

Annotation Compatibility Working Group Report*

Contributors: A. Meyers, A. C. Fang, L. Ferro, S. Kübler, T. Jia-Lin, M. Palmer, M. Poesio, A. Dolbey, K. K. Schuler, E. Loper, H. Zinsmeister, G. Penn, N. Xue, E. Hinrichs, J. Wiebe, J. Pustejovsky, D. Farwell, E. Hajicova, B. Dorr, E. Hovy, B. A. Onyshkevych, L. Levin
Editor: A Meyers meyers@cs.nyu.edu

*As this report is a compilation, some sections may not reflect the views of individual contributors.

Abstract

This report explores the question of compatibility between annotation projects including translating annotation formalisms to each other or to common forms. Compatibility issues are crucial for systems that use the results of multiple annotation projects. We hope that this report will begin a concerted effort in the field to track the compatibility of annotation schemes for part of speech tagging, time annotation, treebanking, role labeling and other phenomena.

1. Introduction

Different corpus annotation projects are driven by different goals, are applied to different types of data (different genres, different languages, etc.) and are created by people with different intellectual backgrounds. As a result of these and other factors, different annotation efforts make different underlying theoretical assumptions. Thus, no annotation project is really theory-neutral, and in fact, none should be. It is the theoretical concerns which make it possible to write the specifications for an annotation project and which cause the resulting annotation to be consistent and thus usable for various natural language processing (NLP) applications. Of course the theories chosen for annotation projects tend to be theories that are useful for NLP. They place a high value on descriptive adequacy (they cover the data), they are formalized sufficiently for consistent annotation to be possible, and they tend to share major theoretical assumptions with other annotation efforts, e.g., the noun is the head of the noun phrase, the verb is the head of the sentence, etc. Thus the term *theory-neutral* is often used to mean something like *NLP-friendly*. Obviously, the annotation compatibility problem that we address here is much simpler than it would be if we had to consider theories which

place a low emphasis on NLP-friendly properties (Minimalism, Optimality Theory, etc.).

As annotation projects are usually research efforts, the inherent theoretical differences may be viewed as part of a search for the truth and the enforcement of adherence to a given (potentially wrong) theory could hamper this search. In addition, annotation of particular phenomena may be simplified by making theoretical assumptions conducive to describing those phenomena. For example, relative pronouns (e.g., *that* in the NP *the book that she read*) may be viewed as pronouns in an anaphora annotation project, but as intermediate links to arguments for a study of predicate argument structure.

On the other hand, many applications would benefit by merging the results of different annotation projects. Thus, differences between annotation projects may be viewed as obstacles. For example, combining two or more corpora annotated with the same information may improve a system (i.e., "there's no data like more data.") To accomplish this, it may be necessary to convert corpora annotated according to one set of specifications into a different system or to convert two annotation systems into a third system. For example, to obtain lots of part of speech data for English, it is advantageous to convert POS tags from several tagsets (see Section 2) into a common form. For more temporal data than is available in Timex3 format, one might have to convert Timex2 and Timex3 tags into a common form (See Section 5). Compromises that do not involve conversion can be flawed. For example, a machine learner may determine that feature A in framework 1 predicts feature A' in framework 2. However, the system may miss that features A and B in framework 1 actually both correspond to feature A', i.e., they are subtypes. In our view, directly modeling the parameters of compatibility would be preferable.

Some researchers have attempted to combine a number of different resource annotations into a single merged form. One motivation is that the merged representation may be more than the sum of its parts. It is likely that inconsistencies and errors (often induced by task-specific biases) can be identified and adjusted in the merging process; inferences may be drawn from how the component annotation systems interact; a complex annotation in a single framework may be easier for a system to process than several annotations in different frameworks; and a merged framework will help guide further annotation research (Pustojevsky, et. al. 2005). Another reason to merge is that a merged resource in language *A* may be similar to an existing resource in language *B*. Thus merging resources may present opportunities for constructing nearly parallel resources, which in turn could prove useful for a multilingual application. Merging PropBank (Kingsbury, and Palmer 2002) and NomBank (Meyers, et. al. 2004) would yield a predicate argument structure for nouns and verbs, carrying more similar information to the Prague Dependency TreeBank's TectoGrammatical structure (Hajicova and Ceplova, 2000) than either component.

This report and an expanded online version <http://nlp.cs.nyu.edu/wiki/corpuswg/AnnotationCompatibility> both describe how to find correspondences between annotation frameworks. This information can be used to combine various annotation resources in different ways, according to one's research goals, and, perhaps, could lead to some standards for combining annotation. This report will outline some of our initial findings in this effort with an eye towards maintaining and updating the online version in the future. We hope this is a step towards making it easier for systems to use multiple annotation resources.

2. Part of Speech and Phrasal Categories

On our website, we provide correspondences among a number of different part of speech tagsets in a version of the table from pp. 141--142 of Manning and Schütze (1999), modified to include the POS classes from CLAWS1 and ICE. Table 1 is a sample taken from this table for expository purposes (the full table is not provided due to space limitations). Traditionally,

part of speech represents a fairly coarse-grained division among types of words, usually distinguishing among: nouns, verbs, adjectives, adverbs, determiners and possibly a few other classes. While part of speech classifications may vary for particular words, especially closed class items, we have observed a larger problem. Most part of speech annotation projects incorporate other distinctions into part of speech classification. Furthermore, they incorporate different types of distinctions. As a result, conversion between one tagset and another is rarely one to one. It can, in fact, be many to many, e.g., BROWN does not distinguish the

Table 1: Part of Speech Compatibility

Table 1: Part of Speech Compatibility					
Extending Manning and Schütze 1999, pp. 141-142, to cover Claws1 and ICE -- Longer Version Online					
Class	Wrds	Claws c5, Claws1	Brown	PTB	ICE
Adj	Happy, bad	AJ0	JJ	JJ	ADJ.ge
Adj, comp	happier, worse	AJC	JJR	JJR	ADJ.comp
Adj, super	nicest, worst	AJS	JJT	JJS	ADJ.sup
Adj, past part	eaten	JJ	??	VCN, JJ	ADJ.edp
Adj, pres part	calming	JJ	??	VBG, JJ	ADJ.ingp
Adv	slowly, sweetly	AV0	RB	RB	ADV.ge
Adv comp	faster	AV0	RBR	RBR	ADV.comp
Adv super	fastest	AV0	RBT	RBS	ADV.sup
Adv Part	up, off, out	AVP, RP, RI	RP	RP	ADV.{phras, ge}
Conj coord	and, or	CJC, CC	CC	CC	CON-JUNC.

