

# ELSNET-Project: Syntactic and Semantic Annotation of selected Verbs

Hannah Kermes, Ulrich Heid

Spring 2000

## 1 Selection of German lemmas

### 1.1 Criteria

The following criteria were used to select the German verbs to ensure that they prove useful for the project and are comparable with the Italian counterparts:

- The verbs are translation equivalents of non-peripheral readings of the Italian verbs.
- The verbs display a diversity of subcategorization frames (SC-frames).
- The verbs display a diversity of collocations and of sortal types of complement heads.
- The verbs are frequent enough to ensure a good representation within the corpora.

As a first step in selecting the German verbs for the ELSNET-project, a list of translations for the Italian verbs was made. These verbs were then checked superficially for diversity of their subcategorization frames, collocations and sortal complement types as well as for their frequency. A second list of verbs was made containing those verbs that seemed interesting and frequent enough. The goal was to keep the verbs semantically and syntactically as divers as possible, i.e., to keep a one to one relation between Italian and German whenever possible.

In a second step the candidate verbs were checked in detail to see which of them matched the criteria best and promised to give the most interesting and telling results in a syntactic and semantic annotation.

Problems arose with the attempt to find German equivalents for the Italian reflexive verbs (*arrestarsi*, *chiamarsi*, *mantenersi*) the German counterparts of which being of low to very low frequency (if at all in the corpora). Besides, for some Italian verbs the most typical German verbal equivalent seemed not to be very useful, mostly because of the low frequency. It was decided to leave the equivalents of Italian reflexive verbs aside. Instead, the reflexive realization of some of the other words was considered. However, the reflexive subcategorization frames were not listed under a separate entry but were added to the subcategorization frames of the non-reflexive verb.

## 1.2 Selected verbs

List of selected verbs	
Italian	German
abbandonare	aufgeben SC:4 <sup>1</sup> ; verlassen SC:2
arrestare	festnehmen SC:1; aufhalten SC:1+3refl
aprire	öffnen SC:2+3refl; eröffnen SC:3+1refl
prevedere	vorsehen/planen SC:2+1(2)refl/SC:2; vermuten/ahnen SC:1/SC:2
tagliare	reduzieren SC:2+1refl; unterbrechen SC:3(4)
comprendere	verstehen SC:3(4)+2(3)refl
chiamare	rufen SC:4(6); aufrufen SC:(3)
chiedere	bitten SC:3; fragen (nach) SC:2+2(nach)+1refl
perseguire	fortsetzen SC:2+2refl; verfolgen SC:2
coprire	umfassen SC:2+2refl; decken; schützen SC:4(6)+2(4)refl
entrare	passen (in) SC:3+1(in)+1(zu); eintreten SC:3(4); sich beteiligen SC:2+2refl
vedere	sehen SC:4(5)+1refl; erkennen SC:3(5)+1refl
mantenere	halten SC:7(8)
mantenersi	andauern SC:2; sich halten (an) SC:2(3)
percepire	erhalten SC:1
esercitare	tätig sein/betreiben SC:2; üben/sich üben SC2+2refl
presentare	zeigen SC:5+2refl; präsentieren SC:3(4)+2refl
portare	bringen SC:4(5); tragen SC:4+1refl
provare	üben/proben SC:3; versuchen SC:1+2(4)refl; spüren/empfinden SC:3/SC:2(3)
realizzare	realisieren SC:1; begreifen SC:3
arrivare	erreichen SC:1; ankommen SC:6(8)

## 2 Selection of German corpus material

### 2.1 Corpora

Corpus	period	volumes	size
<i>Frankfurter Rundschau</i>	1992/93	2	40 M
<i>Stuttgarter Zeitung</i>	1992/93	2	36 M
<i>die tageszeitung</i>	1987-93	7	103 M
'European News Cp.': dpa, afp...	1990-94		100 M

Figure 1: Journalistic Corpora used

### 2.2 Subcategorization frames and lexical combinations - criteria for selection

The main goal of the ELSNET project being an annotation at the semantic level of description displaying the various types of subcategorization frames the question arose whether to include sentential complements in the analysis. On the one hand, these complements are an important part of the subcategorization properties of the verb and, therefore, could be telling with respect to the syntactic as well as semantic selection of the verb. On the other hand, they are difficult to annotate and to group under semantic sorts. For Italian the same problem had come up

and it was decided to include only very few examples of sentential complements in order not to avoid them completely and to be able to display some typical meanings of verbs that can only be expressed by sentential complements. For German it was decided to exclude sentential complements altogether with respect to the annotation. However, they were kept in mind and were listed in the appendix wherever they can possibly alternate with nominal or prepositional complements in order not to lose completely the information they can give to this respect.

