or TER scores. The recently presented SDMT-SMV project at the Copenhagen Business School orally reports domain-specific BLEU scores of 0.6-0.79 with edited system output as a reference (workshop communication, Maegaard & Offersgaard), and intends - like Dan2eng - to exploit dependency relations, making the system a good candidate for a future direct comparison in both methodological and performance terms.

7 Perspectives: Statistical smoothing

In spite of the fact that Dan2Eng employs tens of thousands of hand-written lexical transfer rules, it is extremely difficult to cover all idiosyncrasies of, for instance, preposition usage or choice of synonym in a rule based way. Furthermore, mismatches are more likely when chaining two translations. On the other hand, statistical methods allow to check the probabilities of rule-suggested translations in a given context, smoothing out translational rough spots. Given the lack of large bilingual Danish-English corpora, it is an added advantage that such methods work with *monolingual*, target language corpora - of which there are almost unlimited amounts available in the case of English. To prepare for an integration of TL smoothing, we performed dependency annotation of 1 billion words, and started extracting n-gram information as well as what we call dep-grams - hierarchical chains of dependency-linked words, the former with the perspective of preposition-smoothing, the latter for argument-smoothing.

Future evaluations, to be conducted after a trial and debugging period, should address not only the overall quality of the MT system as a whole, but also the rela-

tive contributions of rule based and statistical modules, as well as performance in different applicative contexts.

References

Bick, Eckhard & Ahlmann, Jens (2006). The Fyntour Multilingual Weather and Sea Dialogue System - Description and Assessment. In: Proceedings of DECALOG - The 2007 Workshop on the Semantics and Pragmatics of Dialogue, May 30 -1 June 2007. Roverto, Italy.

Bick, Eckhard & Nygaard, Lars (2007). Using Danish as a CG Interlingua: A Wide-Coverage Norwegian-English Machine Translation System. In: Proceedings of the 16th Nordic Conference of Computational Linguistics. May 24-26, 2007. Tartu, Estonia.

Bick, Eckhard (2003). A CG & PSG Hybrid Approach to Automatic Corpus Annotation. In: Kiril Simow & Petya Osenova (eds.), *Proceedings of SProLaC2003* (at Corpus Linguistics 2003, Lancaster), pp. 1-12

Bick, Eckhard (2005). Turning Constraint Grammar Data into Running Dependency Treebanks. In: Civit, M. & Kübler, S. & Martí, M. A. (red.), *Proceedings of TLT 2005, Barcelona, Dec 9th - 10th, 2005)*, pp.19-27

Koehn, Philipp (2005). Europarl: A Multilingual Corpus for the Evaluation of Machine Translation. MT Summit X, Sept. 12-16, 2005. Phuket, Thailand.

Maegaard, B. & V. Hansen (1995). PaTrans - Machine Translation of Patent Texts. From Research to Practical Application. In: *Convention Digest: Second Language Engineering Convention*, London, pp. 1-8.

Maegaard, B & L. Offersgaard (2007). *Statistisk maskinoversættelse (SDMT-SMV workshop).* 2007-01-18.

Papineni, Kishore et al. (2002). BLEU: A Method for Automatic Evaluation of Machine Translation. In: *Proceedings of the 40th ACL, Philadelphia, July 2000.* pp. 311-318

Snover, Matthew et al.. (2006). A Study of Translation Edit Rate with Targeted Human Annotation,. *Proceedings of Association for Machine Translation in the Americas.*