					coord
Det	this, each, another	DT0, DT	DT	DT	PRON. dem. sing, PRON (recip)
Det. pron	any, some	DT0, DTI	DT1	DT	PRON. nonass, PRON. ass
Det pron Plur	these those	DT0, DTS	DTS	DT	PRON. dem. plu
Det preq	quite	DT0, aBL	ABL	PDT	ADV .intens
Det preq	all, half	DT0, ABN	ABN	PDT	PRON. univ, PRON. quant
Noun	air-craft, data	NN0	NN	NN	N.com. sing
Noun sing	cat, pen	NN1	NN	NN	N.com. sing
Noun plur	cats, pens	NN2	NNS	NNS	N.com. plu
Noun prop sing	Paris, Mike	NP0	NP	NNP	N.prop .sing
Verb. base pres	take, live	VVB	VB	VBP	V.X. {pres, imp}
Verb, infin	take, live	VVI	VB	VB	V.X. infin
Verb, past	took, lived	VVD	VBD	VBD	V.X. past
Verb, pres part	tak-ing, liv-ing	VVG	VBG	VBG	V.X. ingp
Verb, past-part	taken, lived	VVN	VCN	VCN	V.X. edp
Verb, pres	takes,	VVZ	VBZ	VBZ	V.X. pres

--	--	--	--	--	--

infinitive form of a verb (VB in the Penn Treebank, V.X.infin in ICE) from the present-tense form (VBP in the Penn Treebank, V.X.pres in ICE) that has the same spelling (e.g., *see* in *They see no reason to leave*). In contrast, ICE distinguishes among several different subcategories of verb (cop, intr, cxt, dimontr, ditr, montr and TRANS) and the Penn Treebank does not.¹ In a hypothetical system which merges all the different POS tagsets, it would be advantageous to factor out different types of features (similar to ICE), but include all the distinctions made by all the tag sets. For example, if a token *give* is tagged as VBP in the Penn Treebank, VBP would be converted into VERB.anysubc.pres. If another token *give* was tagged VB in Brown, VB would be converted to VERB.anysubc{infin,n3pres} (n3pres = not-3rd-person and present tense). This allows systems to acquire the maximum information from corpora, tagged by different research groups.

CKIP Chinese-Treebank (CCTB) and Penn Chinese Treebank (PCTB) are two important resources for Treebank-derived Chinese NLP tasks (CKIP, 1995; Xia et al., 2000; Xu et al., 2002; Li et al., 2004). CCTB is developed in traditional Chinese (BIG5-encoded) at the Academia Sinica, Taiwan (Chen et al., 1999; Chen et al., 2003). CCTB uses the Information-based Case Grammar (ICG) framework to express both syntactic and semantic descriptions. The present version CCTB3 (Version 3) provides 61,087 Chinese sentences, 361,834 words and 6 files that are bracketed and post-edited by humans based on a 5-million-word tagged Sinica Corpus (CKIP, 1995). CKIP POS tagging is a hierarchical system. The first POS layers include eight main syntactic categories, i.e. **N** (noun), **V** (verb), **D** (adverb), **A** (adjective), **C** (conjunction), **I** (interjection), **T** (particles) and **P** (preposition). In CCTB, there are 6 non-terminal phrasal categories: **S** (a complete tree headed by a predicate), **VP** (a phrase headed by a predicate), **NP** (a phrase headed by an N), **GP** (a phrase headed by locational noun or adjunct), **PP**

¹ In the ICE column of Table 1 X represents a the disjunction of verb subcategorization types {cop, intr, cxt, dimontr, ditr, montr, trans}.

(a phrase headed by a preposition) and **XP** (a conjunctive phrase that is headed by a conjunction).

Exam- ples	Top Layer (TL)		Bottom Layer (BL)	
	PCTB	CCTB	PCTB	CCTB
換句 話說 in other words	ADVP	Head	AD	Dk
於是 there- fore	ADVP	result	AD	Cbca
因為 be- cause	P	reason	P	Cbaa
過去 past	NP- TMP	time:NP	NT	Ndda
去年 last year	NP- TMP	NP	NT	Ndaba
其中 among	NP- ADV	NP	NN	Nep
同樣 地 also	DVP	ADV	AD:DEV	Dk
近年 來 in the last few years	LCP- TMP	GP	NT:LCG P	Nddc:N g

PCTB annotates simplified Chinese texts (GB-encoded) from newswire sources (Xinhua newswire, Hong Kong news and Sinorama news magazine, Taiwan). It is developed at the University of Pennsylvania (UPenn). The PCTB annotates Chinese texts with syntactic bracketing, part of speech information, empty categories and function tags (Xia et al, 2000, 2002, 2005). The predicate-argument structure of Chinese verbs for the PCTB is encoded in the Penn Chinese Proposition Bank (Xue, et. Al. 2005). The present version PCTB5.1 (PCTB

Version 5.1), contains 18,782 sentences, 507,222 words, 824,983 Hanzi and 890 data files.

PCTB's bracketing annotation is in the same framework as other Penn Treebanks, bearing a loose connection to the Government and Binding Theory paradigm. The PCTB annotation scheme involves 33 POS-tags, 17 phrasal tags, 6 verb compound tags, 7 empty category tags and 26 functional tags.

Table 2 includes Top-Layer/Bottom-Layer POS and phrasal categories correspondences between PCTB4 and CCTB3 for words/phrases expressed as the same Chinese characters in the same order.

3. Differences Between Frameworks

We assume that certain high level differences between annotation schemata should be ignored if at all possible, namely those that represent differences of analyses that are notationally equivalent. In this section, we will discuss those sorts of differences with an eye towards evaluating whether real differences do in fact exist, so that way users of annotation can be careful should these differences be of significance to their particular application.

To clarify, we are talking about the sort of high level differences which reflect differences in the linguistic framework used for representing annotation, e.g., many frameworks represent long distance dependencies in equivalent, but different ways. In this sense, the linguistic framework of the Penn Treebank is a phrase structure based framework that includes a particular set of node labels (POS tags, phrasal categories, etc.), function tags, indices, etc.².

3.1 Dependency vs. Constituency

Figure 1 is a candidate rule for converting a phrase structure tree to a dependency tree or vice versa. Given a phrase consisting of constituents $C(n-i)$ to $C(n+j)$, the rule assumes that: there is one unique constituent $C(n)$ that is the head of the phrase; and it is possible to identify this head in the phrase structure grammar, either using a reliable heuristic or due to annotation that marks the head of the phrase. When converting the

² Linguistic frameworks are independent of encoding systems, e.g., Penn Treebank's inline LISP-ish notation, can be converted to inline XML, offset annotation, etc., Such encoding differences are outside the scope of this report

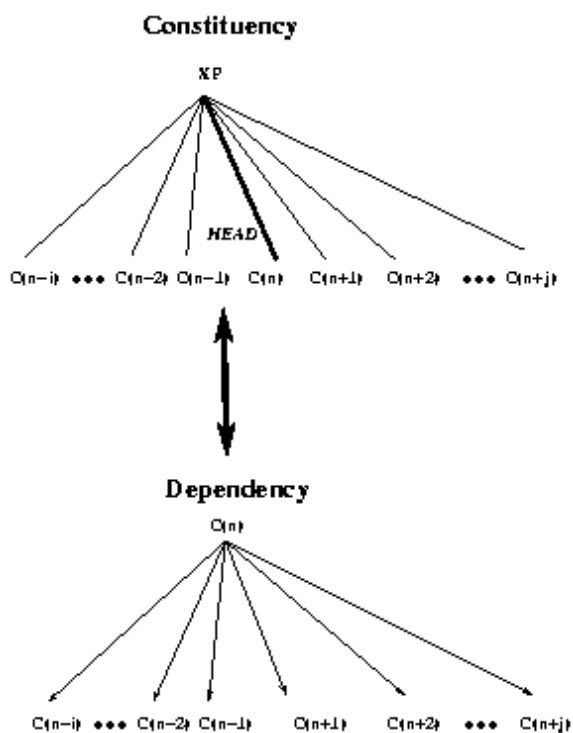


Fig. 1: Candidate Constituency/Dependency Mapping

phrase structure tree to the dependency tree, the rule promotes the head to the root of the tree. When converting a dependency tree to a phrase structure tree, the rule demotes the root to a constituent, possibly marking it as the head, and names the phrase based on the head's part of speech, e.g., nouns are heads of NPs. This rule is insufficient because: (1) Identifying the head of a phrase by heuristics is not 100% reliable and most phrase structure annotation does not include a marking for the head; (2) Some phrase structure distinctions do not translate well to some Dependency Grammars, e.g., the VP analysis and nestings of prenominal modifiers³; and (3) The rule only works for phrases that fit the head plus modifiers pattern and many phrases do not fit this pattern (uncontroversially).