List of subcategorization frames:

- (subj(NP\_nom))
- (subj(NP\_nom),arg(NP\_acc))
- (subj(NP\_nom),arg(PRON\_refl-acc))
- (subj(NP\_nom),arg(PRON\_refl-acc),iobj(NP\_dat))
- (subj(NP\_nom),arg(PRON\_refl-acc),mod-loc(AdvP))
- (subj(NP\_nom),arg(PRON\_refl-acc),p-obj  
(PP\_an\_acc), (PP\_an\_dat), (PP\_auf\_acc), (PP\_auf\_dat), (PP\_für\_acc), (PP\_gegen\_acc),  
(PP\_in\_acc), (PP\_in\_dat), (PP\_mit\_dat), (PP\_mit\_dat), (PP\_vor\_dat), (PP\_zu\_dat))
- (subj(NP\_nom),iobj(NP\_dat))
- (subj(NP\_nom),mod-loc(AdvP))
- (subj(NP\_nom),obj(NP\_acc))
- (subj(NP\_nom),obj(NP\_acc),iobj(NP\_dat))
- (subj(NP\_nom),obj(NP\_acc),mod-dir(AdvP))
- (subj(NP\_nom),obj(NP\_acc),obj-pred  
(als\_Adj), (für\_Adj), (PP\_als))
- (subj(NP\_nom),obj(NP\_acc),p-obj  
etc.
- (subj(NP\_nom),p-obj  
etc.
- (subj(NP\_nom),subj-pred(PP\_als))
- (subj(PRON\_dummy),p-obj(PP\_auf\_acc))

### 2.3 Extracting the data from corpora

A first step in extracting data from corpora was to create a subcorpus of all sentences containing the lemmata of the selected verbs. (Thereby, sentence refers not to the whole sentence but only to the clausal part containing the respective verb.) The applicable SC-frames were identified along with the respective templates. By means of these templates sentences were extracted out of the lemma corpus and stored as different subcorpora, one for each template.

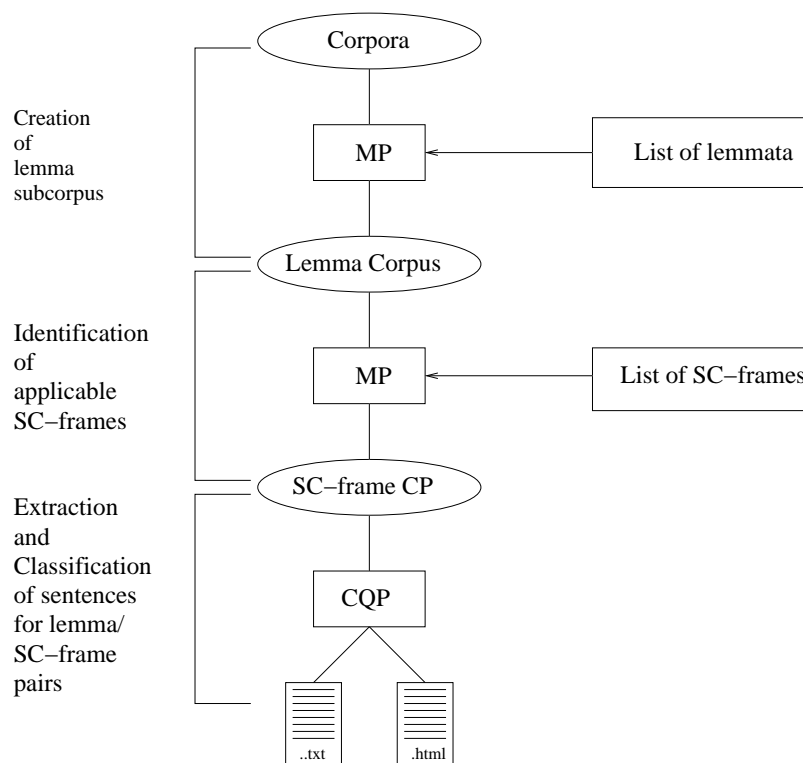


Figure 2: Extraction of data from corpora

In a separate step groupings for each template were made displaying the frequency of this template for each lemmata. These groupings were then aligned with the respective SC-frames and manually filtered.

