自然言語処理 126－3
(1998. 7. 23)

# ENHANCE OF MACHINE TRANSLATION WITH GDA-TAGS

ノイマン クリストフ
東京工業大学情報理工学部田中研究室
〒152-8552 東京都目黒区大岡山 2-12-1
TEL03(5734)2831
電子メール neumann@cs.titech.ac.jp

橋田浩一
電子技術総合研究所
〒305 つくば市梅園 1-1-4
0298-54-5928

**あらまし** パターンベース機械翻訳システム（PBMT）の原文書は、GDA(大域文書修飾)タグ付きである。その GDA タグは、自然言語処理のための曖昧性情報を表す。日英機械翻訳の 28 の主な曖昧性の問題に対して、トランスファーパターンルールを作成した。しかも、意味 Desambig.のため、メタ単語・ WWW 辞書も利用されている。L2 文書の生成もパターン式ルールを含めている。

パターンマッチングアルゴリズムは、Prolog で導出される節集合として実相されている。GDA タグ付きの文書の処理は、一つの例で説明する。

**キーワード** GDA、機械翻訳、ハイパーテクスト、PBMT、曖昧性, Prolog

# ENHANCE OF MACHINE TRANSLATION WITH GDA-TAGS

---

Christoph Neumann
Tokyo Institute of Technology
〒152-8552 Ookayama 2-12-1, Meguro, Tokyo
TEL03(5734)2831

Koiti Hasida
Electrotechnical Laboratory
1-1-4, Umezono, Tukuba, Ibaraki 305
0298-54-5928

eMail; neumann@cs.titech.ac.jp

**Abstract** The source text in the MT system comes with GDA tags, which are designed to provide disambiguation information needed for NLP applications. The tagged text matches, as a feature structure, patterns of structure transfer L1 – L2. For 28 ambiguity problems in Japanese – English MT, such transfer patterns were created. For further semantic disambiguation, "meta-words" and a WWW dictionary are used. Generation of actual L2-text involves again pattern matching rules. The pattern matching algorithm is implemented as a clause set, which is resolved in Prolog. The working strategy is explained through translating a multi-ambiguous example sentence. The system shows an unconventional way to MT and the feasibility of doing MT with GDA tags.

**Keywords** GDA, hypertext, machine translation, PBMT, Prolog

## 1. FIELD OF STUDY

One of the reasons why even advanced machine translation (MT) still fail so quickly, seems to be not any more the insufficiency of their L1-L2 mapping rules, but rather the insufficiency of the linguistic information on the source text. If the L1-parser would provide more information, the transfer systems would become more powerful.

It is desirable to make such hidden linguistic information overt by inserting it as tags into the L1-text. The "Global Document Annotation" (GDA) initiative [Hasida et al. 1997] provides such a tag set for encoding tagging semantic, syntactical and pragmatic features of electronic documents.

The main goal of this project is: Showing that MT of very high quality is possible, if the input comes tagged with all linguistic information needed. This implies to show the feasibility of doing MT with tags and also the adequacy of GDA-tags for MT needs.

## 2. THE GDA TAG SET

The GDA tag set builds formally on XML (like TEI, CES or EAGLES), but has a different aim: Mainly the ambiguous, hidden linguistic information is tagged in a code as language-independent as possible.

The tagging process is done by a tagging-editor program (TE) interacting with the author of the text. The TE includes a conventional parser plus access to an WWW-online dictionary (e.g. Wordnet). Unambiguous passages (i.e. mostly syntactical bracketing) are tagged automatically. When the TE discovers ambiguous passages, it prompts the user for resolution (in natural language) and encodes the answer in tags.

This tagging interaction user-TE is independent of the MT process itself, because GDA tags are designed for a general use by all NLP applications. But this *conditio sine qua non* puts the system basically in the range of Human-Aided Machine Translation (HAMT).

MT being a major NLP application, the use of the tags within MT was a decisive criterion for their definition. Based on two extensive failure surveys for Japanese-English MT software [ASCII 1996, Ajiataiheiyoukikaihon'yakukyokai 1994] and experiments with software for various language pairs, 22 failure patterns were designed.