While most assume that verbs act like the head of the sentence, a Subject + VP analysis of a sentence complicates this slightly. Regarding S-bars (relative clauses, that-S, subordinate-conjunction + S, etc.), there is some variation

³ The Prague Dependency Treebank orders dependency branches from the same head to represent the scope of the dependencies. Applicative Universal Grammar (Shayman 1977) incorporates phrases into dependency structure.

among theories whether the verb or the pre-S element (that, subordinate conjunction, etc.) is assigned the *head* label. Names, Dates, and other "patterned" phrases don't seem to have a unique head. Rather there are sets of constituents which together act like the head. For example, in *Dr. Mary Smith*, the string *Mary Smith* acts like the head. Idioms are big can of worms. Their headedness properties vary quite a bit. Sometimes they act like normal headed phrases and sometimes they don't. For example, the phrase *pull strings for John* obeys all the rules of English that would be expected of a verb phrase that consists of a verb, an NP and a complement PP. In contrast, the phrase *let alone* (Fillmore, et al. 1988) has a syntax unique to that phrase. Semi-idiomatic constructions (including phrasal verbs, complex prepositions, etc.) raise some of the same questions as idioms. While making headedness assumptions similar to other phrases is relatively harmless, there is some variation. For example, in the phrase *Mary called Fred up on the phone*, there are two common views: (a) *called* is the head of the VP (or S) and *up* is a particle that depends on *called*; and (b) the VP has a complex head *called up*. For most purposes, the choice between these two analyses is arbitrary. Coordinate structures also require different treatment from head + modifier phrases -- there are multiple head-like constituents.

A crucial factor is that the notion *head* is used to represent different things. (cf. Corbett, et. al. 1993, Meyers 1995). However, there are two dominant notions. The first we will call the *functor* (following Categorical Grammar). The functor is the glue that holds the phrase together -- the word that selects for the other words, determines word order, licenses the construction, etc. For example, coordinate conjunctions are functors because they link the constituents in their phrase together. The second head like notion we will call the *thematic head*, the word or words that determine the external selectional properties of the phrase and usually the phrasal category as well. For example, in the noun phrase *the book and the rock*, the conjunction *and* is the functor, but the nouns *book* and *rock* are thematic heads. The phrase is a concrete noun phrase due to *book* and *rock*. Thus the following sentence is well-formed: *I held the book and the rock*, but the following sentence is ill-formed **I held the noise and the redness*. Furthermore, the phrase *the book and the rock* is a noun phrase, not a conjunction phrase.

In summary, there are some differences between phrase structure and dependency analyses which may be lost in translation, e.g., dependency analyses include head-marking by default and phrase structure analyses do not. On the other hand, phrase structure analyses include relations between groupings of words which may not always be preserved when translating to dependencies. Moreover, both identifying heads and combining words into phrases have their own sets of problems which can come to the forefront when translation between the two modalities is attempted. To be descriptively adequate, frameworks that mark heads do deal with these issues. The problem is that they are dealt with in different ways across dependency frameworks and across those phrase structure frameworks where heads are marked. For example, conjunction may be handled as being a distinct phenomenon (another dimension) that can be filtered through to the real heads. Alternatively, a head is selected on theoretical or heuristic grounds (the head of the first the conjunct, the conjunction, etc.) When working with multiple frameworks, a user must adjust for the assumptions of each framework.

3.2 Gap Filling Mechanisms

It is well-known that there are several equivalent ways to represent long distance and lexically based dependencies, e.g., (Sag and Fodor, 1994). Re-entrant graphs (graphs with shared structure), empty category/antecedent pairs, representations of discontinuous constituents, among other mechanisms can all be used to represent that there is some relation **R** between two (or more) elements in a linguistic structure that is, in some sense, noncanonical. The link by any of these mechanisms can be used to show that the relation **R** holds in spite of violations of a proximity constraint (long distance dependencies), a special construction such as control, or many other conditions. Some examples follow:

1. *What_i did you read e_i? (WH extraction)*
2. *The terrorist_i was captured e_i (Passive)*
3. *I_i wanted e_i to accept it. (Control)*

It seems to us that the same types of cases are difficult for all such approaches. In the unproblematic cases, there is a gap (or equivalent) with a unique filler found in the same sentence. In the "difficult" cases, this does not

hold. In some examples, the filler is hypothetical and should be interpreted something like the pronoun *anyone* (4 below) or the person being addressed (5). In other examples, the identity between filler and gap is not so straight-forward. In examples like (6), filler and gap are type identical, not token identical (they represent different *reading* events). In examples like (7), a gap can take split antecedents. Conventional filler/gap mechanisms of all types have to be modified to handle these types of examples.

4. *They explained how e to drive the car*
5. *e don't talk to me!*
6. *Sally [read a linguistics article]_i, but John didn't e_i.*
7. *Sally_i spoke with John_j about e_{i,j}, leaving together.*

3.3 Coreference and Anaphora

There is little agreement concerning coreference annotation in the research community. Funding for the creation of the existing anaphorically annotated corpora (MUC6/7, ACE) has come primarily from initiatives focused on specific application tasks, resulting in task-oriented annotation schemes. On the other hand, a few (typically smaller) corpora have also been created to be consistent with existing, highly developed theoretical accounts of anaphora from a linguistic perspective. Accordingly, many schemes for annotating coreference or anaphora have been proposed, differing significantly with respect to: (1) the task definition, i.e., what type of semantic relations are annotated; (2) the flexibility that annotators have.

By far the best known and most used scheme is that originally proposed for MUC 6 and later adapted for ACE. This scheme was developed to support information extraction and its primary aim is to identify all mentions of the same objects in the text ('coreference') so as to collect all predications about them. A <coref> element is used to identify mentions of objects (the MARKABLES); each markable is given an index; subsequent mentions of already introduced objects are indicated by means of the REF attribute, which specifies the index of the previous mention of the same object. For example, in (1), markable 10 is a mention of the same object as markable 9. (This example is adapted from a presentation by Jutta Jaeger.)