Each lemma SC-frame pair was then separately applied to the respective subcorpus using CQP commands to extract the relevant sentences. These sentences were then stored in txt and html lemma files, which display the sentences sorted according to their SC-frames.

Concentrating on the diversity of verb-noun-collocations an additional method was used to extract data from corpora. Assuming that the relevant complements are to the right of a sentence preceding the main verb, verbs were extracted along with the next noun to the left, where appropriate also with the respective preposition. These verb-noun-pairs were then listed and manually checked for interesting and applicable collocations. The respective sentences were extracted out of the corpus.

## 2.4 Selection of sentences

The extracted sentences were checked manually. Sentences displaying redundancies concerning SC-frames and collocations as well as sentences that could pose a problem for the parser were sorted out. Overall the number of sentences for each verb was reduced to about the required 50.

### 3 Parsing of the sentences

#### 3.1 LFG (Lexical Functional Grammar) Parsing

The selected sentences were tokenized and parsed using the XLE(Xerox Linguistic Environment) parser. Yet, format and features of this parser differ from the MATE/SPARKLE-format.

As LFG has more functional features they had to be mapped to the required features in the following way:

LFG	target
SUBJ	-> SUBJ
OBJ	-> OBJ
OBJ2	-> IOBJ (= NPgen,NPdat, second NPacc)
OBL	-> ARGMOD (preposition as INTRO)
XCOMP	-> XCOMP (without predicative XCOMPs)
ADJUNCT, APP, POSS,	-> MOD
XCOMP	-> XCOMP-PRED (predicative XCOMPs)
MOD	-> COMPOUND (in compounds)

The format we received as a result of the parser after the mapping of the functional features looked as follows:

```
fstructure('Translation of: h_127799749: wenn sie sich nicht selbst aufgeben will',
  % Properties:
  [
    transfer_grammar_name('transfer rules for TIGER'),
    time('Unspecified'),
    rootcategory('VLAST'),
    grammar('GERMAN GRAM')
  ],
  % Choices:
  [
  ],
  % Equivalences:
  [

  ],
  % Constraints:
  [
    cf(1, eq(attr(var(0), 'SENTENCE-ID'), h_127799749)),
    cf(1, eq(attr(var(0), 'SUBJ'), var(3))),
    cf(1, eq(attr(var(0), 'XCOMP'), var(2))),
    cf(1, in_set(var(7), var(13))),
    cf(1, in_set(var(6), var(13))),
    cf(1, eq(attr(var(2), 'DOBJ'), var(5))),
    cf(1, eq(attr(var(0), 'MOD'), var(13))),
    cf(1, eq(attr(var(0), 'PRED-ALL'), var(19))),
```

```

cf(1, in_set('auf#geben',var(19))),
cf(1, in_set(selbst,var(19))),
cf(1, in_set(nicht,var(19))),
cf(1, in_set(sie,var(19))),
cf(1, in_set(sich,var(19))),
cf(1, in_set(wollen,var(19))),
cf(1, eq(attr(var(0),'LEMMA'),wollen)),
cf(1, eq(attr(var(5),'LEMMA'),sich)),
cf(1, eq(attr(var(3),'LEMMA'),sie)),
cf(1, eq(attr(var(7),'LEMMA'),nicht)),
cf(1, eq(attr(var(6),'LEMMA'),selbst)),
cf(1, eq(attr(var(2),'LEMMA'),'auf#geben'))
],
[]).

```

fstructure('Translation of: h\_107504781: wenn Sie sich nicht auf die Originalsprache verlassen müssen',

```

% Properties:
[
transfer_grammar_name('transfer rules for TIGER'),
time('Unspecified'),
rootcategory('VLAST'),
grammar('GERMAN GRAM')
],
% Choices:
[
],
% Equivalences:
[

],
% Constraints:
[
cf(1, eq(attr(var(0),'SENTENCE-ID'),h_107504781)),
cf(1, eq(attr(var(5),'INTRO'),auf)),
cf(1, eq(attr(var(0),'SUBJ'),var(3))),
cf(1, eq(attr(var(0),'XCOMP'),var(2))),
cf(1, in_set(var(11),var(17))),
cf(1, eq(attr(var(2),'DOBJ'),var(6))),
cf(1, eq(attr(var(2),'ARGMOD'),var(5))),
cf(1, eq(attr(var(0),'MOD'),var(17))),
cf(1, eq(attr(var(0),'PRED-ALL'),var(23))),
cf(1, in_set(verlassen,var(23))),
cf(1, in_set('Sprache',var(23))),
cf(1, in_set(nicht,var(23))),
cf(1, in_set('Sie',var(23))),
cf(1, in_set(sich,var(23))),