These failure patterns are the base for the transfer patterns described below, and they lead to several proposals for GDA tag definition.

The currently 29 GDA-tags encode syntactic categories[1], include XML-standards, other units as <date>, <mentioned> and a number of "unit insertion" tags must be empty like <el> (ellipsis) or <pro> (pronoun). The core coding of linguistic information is done through the attributes, however.

A tagged text unit (called "tag") can be rewritten in a feature structure. Feature structures are also the data structure used in the system. The system uses feature structures in the form of Prolog predicates.

## 3. PATTERN-BASED MT

Pattern-based (also: case-based, example-based) machine translation systems are quite successful because linguistic information is not chaffed and synthesized again, but L1-rules can be directly rewritten in L2-rules [Watanabe 1995]. This "feature pattern rewriting" presents itself to be most promising to profit from the natural structure of tagged sentences.

20 of the 28 encoded transfer patterns are listed in the annex. #13 (Generic Nouns) shall illustrate a generation pattern.

Basically, a feature structure matching (=unifying with) a pattern, triggers and results in another feature structure, which is a modification of the source structure (at least one feature must be reduced or changed to avoid infinite loops - "downsizing"). The result structure itself then tries to unify with

[1] Phrasal and terminal categories are encoded with the same tag. (<n> can thus stand for N, N', ... or NP)

patterns which may lead to another result structure. Sub-structures (e.g., NPs within a VP) also trigger transfer patterns. [2]

The whole mechanism is implemented in Prolog, as a resolution of goals through sub-goals. Top-down: If a feature structure has "exhausted" all possible patterns, it is divided into its daughter patterns, who have their "trigger turn". The resolution process comes to a halt through default patterns (as the only Prolog facts) at the end of the clause set.

Combination of non-isomorphic rules is desired and possible through integration of general unifiers into daughter structures.[3] Preferences are encoded through the order of the Prolog clauses (of course, probabilistic reasoning is difficult).

**Translation process**

Parsing (analysis) is the first stage in most MT processes, but the system can practically skip this step as the source text comes already with all necessary linguistic information. The source text is only converted into a Prolog query.

Pattern transfer is the actual first step. The input feature structure is put again and again into the pattern matching process. The result still contains no actual L2-word, only a transformed structured (including re-ordering) and several "meta-words" representing linguistic feature bundles (^and, ^because). Some L1-words (e.g. functional particles) may actually have disappeared in the process.

Only now, the (remaining) L1-words are looked up in a bilingual dictionary in combination with the same WWW-dictionary to which the actual meaning of a homonym was already encoded by the TE as a reference. Now, this reference helps to decide on the L2 translation, if there are two or more different words.

In the generation, the feature structure and the meta-words trigger "generation" patterns. Afterwards, a conventional morpho-syntactical generator does the final polishing, such as re-ordering, agreement or pronominalisation.

## 4. EXAMPLE TRANSLATION

The system now translates a GDA-tagged example (sentence from [Ajiataiheiyoukikaihon'yakukyokai 1994]) from Japanese to English. It is a good occasion to show how several central themes in MT Japanese-English are treated within the system.

*Senzen, sengo-toiu-kotoba-ga wakai-mono-ni tsuuyou-shinai.*

(Pre-war, post-war-TO[4] say word-SUBJ young generation-DAT validity do-not)

**Transfer pattern matching**

In Prolog query form [?-translate(fig_1,Translation).], the tagged sentence (Fig. 1) results in Fig. 2 through these resolution patterns in the following order (with number of transfer type):

Subject identification in clause (#1)

Quoting nouns (#12)

Meta-language (#20)

Constituent order (#8)

Verb negation (#17)

Experiencer (#10)

Note that all particles (*to*,ga and *ni*) have vanished. *ni* and *to* were replaced by the meta-words indicating their deep-case content function. MT systems tend to stumble over *to* with its multiple functions (quote, time, condition, coordination). Here, it was replaced (together with *iu* "say") , which avoided triggering a risky disambiguation process.

---

[2] Superset-patterns are matched by subsets, e.g. in Japanese, <cs> „clause" matches <su> „sentential unit", <date> matches <n>

[3] [Watanabe 1995] develops a special „gluing" algorithm for rule combination.