1. `<coref id="9">The New Orleans Oil and Gas [...] company</coref> added that <coref id="10" type="ident" ref="9"> it</coref> doesn't expect [...].`

The purpose of the annotation is to support information extraction. To increase coding reliability, the MUC scheme conflates different semantic relations into a single IDENT relation. For example, coders marked pairs of NPs as standing in IDENT relations, even when these NPs would more normally be assumed to be in a predication relations, e.g., appositions as in 2 and NPs across a copula as in 3. This conflation of semantic relations is a convenient simplification in many cases but it is untenable in general, as discussed by van Deemter & Kibble (2001).

2. *Michael H. Jordan, the former head of Pepsico's international operations*
3. *Michael H. Jordan is the former head of Pepsico's international operations*

From the point of view of markup technology, the way used to represent coreference relations in MUC is very restrictive. Only one type of link can be annotated at a time: i.e., it is not possible to identify a markable as being both a mention of a previously introduced referent and as a bridging reference on a second referent. In addition, the annotators do not have the option to mark anaphoric expressions as ambiguous.

The MATE 'meta-scheme' (Poesio, 1999) was proposed as a very general repertoire of markup elements that could be used to implement a variety of existing coreference schemes, such as MUC or the [MapTask](#) scheme, but also more linguistically motivated schemes. From the point of view of markup technology, the two crucial differences from the MUC markup method are that the MATE meta-scheme is (i) based on standoff technology, and, most relevant for what follows, (ii) follows the recommendations of the Text Encoding Initiative (TEI) which suggest separating relations ('LINKs') from markables. LINKs can be used to annotate any form of semantic relations (indeed, the same notion was used in the TimeML annotation of temporal relations). A *structured* link, an innovation of MATE, can represent ambiguity (Poesio & Artstein, 2005). In (4), for example, the antecedent of the pronoun realized by markable ne03 in utterance 3.3 could be either *engine E2*

or *the boxcar at Elmira*; with the MATE scheme, coders can mark their uncertainty.

4. *[in file markable.xml]*

3.3: hook `<COREF:DE ID="ne01">engine E2</COREF:DE>` to `<COREF:DE ID="ne02">the boxcar at ... Elmira </COREF:DE>`

5.1: and send `<COREF:DE ID="ne03">it</COREF:DE>` to `<COREF:DE ID="ne04">Corning</COREF:DE>`

[in a separate file – e.g., link.xml]

```
<COREF:LINK HREF=
"markable.xml#id(ne03)" type="ident">
<COREF:ANCHOR HREF=
"markable.xml#id(ne01)" />
<COREF:ANCHOR HREF=
"markable.xml#id(ne02)" />
</COREF:LINK>
```

The MATE meta-scheme also allowed a richer set of semantic relations in addition to IDENT, including PART-OF, PRED for predicates, etc., as well as methods for marking antecedents not explicitly introduced via an NP, such as plans and propositions. Of course, using this added power is only sensible when accompanied by experimentally tested coding schemes.

The MATE meta-scheme was the starting point for the coding scheme used in the GNOME project (Poesio 2004). In this project, a scheme was developed to model anaphoric relations in text in the linguistic sense—e.g., the information about discourse entities and their semantic relations expressed by the text. A relation called IDENT was included, but it was only used to mark the relation between mentions of the same discourse entity; so, for example, neither of the relations in (2) would be marked in this way.

From the point of view of coding schemes used for resource creation, the MATE meta-scheme gave rise to two developments: the standoff representation used in the MMAX annotation tool, and the Reference Annotation Framework (Salmon-Alt & Romary, 2004). MMAX was the first usable annotation tool for standoff annotation of coreference (there are now at least three alternatives: Penn's [WordFreak](#), MITRE's CALISTO, and the NITE XML tools). The markup scheme was a simplification of the MATE scheme, in several respects. First of all, cross-level reference is not done using href links,

but by specifying once and for all which files contain the base level and which files contain each level of representation; each level points to the same base level. Secondly, markables and coref links are contained in the same file.

5. [markable file]

```
<?xml version="1.0"?> <markables> .....  
<markable id="markable_36" span=  
"word_5,word_6, word_7"member="set_22" >  
</markable> ... <markable id="markable_37"  
span="word_14, word_15, word_16"  
member="set_22" > </markable> ....  
</markables>
```

The original version of MMAX, 0.94, only allowed to specify one identity link and one bridging reference per markable, but beginning version 2.0, multiple pointers are possible. An interesting aspect of the proposal is that identity links are represented by specifying membership to coreference chains instead of linking to previous mentions. Multiple pointers were used in the ARRAU project to represent ambiguous links, with some restrictions. The RAF framework was proposed not to directly support annotation, but as a rich enough markup framework to be used for annotation exchange.

3.3.2 Conversion

Several types of conversion between formats for representing coreference information are routinely performed. Perhaps the most common problem is to convert between inline formats used for different corpora: e.g., to convert the MUC6 corpus into GNOME. However, it is becoming more and more necessary to convert standoff into inline formats for processing (e.g., MMAX into MAS-XML), and viceversa.

The increasing adoption of XML as a standard has made the technical aspect of conversion relatively straightforward, provided that the same information can be encoded. For example, because the GNOME format is richer than both the MUC and MMAX format, it should be straightforward to convert a MUC link into a GNOME link. However, the correctness of the conversion can only be ensured if the same coding instructions were followed; the MUC IDENT links used in (2) and (3) would not be expressed in the GNOME format as IDENT links. There is no standard method we know of

for identifying these problematic links, although syntactic information can sometimes help. The opposite of course is not true; there is no direct way of representing the information in (4) in the MUC format. Conversion between the MAS-XML and the MMAX format is also possible, provided that pointers are used to represent both bridging references and identity links.

4 Predicate-Argument Relations

Predicate argument relations are labeled relations between two words/phrases of a linguistic description such that one is a semantic predicate or functor and the other is an argument of this predicate. In the sentence *The eminent linguist read the book*, there is a SUBJECT (or AGENT, READER, ARG0, DEPENDENT etc.) relation between the functor *read* and the phrase *The eminent linguist* or possibly the word *linguist* if assuming a dependency framework. Typically, the functor imposes selectional restrictions on the argument. The functor may impose word order restrictions as well, although this would only effect "local" arguments (e.g., not arguments related by WH extraction). Another popular way of expressing this relation is to say that *read* assigns the SUBJECT role to *The eminent linguist* in that sentence. Unfortunately, this way of stating the relation sometimes gives the false impression that a particular phrase can only be a member of one such relation. However, this is clearly not the case, e.g., in *The eminent linguist who John admires read the book*, *The eminent linguist* is the argument of: (1) a SUBJECT relation with *read* and an OBJECT relation with *admires*. Predicate-argument roles label relations between items and are not simply tags on phrases (like Named Entity Tags, for example).

There are several reasons why predicate argument relations are of interest for natural language processing, but perhaps the most basic reason is that they provide a way to factor out the common meanings from equivalent or nearly equivalent utterances. For example, most systems would represent the relation between *Mary* and *eat* in much the same way in the sentences: *Mary ate the sandwich*, *The sandwich was eaten by Mary*, and *Mary wanted to eat the sandwich*. Crucially, the shared aspect of meaning can be modeled as a relation with *eat* (or *ate*) as the functor and *Mary* as the argument (e.g., SUBJECT). Thus providing predicate

argument relations can provide a way to generalize over data and, perhaps, allow systems to mitigate against the sparse data problem.