```

```

cf(1, in_set(müssen,var(23))),
cf(1, eq(attr(var(0),'LEMMA'),müssen)),
cf(1, eq(attr(var(6),'LEMMA'),sich)),
cf(1, eq(attr(var(3),'LEMMA'),'Sie')),
cf(1, eq(attr(var(11),'LEMMA'),nicht)),
cf(1, eq(attr(var(5),'LEMMA'),'Sprache')),
cf(1, eq(attr(var(2),'LEMMA'),verlassen))
],
[]).

```

This format was then mapped in a two-step-process to the MATE/SPARKLE format by means of prolog and perl. The first step produced a flattened structure of the analysis, which could be checked easily. Besides, now the whole sentence was displayed and the nouns were extracted out of the clausal part containing the respective verb making them available for the semantic annotation.

h\_127799749:

```

01. 02.          03.  04.  05.  06.  07.          08.          09. 10. 11.
An  irgendeinem Punkt hatte sich das Bewußtsein durchgesetzt , daß die
12. 13. 14.      15.   16. 17. 18. 19. 20. 21. 22. 23.      24.
taz eine Grenze ziehen muß , wenn sie sich nicht selbst aufgeben will
25.
.

23  DOBJ    20
24  MOD     21
24  MOD     22
24  SUBJ    19
24  XCOMP   23

```

h\_107504781:

```

01. 02. 03. 04.          05.  06. 07.  08. 09. 10. 11. 12. 13.
Der Weg zur walisischen Küche ist leicht , wenn Sie sich nicht auf
14. 15.          16.      17.  18.
die Originalsprache verlassen müssen .

17  MOD     12
17  SUBJ    10
17  XCOMP   16
16  DOBJ    11
16  ARGMOD  15 (INTRO auf)

```

NOUN Sprache

In a second step the format was mapped to the final MATE/SPARKLE format. It additionally includes the part-of-speech tags of all the words in the sentence and a reference to the corpora.

src=taz

An irgendeinem Punkt hatte sich das Bewußtsein durchgesetzt ,  
daß die taz eine Grenze ziehen muß , wenn sie sich nicht selbst aufgeben will .

```
<w id=w01 type="APPR" lemma="an"> An </w>
<w id=w02 type="PIDAT" lemma="irgendeinem"> irgendeinem </w>
<w id=w03 type="NN" lemma="Punkt"> Punkt </w>
<w id=w04 type="VAFIN" lemma="haben"> hatte </w>
<w id=w05 type="PRF" lemma="sich"> sich </w>
<w id=w06 type="ART" lemma="d"> das </w>
<w id=w07 type="NN" lemma="Bewußtsein"> Bewußtsein </w>
<w id=w08 type="VPPP" lemma="durchsetzen"> durchgesetzt </w>
<w id=w09 type="$, " lemma=","> , </w>
<w id=w10 type="KOUS" lemma="daß"> daß </w>
<w id=w11 type="ART" lemma="d"> die </w>
<w id=w12 type="NN" lemma="taz"> taz </w>
<w id=w13 type="ART" lemma="ein"> eine </w>
<w id=w14 type="NN" lemma="Grenze"> Grenze </w>
<w id=w15 type="VVINF" lemma="ziehen"> ziehen </w>
<w id=w16 type="VMFIN" lemma="müssen"> muß </w>
<w id=w17 type="$, " lemma=","> , </w>
<w id=w18 type="KOUS" lemma="wenn"> wenn </w>
<w id=w19 type="PPER" lemma="PPER"> sie </w>
<w id=w20 type="PRF" lemma="sich"> sich </w>
<w id=w21 type="PTKNEG" lemma="nicht"> nicht </w>
<w id=w22 type="ADV" lemma="selbst"> selbst </w>
<w id=w23 type="VVINF" lemma="aufgeben"> aufgeben </w>
<w id=w24 type="VMFIN" lemma="wollen"> will </w>
<w id=w25 type="$. " lemma="."> . </w>
```

```
<link id=01 s-type="DOBJ" targets = "w23 w20">
<link id=02 s-type="MOD" targets = "w24 w21">
<link id=03 s-type="MOD" targets = "w24 w22">
<link id=04 s-type="SUBJ" targets = "w24 w19">
<link id=05 s-type="XCOMP" targets = "w24 w23">
```