[4] TO stands for the multi-functional particle *to* (see below)

<su subj=1><pp id=1><n sem=word0 rtyp=gn><v rel=cnt><n ctyp=disj><mentioned>***Senzen***</mentioned>, <mentioned>***sengo***</mentioned></n>***toiu***</v> ***kotoba***</n>***ga***</pp><v ctyp=fd><pp rel=exp><n rtyp=gn><aj>***wakai***</aj> <n sem=people0>***mono***</n></n>***ni***</pp><v><v sem=be_valid0>***tsuuyou-shi***</v><aj sem=not0>***nai***</aj></v></v></v>.</su>

Fig. 1 Tagged example sentence

<su subj=1><n id=1 sem=word0 rtyp=gn>***kotoba*** **^meaning** <n>**^mentioned** ***Senzen*** **^mentioned** **^or** ***sengo***</n></n> <v ctyp=bd><v sem=be_valid0> ***tsuuyou-shi*** **^not** </v><pp>**^experiencer**<n sem=people0 rtyp=gn><aj>***wakai***</aj><n>***mono***</n></n> </pp></v>.</su>

Fig. 2 Example sentence after transfer

*ga* 's subject designing function triggered a re-positioning of the nominal phrase. The structural ambiguous interpretation of the part *Senzen... wakai* (words ... are young) as a relative clause of *mono* was also intrinsically excluded by the assignation of the *ga* –NP as subject of the main (thus the only) clause.

The adjective-noun combination <n><aj>*wakai*</aj><n>*mono*</n>< /n> remains unchanged (triggers nothing) because in both Japanese and English, the canonical adjectival attribute position is in front of the noun. No additional en- and decoding is thus necessary. For French with adjectives after the noun, a re-ordering rule could be created:

<n><aj>A</aj><n>B</n></n>
← <n><n>B</n><aj>A</aj></n>

The assignment of a semantic content to <v sem=be_valid0>*tsuuyou-shi*</v> hinders the separate semantic analysis of *tsuuyou* ("validity") and *shi* (from *suru* "make"), which is undesired because *tsuuyou-suru* – like many Noun+*suru*-combinations - is translated by a verb construction not containing "make"..

**Dictionary consultation**

Now, the direct word mapping translation consisting of a simple dictionary look-up already produces Fig. 3

Note that the polysemy of *kotoba* (meaning "language" or "word") was disambiguated through the online dictionary by the *sem*-indication.

**Generation**

Finally, generation patterns are triggered by some remaining tags and the meta-words are mapped into English words:

<su subj=1><n id=1 sem=word0 rtyp=gn >***word*** **^meaning** <n> **^mentioned** ***pre-war*** ^or **^mentioned** ***post-war***</n> <v ctyp=bd><v>***be valid*** **^not**<pp>**^experiencer** <n rtyp=gn ><aj>***young***</aj><n sem=people0>***people***</n></n></pp></v></v>.</su>

(without tags) *word* ^meaning ^mentioned *pre-war* ^or ^mentioned *post-war be valid* ^not ^experiencer *young people.*

Fig 3. Example sentence after dictionary consultation

Generic nouns (#13) <n sem=X rtyp=gn>A</n> ← if X countable, use generic plural form (plural, no article in English) for A; if X uncountable, use partitive form of X (singular, no article in English)

Meta-words get an English equivalent:

^experiencer ←*to* (through case frame of verb )
^or ← *or* ; ,
^meaning ← *with the meaning*
^not ← *not*
^mentioned ← (punctuation - sets "" around next outer element)

The sentence looks now like this:

*words with the meaning "pre-war" or "post-war" be valid not to young people.*

Note article and number assignment, a main subject of MT research as Japanese nouns these features [Heine 1997]. Here, all four nouns were correctly synthesised in English. The two nouns with generic meaning got the respective morphemes expressing generic use in English. And the generator did not try to assign articles or number to "pre-war" or "post-war" as they were not marked as nouns, but as "mentioned" elements.