Systems for representing predicate argument relations vary drastically in granularity. In particular, there is a long history of disagreement about the appropriate level of granularity of role labeling, the tags used to distinguish between predicate argument relations. At one extreme, no distinction is made between predicate relations, one simply marks that the functor and argument are in a predicate-argument relation (e.g., unlabeled dependency trees). In another approach, one might distinguish among the arguments of each predicate with a small set of labels, sometimes numbered -- examples of this approach include Relational Grammar (Perlmutter 1984), PropBank and NomBank. These labels have different meanings for each functor, e.g., the subject of *eat*, *write* and *devour* are distinct. This assumes a very high level of granularity, i.e., there are several times the number of possible relations as there are distinct functors. So 1000 verbs may license as many as 5000 distinct relations. Under other approaches, a small set of relation types are generalized across functors. For example, under Relational Grammar's Universal Alignment Hypothesis (Perlmutter and Postal 1984, Rosen 1984), subject, object and indirect object relations are assumed to be of the same types regardless of verb. These terms thus are fairly coarse-grained distinctions between types of predicate/argument relations between verbs and their arguments.

Some predicate-neutral relations are more fine grained, including Panini's Karaka of 2000 years ago, and many of the more recent systems which make distinctions such as agent, patient, theme, recipient, etc. (Gruber 1965, Fillmore 1968, Jackendoff 1972). The (current) International Annotation of Multilingual Text Corpora project (<http://aitc.aitc.net.org/nsf/iamtc/>) takes this approach. Critics claim that it can be difficult to maintain consistency across predicates with these systems without constantly increasing the inventory of role labels to describe idiosyncratic relations, e.g., the relation between the verbs *multiply*, *conjugate*, and their objects. For example, only a very idiosyncratic classification could capture the fact that only a large round object (like the Earth) can be the object of *circumnavigate*. It can also be unclear which of two role labels apply. For example, there can be

a thin line between a recipient and a goal, e.g., the prepositional object of *John sent a letter to the Hospital* could take one role or the other depending on a fairly subtle ambiguity.

To avoid these problems, some annotation research (and some linguistic theories) has abandoned predicate-neutral approaches, in favor of the approaches that define predicate relations on a predicate by predicate basis. Furthermore, various balances have been attempted to solve some of the problems of the predicate-neutral relations. FrameNet defines roles on a scenario by scenario basis, which limits the growth of the inventory of relation labels and insures consistency within semantic domains or frames. On the other hand, the predicate-by-predicate approach is arguably less informative than the predicate-neutral approach, allowing for no generalization of roles across predicates. Thus although PropBank/NomBank use a strictly predicate by predicate approach, there have been some attempts to regularize the numbering for semantically related predicates. Furthermore, the descriptors used by the annotators to define roles can sometimes be used to help make finer distinctions (descriptors often include familiar role labels like agent, patient, etc.)

The diversity of predicate argument labeling systems and the large inventory of possible role labels make it difficult to provide a simple mapping (like Table 1 for part of speech conversion) between these types of systems. The SemLink project provides some insight into how this mapping problem can be solved.

4.2 SemLink

SemLink is a project to link the lexical resources of FrameNet, PropBank, and VerbNet. The goal is to develop computationally explicit connections between these resources combining individual advantages and overcoming their limitations.

4.2.1 Background

VerbNet consists of hierarchies of verb classes, extended from those of Levin 1993. Each class and subclass is characterized extensionally by its set of verbs, and intensionally by argument lists and syntactic/semantic features of verbs. The full argument list consists of 23 thematic roles, and

possible selectional restrictions on the arguments are expressed using binary predicates. VerbNet has been extended from the Levin classes, and now covers 4526 senses for 3175 lexemes. A primary emphasis for VerbNet is grouping verbs into classes with coherent syntactic and semantic characterizations in order to facilitate acquisition of new class members based on observable syntactic/semantic behavior. The hierarchical structure and small number of thematic roles is intended to support generalizations.

FrameNet consists of collections of semantic frames, lexical units that evoke these frames, and annotation reports that demonstrate uses of lexical units. Each semantic frame specifies a set of frame elements. These are elements that describe the situational props, participants and components that conceptually make up part of the frame. Lexical units appear in a variety of parts of speech, though we focus on verbs here. A lexical unit is a lexeme in a particular sense defined in its containing semantic frame. They are described in reports that list the syntactic realizations of the frame elements, and valence patterns that describe possible syntactic linking patterns. 3486 verb lexical units have been described in FrameNet which places a primary emphasis on providing rich, idiosyncratic descriptions of semantic properties of lexical units in context, and making explicit subtle differences in meaning. As such it could provide an important foundation for reasoning about context dependent semantic representations. However, the large number of frame elements and the current sparseness of annotations for each one has hindered machine learning.

PropBank is an annotation of 1M words of the Wall Street Journal portion of the Penn Treebank II with semantic role labels for each verb argument. Although the semantic roles labels are purposely chosen to be quite generic, i.e., ArgO, Arg1, etc., they are still intended to consistently annotate the same semantic role across syntactic variations, e.g., Arg1 in "John broke the window" is the same window (syntactic object) that is annotated as the Arg1 in "The window broke" (syntactic subject). The primary goal of PropBank is to provide consistent general labeling of semantic roles for a large quantity of text that can provide training data for supervised machine learning algorithms. PropBank can also provide frequency counts for (statistical) analysis or generation. PropBank includes a lexicon

which lists, for each broad meaning of each annotated verb, its "frameset", the possible arguments, their labels and all possible syntactic realizations. This lexical resource is used as a set of verb-specific guidelines by the annotators, and can be seen as quite similar in nature to FrameNet, although much more coarse-grained and general purpose in the specifics.

To summarize, PropBank and FrameNet both annotate the same verb arguments, but assign different labels. PropBank has a small number of vague, general purpose labels with sufficient amounts of training data geared specifically to support successful machine learning. FrameNet provides a much richer and more explicit semantics, but without sufficient amounts of training data for the hundreds of individual frame elements. An ideal environment would allow us to train generic semantic role labelers on PropBank, run them on new data, and then be able to map the resulting PropBank argument labels on rich FrameNet frame elements.

The goal of SemLink is to create just such an environment. VerbNet provides a level of representation that is still tied to syntax, in the way that PropBank is, but provides a somewhat more fine-grained set of role labels and a set of fairly high level, general purpose semantic predicates, such as $\text{contact}(x,y)$, $\text{change-of-location}(x, \text{path})$, $\text{cause}(A, X)$, etc. As such it can be seen as a mediator between PropBank and FrameNet. In fact, our approach has been to use the explicit syntactic frames of VerbNet to semi-automatically map the PropBank instances onto specific VerbNet classes and role labels. The mapping can then be hand-corrected. In parallel, SemLink has been creating a mapping table from VerbNet class(es) to FrameNet frame(s), and from role label to frame element. This will allow the SemLink project to automatically generate FrameNet representations for every VerbNet version of a PropBank instance with an entry in the VerbNet-FrameNet mapping table.