src=taz

Der Weg zur walisischen Küche ist leicht ,  
wenn Sie sich nicht auf die Originalsprache verlassen müssen .

```
<w id=w01 type="ART" lemma="d"> Der </w>
<w id=w02 type="NN" lemma="Weg"> Weg </w>
<w id=w03 type="APPRART" lemma="zur"> zur </w>
<w id=w04 type="ADJA" lemma="walisisch"> walisischen </w>
<w id=w05 type="NN" lemma="Küche"> Küche </w>
```



```

<w id=w06 type="VAFIN" lemma="sein"> ist </w>
<w id=w07 type="ADJD" lemma="leicht"> leicht </w>
<w id=w08 type="$, " lemma=","> , </w>
<w id=w09 type="KOUS" lemma="wenn"> wenn </w>
<w id=w10 type="PPER" lemma="PPER"> Sie </w>
<w id=w11 type="PRF" lemma="sich"> sich </w>
<w id=w12 type="PTKNEG" lemma="nicht"> nicht </w>
<w id=w13 type="APPR" lemma="auf"> auf </w>
<w id=w14 type="ART" lemma="d"> die </w>
<w id=w15 type="NN" lemma="Originalsprache" semtype="language"> Originalsprache </w>
<w id=w16 type="VVINF" lemma="verlassen"> verlassen </w>
<w id=w17 type="VMINF" lemma="müssen"> müssen </w>
<w id=w18 type="$. " lemma="."> . </w>

<link id=01 s-type="MOD" targets = "w17 w12">
<link id=02 s-type="SUBJ" targets = "w17 w10">
<link id=03 s-type="XCOMP" targets = "w17 w16">
<link id=04 s-type="DOBJ" targets = "w16 w11">
<link id=05 s-type="ARGMOD" intro = "auf" targets = "w16 w15">

```

## 4 Semantic annotation

The semantic annotation was performed on head nouns of arguments and modifiers, as well as on modifier nouns of arguments. They were assigned a semantic type according to the SIMPLE guidelines. The goal was to be as precise as possible using the most restrictive type possible. In addition, the nouns were assembled in domains, where such a classification seemed interesting and telling with respect to the semantic and syntactic features of the specific verb they occurred with.

Although the SIMPLE guidelines seem very detailed, there nevertheless arose some problems for the annotation. Some of the words did not seem to fit in any of the classes, e.g. *Frisur* refers to the way hair is cut and styled, it is something typical of mankind, yet it is not a 'body part', nor 'clothing', and it seems also awkward to classify simply as 'artifact'. Consequently, it was classified with two semtypes, 'artifact' (produced by man) and 'body part'. Another word which was hard to classify is *Tote* (someone dead), it is a human being, however, it is not a 'living entity' (the superclass of 'human'). The counterpart 'dead entity' does not exist. Thus, it was classified as a 'organic object', which seems rather far fetched. The word *Bühne* (*stage*) as well seemed problematic. It is neither really a 'part of a building' nor 'furniture', 'artifact' and 'location' on the other hand also seemed odd. Thus, again, it was classified with two semtypes: 'part of building' and 'furniture'.

The results were stored in a database (s. screen shot on next page).

GTKSQL: ssa Database Query Table Help

relations

Field Type Length  
 cat int4 4  
 verb text var  
 rel text var  
 intro text var  
 format text var

relations

Field Type Length  
 cat int4 4  
 verb text var  
 rel text var  
 intro text var  
 format text var

select cat, verb, subcat, rel, intro, target, semtype, sent  
 from relations  
 where verb = 'beteiligen'