Only now, the conventional morpho-syntactical generator does the final "tidy-up" by assuring agreement and positioning of the negation of the predicate, resulting in

*words with the meaning "pre-war" or "post-war" are not valid to young people.*

## 5. STATISTICS

At present, 28 ambiguity problems of Japanese are encoded in transfer patterns to English as target language in SICStus Prolog. 6 exemplary sentences were tagged by hand (the TE is not yet available). We obtained the desired feature structure for all test sentences (precision: 100%); unrefined backtracking caused a bad recall rate (35,3%).

## 6. EVALUATION

The system seems unorthodox, because it "avoids" the challenge of searching for scientific disambiguation algorithms. Even the "difficult" language combination Japanese-English shows that MT based on GDA-tags is not only feasible, but can provide

high quality results. As indicated above for French, PaBMT with GDA tags is realisable for any language combination.

## 7. REFERENCES


Ajiataiheiyokikaihon'yakukyokai (1994), *Kenkyuseikahokokusho. Reibunhyokakenkyuka-hokokusho.* [Report on research results. Report of the workgroup on evaluation of example sentences]

ASCII (1996) *"Einichihon'yaku sofuto14pon"no hontono jitsuryoku.* [The true power of "14 English-Japanese translation software programmes"] ASCII, Vol. 20 (1996/7), pp. 400-413.

Dorr B. (1993), *Machine Translation. A View from the Lexicon.*

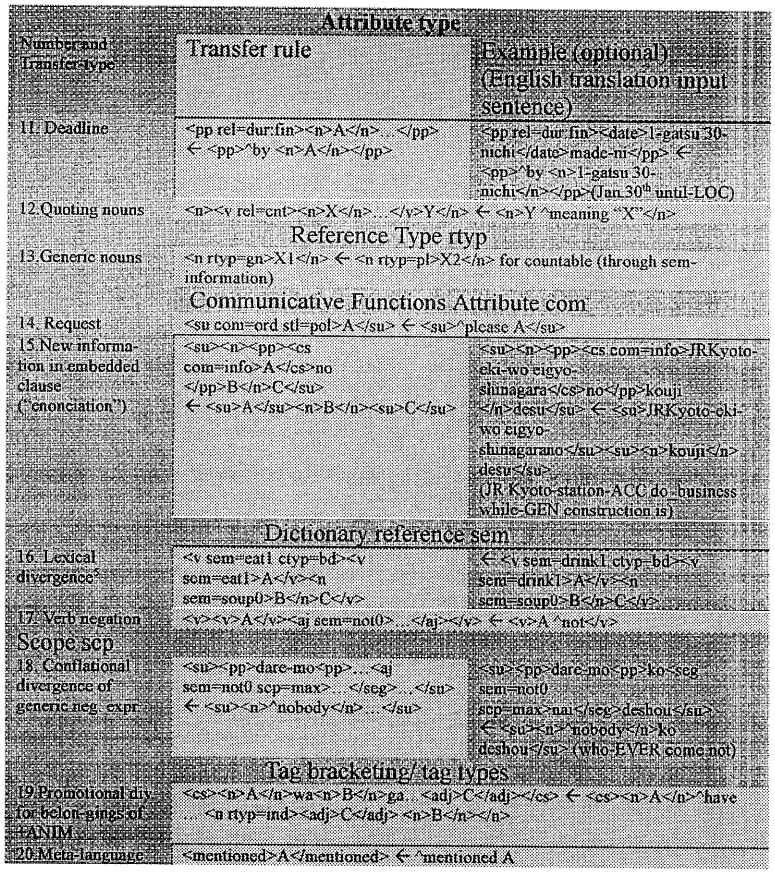
Heine J. (1997), *Ein Algorithmus zur Bestimmung der Definitheitswerte japanischer Nominalphrasen.* Diplomarbeit, Universität des Saarlandes

Hasida K., Sugimura R., Kashioka H., Uchiyama M. and Neumann C. (1997), *Taiikibunsyosyusyoku: Hyozyuntaguniyoru gengodetano daikibona kozokato sairiyo.* [Global Document Annotation: Structuring and Reusing of Linguistic Data through Standardised Tags] Proceedings of the Third Annual Meeting of The Association for Natural Language Processing (Kyoto), pp. 135-138

Watanabe H. (1995) *A Model of a Bi-Directional Transfer Mechanism Using Rule Combinations.* Machine Translation, 10, pp. 269-291.