4.2.2 VerbNet \Leftrightarrow FrameNet linking

One of the tasks for the [SemLink](#) project is to provide explicit mappings between VerbNet and FrameNet. The mappings between these two resources which have complementary information about verbs and disjoint coverage open several possibilities to increase their

robustness. The fact that these two resources are now mapped gives researchers different levels of representation for events these verbs represent to be used in natural language applications. The mapping between VerbNet and FrameNet was done in two steps: (1) mapping VerbNet verb senses to FrameNet lexical units; (2) mapping VerbNet thematic roles to the equivalent (if present) FrameNet frame elements for the corresponding class/frame mappings uncovered during step 1.

In the first task, VerbNet verb senses were mapped to corresponding FrameNet senses, if available. Each verb member of a VerbNet class was assigned to a (set of) lexical units of FrameNet frames according to semantic meaning and to the roles this verb instance takes. These mappings are not one-to-one since VerbNet and FrameNet were built with distinctly different design philosophies. VerbNet verb classes are constructed by grouping verbs based mostly on their participation in diathesis alternations. In contrast, FrameNet is designed to group lexical items based on frame semantics, and a single FrameNet frame may contain sets of verbs with related senses but different subcategorization properties and sets of verbs with similar syntactic behavior may appear in multiple frames.

The second task consisted of mapping VerbNet thematic roles to FrameNet frame elements for the pairs of classes/frames found in the first task. As in the first task, the mapping is not always one-to-one as FrameNet tends to record much more fine-grained distinctions than VerbNet.

So far, 1892 VerbNet senses representing 209 classes were successfully mapped to FrameNet frames. This resulted in 582 VerbNet class – FrameNet frame mappings, across 263 unique FrameNet frames, for a total of 2170 mappings of VerbNet verbs to FrameNet lexical units.

4.2.3 PropBank \Leftrightarrow VerbNet linking

SemLink is also creating a mapping between VerbNet and PropBank, which will allow the use of the machine learning techniques that have been developed for PropBank annotations to generate more semantically abstract VerbNet representations. The mapping between VerbNet and PropBank can be divided into two parts: a "lexical mapping" and an "instance classifier."

The lexical mapping defines the set of possible mappings between the two lexicons, independent of context. In particular, for each item in the source lexicon, it specifies the possible corresponding items in the target lexicon; and for each of these mappings, specifies how the detailed fields of the source lexicon item (such as verb arguments) map to the detailed fields of the target lexicon item. The lexical mapping provides a set of possible mappings, but does not specify which of those mappings should be used for each instance; that is the job of the instance classifier, which looks at a source lexicon item in context, and chooses the most appropriate target lexicon items allowed by the lexical mapping.

The lexical mapping was created semi-automatically, based on an initial mapping which put VerbNet thematic roles in correspondence with individual PropBank framesets. This lexical mapping consists of a mapping between the PropBank framesets and VerbNet's verb classes; and a mapping between the roleset argument labels and the VerbNet thematic roles. During this initial mapping, the process of assigning a verb class to a frameset was performed manually while creating new PropBank frames. The thematic role assignment, on the other hand, was a semi-automatic process which finds the best match for the argument labels, based on their descriptors, to the set of thematic role labels of VerbNet. This process required human intervention due to the variety of descriptors for PropBank labels, the fact that the argument label numbers are not consistent across verbs, and gaps in frameset to verb class mappings.

To build the instance classifier, SemLink started with two heuristic classifiers. The first classifier works by running the SenseLearner WSD engine to find the WordNet class of each verb; and then using the existing WordNet/VerbNet mapping to choose the corresponding VerbNet class. This heuristic is limited by the performance of the WSD engine, and by the fact that the WordNet/VerbNet mapping is not available for all VerbNet verbs. The second heuristic classifier examines the syntactic context for each verb instance, and compares it to the syntactic frames of each VerbNet class. The VerbNet class with a syntactic frame that most closely matches the instance's context is assigned to the instance.

The SemLink group ran these two heuristic methods on the Treebank corpus and are hand-

correcting the results in order to obtain a VerbNet-annotated version of the Treebank corpus. Since the Treebank corpus is also annotated with PropBank information, this will provide a parallel VerbNet/PropBank corpus, which can be used to train a supervised classifier to map from PropBank frames to VerbNet classes (and vice versa). The feature space for this machine learning classifier includes information about the lexical and syntactic context of the verb and its arguments, as well as the output of the two heuristic methods.

5. Version Control

Annotation compatibility is also an issue for related formalisms. Two columns in Table 1 are devoted to different CLAWS POS tagsets, but there

are several more CLAWS tagsets (www.comp.lancs.ac.uk/ucrel/annotation.html), differing both in degree of detail and choice of distinctions made. Thus a detailed conversion table among even just the CLAWS tagsets may prove handy. Similar issues arise with the year to year changes of the ACE annotation guidelines (projects.ldc.upenn.edu/ace/) which include named entity, semantic classes for nouns, anaphora, relation and event annotation. As annotation formalisms mature, specifications can change to improve annotation consistency, speed or the usefulness for some specific task. In the interest of using old and new annotation together (more training data), it is helpful to have explicit mappings for related formalisms. Table 2 is a (preliminary) conversion table for Timex2 and Timex3, the latter of which can be viewed essentially as an elaboration of the former.

Description	TIMEX2	TIMEX3	Comment
Contains a normalized form of the date/time	VAL="1964-10-16"	val="1964-10-16"	Some TIMEX2 points are TIMEX3 durations
Captures temporal modifiers	MOD="APPROX"	mod="approx"	---
Contains a normalized form of an anchoring data/time	ANCHOR_VAL="1964-W22"	---	See TIMEX3 beginPoint and endPoint
Captures relative direction between VAL and ANCHOR_VAL	ANCHOR_DIR="BEFORE"	---	See TIMEX3 beginPoint and endPoint
Identifies set expressions	SET="YES"	type="SET"	---
Provides unique ID number	ID="12"	tid="12"	Used to relate time expressions to other objects
Identifies type of expression	---	type="DATE"	Hold over from TIMEX. Derivable from format of VAL/val
Identifies indexical expressions	---	temporalFunction="true"	In TIMEX3, indexical expressions are normalized via a temporal function, applied as post-process
Identifies reference time used to compute val	---	anchorTimeID="t12"	Desired in TIMEX2
Identifies dis-	---	functionInDocu-	Used for date stamps on

⁴ This preliminary table shows the attributes side by side with only one sample value, although other values are possible

course function		ment="CREATION_TIME"	documents
Captures anchors for durations	---	beginPoint="t11", endPoint="t12"	Captured by TIMEX2 ANCHOR attributes
Captures quantification of a set expression	---	quant="EVERY"	Desired in TIMEX2
Captures number of reoccurrences in set expressions	---	freq="2X"	Desired in TIMEX2

6. The Effect of Language Differences

Most researchers involved in linguistic annotation (particularly for NLP) take it for granted that coverage of a particular grammar for a particular language is of the utmost important. The (explanatory) adequacy of the particular linguistic theory assumed for multiple languages is considered a much less important. Given the diversity of annotation paradigms, we may go a step further and claim that it may be necessary to change theories when going from one language to another. In particular, language-specific phenomena can complicate theories in ways that prove unnecessary for languages lacking these phenomena. For example, English requires a much simpler morphological framework than languages like German, Russian, Turkish or Pashto. It has also been claimed on several occasions that a VP analysis is needed in some languages (English), but not others (Japanese). For the purposes of annotation, it would seem simplest to choose the simplest language-specific framework that is capable of capturing the distinctions that one is attempting to annotate. If the annotation is robust, it should be possible to convert it automatically into some language-neutral formalism should one arise that maximizes descriptive and explanatory adequacy. In the meanwhile, it would seem unnecessary to complicate grammars of specific languages to account for phenomena which do not occur in those languages.

6.1 The German TüBa-D/Z Treebank

German has a freer word order than English. This concerns the distribution of the finite verb and the distribution of arguments and adjuncts. German is a general Verb-Second language which means that in the default structure in declarative main clauses as well as in wh-questions the finite verb surfaces in second position preceded by only one constituent which

is not necessarily the subject. In embedded clauses the finite verb normally occurs in a verbphrase-final position following its arguments and adjuncts, and other non-finite verbal elements. German is traditionally assumed to have a head-final verb phrase. The ordering of arguments and adjuncts is relatively free. Firstly almost any constituent can be topicalised preceding the finite verb in Verb-Second position. Secondly the order of the remaining arguments and adjuncts is still relatively free. Ross (1967) coined the term Scrambling to describe the variety of linear orderings. Various factors are discussed to play a role here such as pronominal vs. phrasal constituency, information structure, definiteness and animacy (e.g. Uszkoreit 1986).

The annotation scheme of the German TüBa-D/Z treebank was developed with special regard to these properties of German clause structure. The main ordering principle is adopted from traditional descriptive analysis of German (e.g. Herling 1821, Höhle 1986). It partitions the clause into 'topological fields' which are defined by the distribution of the verbal elements. The top level of the syntactic tree is a flat structure of field categories including: Linke Klammer - left bracket (LK) and Rechte Klammer - verbal complex (VC) for verbal elements and Vorfeld - initial field (VF), C-Feld - complementiser field (C), Mittelfeld - middle field (MF), Nachfeld - final field (NF) for other elements.

Below the level of field nodes the annotation scheme provides hierarchical phrase structures except for verb phrases. There are no verb phrases annotated in TüBa-D/Z. It was one of the major design decisions to capture the distribution of verbal elements and their arguments and adjuncts in terms of topological fields instead of hierarchical verb phrase structures. The free word order would have required to make extensive use of traces or other mechanisms to relate dislocated constituents to their base

positions, which in itself was problematic since there is no consensus among German linguists on what the base ordering is. An alternative which avoids commitment to specific base positions is to use crossing branches to deal with discontinuous constituents. This approach is adopted for example by the German TIGER treebank (Brants et al. 2004). A drawback of crossing branches is that the treebank cannot be modeled by a context free grammar. Since TüBa-D/Z was intended to be used for parser training, it was not a desirable option. Arguments and adjuncts are thus related to their predicates by means of functional labels. In contrast to the Penn Treebank, TüBa-D/Z assigns grammatical functions to all arguments and adjuncts. Due to the freer word order functions cannot be derived from relative positions only.

The choice of labels of grammatical functions is largely based on the insight that grammatical functions in German are directly related to the case assignment (Reis 1982). The labels therefore do not refer to grammatical functions such as subject, direct object or indirect object but make a distinction between complement and adjunct functions and classify the nominal complements according to their case marking: accusative object (OA), dative object (OD), genitive object (OG), and also nominative 'object' (ON) versus verbal modifier (V-MOD) or underspecified modifier (MOD).

Within phrases a head daughter is marked at each projection level. Exceptions are elliptical phrases, coordinate structures, strings of foreign language, proper names and appositions within noun phrases. Modifiers of arguments and adjuncts are assigned a default non-head function. In case of discontinuous constituents the function of the modifier is either explicitly marked by means of a complex label such as OA-MOD (the modifier of an accusative object) or by means of a secondary edge REFINT in case the modified phrase has a default head or non-head function itself (which holds in the case of e.g. NP complements of prepositions).

Figures 2 to 4 illustrate the German TüBa-D/Z treebank annotation scheme (Telljohann et al. (2005). – it combines a flat topological analysis with structural and functional information.

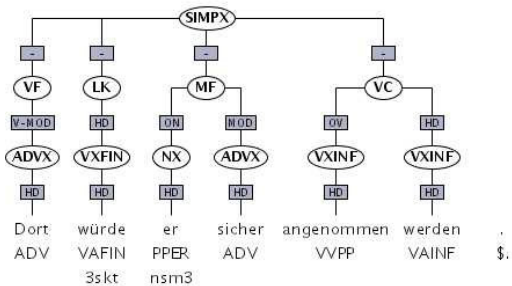


Fig. 2: verb-second
Dort würde er sicher angenommen werden.
there would he surely accepted be
'He would be surely accepted there.'

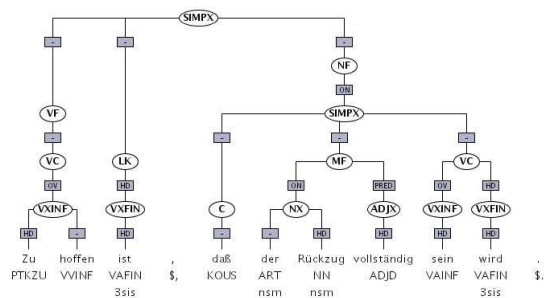


Fig. 3: verb-final
Zu hoffen ist, daß der Rückzug vollständig sein wird.
to hope is that the fallback complete be will
'We hope that they will retreat completely.'

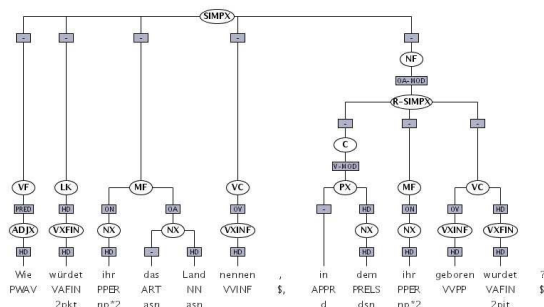


Fig. 4: discontinuous constituent marked OA-MOD
Wie würdet ihr das Land nennen, in dem ihr geboren wurdet?
how would you the country call in which you born were
'How would you call the country in which you were born?'

7. Concluding Remarks

This report has laid out several major annotation compatibility issues, focusing primarily on conversion among different annotation frameworks that represent the same type of information. We have provided procedures for conversion, along with their limitations. As more work needs to be done in this area, we intend to keep the online version available for cooperative elaboration and extension. Our hope is that the conversion tables will be extended and more annotation projects will incorporate details of their projects in order to facilitate compatibility.

The compatibility between annotation frameworks becomes a concern when (for example) a user attempts to use annotation created under two or more distinct frameworks for a single application. This is true regardless of whether the annotation is of the same type (the user wants more data for a particular phenomenon); or of different types (the user wants to combine different types of information).

Acknowledgement

This research was supported, in part, by the National Science Foundation under Grant CNI-0551615.