# ANNEX: Transfer patterns Japanese - English

| Number and Transfer-type | Transfer rule | Example (optional) (English translation input sentence) |
|---|---|---|
| | **Attribute type** | |
| | **Referencing id, coid** | |
| 1.Subject identification in clause | <su subj=X>A<p id=X><n>Y</n>...<pp>B</su> ← <su subj=X>A<n id=X>Y</n>B</su> | |
| 2.Zero-Subject identification in co-text (outside clause) | <pp id=X><n>A</n></pp>...<su subj=X>B</su> ← <pp id=X><n>A</n></pp>...<su subj=Z><pro id=Z coid=X></pro>B</su> | <pp id=1><n>Tomita-san</n>to</pp>atta.<su subj=X>Nemu-sou datta.</su> ← <pp id=1><n>Tomita-san</n>to</pp>atta.<su subj=2><pro id=2 coid=1></pro>Nemu-sou datta.</su> (Tomita-Mr. TO met. Tired seem-PAST) |
| 3. Zero pronoun – referent only in pragmatic context | <su REL=p2>A</su> ← <su REL=X><pro id=X>*you</pro>A</su> | <su rel=p2>Shitsukoi!</su> ← <su REL=1><pro id=1>*you</pro>Shitsukoi!</su> (Be_annoying.) |
| | **Construction type ctyp** | |
| 4. Appositon (with no) | <nctyp=fa><pp><n>A</n>no</pp>...<n>B</n> </n> ← <n ctyp=ba><n>B</n><seg>A</seg></n> | <n ctyp=fa><pp><n>sensei</n>no</pp><n>Suzuki-san</n></n> ← <n ctyp=ba><n> Suzuki-san </n><seg>sensei</seg></n> (teacher-GEN Suzuki-Mr.) |
| 5. Coordination | <nctyp=p><n>X1</n>..<n>X2</n>..<n>Xn</n></n> ← <n rtyp=pl><n>X1</n>*and <n> X2 </n>...*and <n> Xn</n></n> | <n ctyp=p><n>Senzen</n> <n>sengo</n> ← <n><n>Senzen</n>*and<n>sengo</n> (Post-war, pre-war) |
| 6. Disjunction | <n ctyp=disj><n>X1</n>...<n>X2</n>...<n>Xn</n></n> | ← <n rtyp=pl><n>X1</n>*or<n> X2 </n>...*or <n> Xn</n></n> |
| 7. Relatives | <n ctyp=fr><cs Cat_X=X>A</cs><n id=X>B</n></n> ← <n ctyp=br><n id=X>B</n><cs><relpro rel=Cat_X coid=X></relpro>A</cs></n> | <n ctyp=fr><cs loc=87>Kare-ga umareta</cs><n id=87>Toshu</n></n>-wa Toukyou desu ← <n ctyp=br><n id=87>Toshi</n><cs><relpro rel=loc coid=87></relpro>Kare-ga umareta</cs></n>-wa Toukyou desu (He-SUBJ was born-city-THEME Tokyo is.) |
| 8. Constituent order | <vctyp=fd><pp>X1</pp>...<pp>X2</pp>...<pp>Xn<pp>Y</v> ← <v ctyp=bd>Y<pp>X1</pp>...<pp>X2</pp>...<pp>Xn<pp></v> | <v ctyp=fd><pp>daigaku-e<pp>iku</v> ← <v ctyp=bd>iku<pp>daigaku-e<pp><v> (university-DIRECTION go) |
| | **Relations (Semantic, Syntactic, Pragmatic) rel** | |
| 9.Condition | <pp rel=cnd><n>A</n>...</pp> ← <pp>*if <n>A</n></pp> | <cs rel=cnd><cs>anata-ga iku</cs>to</cs> ← <cs>*if<cs>anata-ga iku </cs>...</cs> (you-SUBJ go-TO) |
| 10. Experiencer | <pp rel=exp><n>A</n>...</pp> ← <pp>*experiencer <n>A</n></pp> | |




[5] The divergence types are from [Dorr 1993]