References

- Brants, S., S. Dipper, P. Eisenberg, S. Hansen, E. Knig, W. Lezius, C. Rohrer, G. Smith & H. Uszkoreit, 2004. TIGER: Linguistic Interpretation of a German Corpus. In E. Hinrichs and K. Simov, eds, *Research on Language and Computation, Special Issue. Volume 2*: 597-620.
- Chen, K.-J., Luo, C.-C., Gao, Z.-M., Chang, M.-C., Chen, F.-Y., and Chen, C.-J., 1999. The CKIP Chinese Treebank. In *Journ ees ATALA sur les Corpus annot es pour la syntaxe, Talana, Paris VII*: pp.85-96.
- Chen, K.-J. et al. *Building and Using Parsed Corpora*, 2003. (A. Abeillé eds) KLUWER, Dordrecht. .
- CKIP, 1995. Technical Report no. 95-02, the content and illustration of Sinica corpus of Academia Sinica. Inst. of Information Science.
- G. Corbett, N. M. Fraser, and S. McGlashan, 1993. *Heads in Grammatical Theory*. Cambridge University Press, Cambridge.
- K. Van Deemter and R. Kibble, 2001. On Coreferring: Coreference in MUC and related Annotation schemes. *Journal of Computational Linguistics* 26, 4, S. 629-637
- C. Fillmore, 1968. The Case for Case. In E. Bach and R. T. Harms, eds, *Universals in Linguistic Theory*. Holt, Rinehart and Winston, NY
- C. Fillmore, P. Kay & M. O'Connor. 1988. Regularity and Idiomaticity in Grammatical Constructions: The Case of *Let Alone.*, *Language*, 64: 501-538.
- J. S. Gruber, 1965. *Studies in Lexical Relations*. Ph.D. thesis, MIT
- E. Hajicov and M. Ceplov, 2000. Deletions and Their Reconstruction in Tectogrammatical Syntactic Tagging of Very Large Corpora. In *Proceedings of Coling 2000*: pp. 278-284.
- S. H. A. Herling, 1821. Über die Topik der deutschen Sprache. In *Abhandlungen des frankfurterischen Gelehrtenvereins für deutsche Sprache*. Frankfurt/M. Drittes Stück.
- T. N. Höhle, 1986. Der Begriff 'Mittelfeld'. Anmerkungen über die Theorie der topologischen Felder. In A. Schöne (Ed.), *Kontroversen alte und neue. Akten des 7. Internationalen Germanistenkongresses Göttingen*. 329-340.
- R. Jackendoff, 1972. *Semantic Interpretation in Generative Grammar*. MIT Press, Cambridge.
- P. Kingsbury and M. Palmer 2002. From treebank to propbank. In *Proc. LREC-2002*
- H. Lee, C.-N. Huang, J. Gao and X. Fan, 2004. Chinese chunking with another type of spec. In *SIGHAN-2004*. Barcelona: pp. 41-48.
- B. Levin 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. Univ. of Chicago Press.
- C. Manning and H. Schütze. 1999. *Foundations of Statistical Natural Language Processing*, MIT.

- A. Meyers, R. Reeves, C. Macleod, R. Szekely, V. Zielinska, B. Young, and R. Grishman, 2004. The NomBank Project: An Interim Report. In NAACL/HLT 2004 Workshop Frontiers in Corpus Annotation.
- A. Meyers, 1995. The NP Analysis of NP. In Papers from the 31st Regional Meeting of the Chicago Linguistic Society, pp. 329-342.
- D. M. Perlmutter and P. M. Postal, 1984. The 1-Advancement Exclusiveness Law. In D. M. Perlmutter & C. G. Rosen, eds 1984. *Studies in Relational Grammar 2*. Univ. of Chicago Press.
- D. M. Perlmutter, 1984. *Studies in Relational Grammar 1*. Univ. of Chicago Press.
- M. Poesio, 1999. Coreference, in MATE Deliverable 2.1, http://www.ims.uni-stuttgart.de/projekte/mate/mdag/cr/cr_1.html
- M. Poesio, 2004. "The MATE/GNOME Scheme for Anaphoric Annotation, Revisited", Proc. of SIGDIAL.
- M. Poesio and R. Artstein, 2005. The Reliability of Anaphoric Annotation, Reconsidered: Taking Ambiguity into Account. Proc. of ACL Workshop on Frontiers in Corpus Annotation.
- J. Pustejovsky, A. Meyers, M. Palmer, and M. Poesio, 2005. Merging PropBank, [NomBank](#), [TimeBank](#), Penn Discourse Treebank and Coreference. In ACL 2005 Workshop: Frontiers in Corpus Annotation II: Pie in the Sky.
- M. Reis, 1982. "Zum Subjektbegriff im Deutschen". In: Abraham, W. (Hrsg.): *Satzglieder im Deutschen. Vorschläge zur syntaktischen, semantischen und pragmatischen Fundierung*. Tübingen: Narr. 171-212.
- C. G. Rosen, 1984. The Interface between Semantic Roles and Initial Grammatical Relations. In D. M. Perlmutter and C. G. Rosen, eds, *Studies in Relational Grammar 2*. Univ. of Chicago Press.
- J. R. Ross, 1967. *Constraints on Variables in Syntax*. Doctoral dissertation, MIT.
- I. A. Sag and J. D. Fodor, 1994. Extraction without traces. In R. Aranovich, W. Byrne, S. Preuss, and M. Senturia, eds, Proc. of the Thirteenth West Coast Conference on Formal Linguistics, volume 13, CSLI Publications/SLA.
- S. Salmon-Alt and L. Romary, RAF: towards a Reference Annotation Framework, LREC 2004
- S. Shaumyan, 1977. *Applicative Grammar as a Semantic Theory of Natural Language*. Chicago Univ. Press.
- H. Telljohann, E. Hinrichs, S. Kübler and H. Zinsmeister. 2005. *Stylebook of the Tübinger Treebank of Written German (TüBa-D/Z)*. Technical report. University of Tübingen.
- C. Thielen and A. Schiller, 1996. Ein kleines und erweitertes Tagset fürs Deutsche. In: Feldweg, H.; Hinrichs, E.W. (eds.): *Wiederverwendbare Methoden und Ressourcen zur linguistischen Erschliessung des Deutschen*. Vol. 73 of *Lexicographica*. Tübingen: Niemeyer. 193-203.
- J.-L. Tsai, 2005. A Study of Applying BTM Model on the Chinese Chunk Bracketing. In LINC-2005, IJCNLP-2005, pp.21-30.
- H. Uszkoreit, 1986. "Constraints on Order" in *Linguistics* 24.
- F. Xia, M. Palmer, N. Xue, N., M. E. Okurowski, J. Kovarik, F.-D. Chiou, S. Huang, T. Kroch, and Marcus, M., 2000. Developing Guidelines and Ensuring Consistency for Chinese Text Annotation. In: Proc. of LREC-2000. Greece.
- N. Xue, F. Chiou and M. Palmer. Building a Large-Scale Annotated Chinese Corpus, 2002. In: Proc. of COLING-2002. Taipei, Taiwan.
- N. Xue, F. Xia, F.-D. Chiou and M. Palmer, 2005. The Penn Chinese TreeBank: Phrase Structure Annotation of a Large Corpus. *Natural Language Engineering*, 11(2)-207.