order by subcat, rel, intro, semtype, target

cat	verb	subcat	rel	intro	target	semtype	sent
11	beteiligen	<S DO PO>	DOBU	sich	sich		Dem jungen Mann , der die 11. Klasse des Gymnasiums besucht , akzentfrier Deutsch spricht und vor einem Dreivierteljahr Unklar war bis zuletzt , ob Inkahta sich überhaupt an der Regierung beteiligen würde .
11	beteiligen	<S DO PO>	DOBU	sich	sich		Es sei auch klar , daß die Stadt sich an der Finanzierung des 30-Millionen-Projekts nicht beteiligen werde .
11	beteiligen	<S DO PO>	POBU	Verbrechen	Verbrechen	act	Es seien nur solche Personen eingestellt worden , bei denen eine Überprüfung ergeben habe , daß sie sich nicht an Mens Kinkel will rechtlich prüfen lassen , ob Deutschland sich an Blaunheim-Missionen beteiligen kann .
11	beteiligen	<S DO PO>	POBU	Menschenrechtsverletzung	change_of_state	act	Die Gefangenen geben an , sie seien verschleppt worden , obwohl sie sich an keinerlei Kampfhandlungen beteiligt hätten .
11	beteiligen	<S DO PO>	POBU	Aktion	act		Die regierenden Konservativen warten 30 Labour - Abgeordneten Gesetzesbruch vor , weil sie sich an dem Kopfsteuer-Bi Generalsstaatsanwalt Kim Ki Choon hat bereits angeordnet , daß er gegen weitere KBS- Angestellte Untersuchungsverfahren Nicht betroffen von dem Verbot , zu kandidieren , sind außerdem einzelne Mitglieder der sechs Organisationen , sofern sie Es komme überhaupt nicht in Frage , beschied Gardini dem Britten Sir Jimmy Goldsmith , daß er sich an einem feindlichen Schaick stellte kurz und bündig klar , daß er sich nicht für alles Schlechte in der DDR verantwortlich machen lasse , zumal Einer der von der SPD benannten Fachleute , der Richter am Hessischen Staatsgerichtshof , Helmut Lenz , hat allerdings ! Es wäre also keineswegs überraschend , wenn Daimler und Mitsubishi sich an der Verteilung des Kuchens beteiligen woll Im Januar war er in Miami Beach vorübergehend festgenommen worden , nachdem er sich an einer Wirtschaftsprüfung betete Er habe ein volles Geständnis abgelegt , wonach ihm ein Fundamentalistführer Geld versprochen habe , wenn er sich ø Sikando habe gleichzeitig die betürchtigung geäußert , daß seine Kollegen ihm tätlich angreifen könnten , weil er sich nicht Wirtschaftsminister Walter Hirche ( FDP ) hatte vor der Betriebsversammlung gesagt , die Lage habe sich entspannt , nach Haben die Stadträte und -- ungewöhnlich scharf im Ton -- Finanzbürgermeister Klaus Lang gestern schon Vögele klare Gestern teilte die Blohm und Voss AG mit , daß sie sich nicht an einem nordeutschen Marine-Technikverbund beteiligen v Der deutsche Einfluß im Bündnis sei davon abhängig , ob man sich an solchen Einheiten beteilige .
11	beteiligen	<S DO PO>	POBU	Verbrechen	Verbrechen	act	Bei den Verhandlungen in London wurde versprochen , daß die EPLF sich an einer Übergangsregierung beteiligen würde Die Bereitschaft von VDO , die Boge-Anteile zu verkaufen , wird in der Branche auch als Bestätigung gewertet , daß der in der WEU finden diese Beratungen unter deutschem Vorsitz statt , aber jeder weiß , daß die Präsidentschaft sich an solc Inkahta-Führer Butelezi hatte letzte Woche gesagt , die Einigung auf zwei Stimmen könne " wirklich dazu beitragen " , d Die Ergebnisse der Umfrage müssen allerdings schon deshalb unter Vorbehalt betrachtet werden , weil auffällig viele Wähler Eine lange Liste von Firmen wurde um Sponsoring angesprochen , viele Vereine und Verbände wurden gefragt , ob sie sic Die Hilfe für die ostdeutschen Hochschulen " hätte noch um 440 Millionen Mark höher ausfallen können " , wenn die alter Japans Regierung übernahm dieses Argument der Deutschen in Maden und versprach , daß sie sich durchaus an multilater Bonn ( ap ) - Die Lohnnebenkosten sollen wegen der geplanten Pflegeversicherung nicht steigen , obwohl die Arbeitsgebe Es sei auch klar , daß die Stadt sich an der Finanzierung des 30-Millionen-Projekts nicht beteiligen werde .
11	beteiligen	<S DO PO>	POBU	Verbrechen	Verbrechen	act	Am heutigen Donnerstag müssen die Stadtverordneten entscheiden , ob sie sich an den fünfstelligen Kosten eines im Herl Er deutete ferne an , daß Dänemark sich nicht an einer gemeinsamen Währung beteiligen wolle .
11	beteiligen	<S DO PO>	POBU	Verbrechen	Verbrechen	act	- Der Präsident der Bundesvereinigung der Deutschen Arbeitgeberverbände ( BDA ) , Klaus Murrmann , hat es abgelehnt Dem jungen Mann , der die 11. Klasse des Gymnasiums besucht , akzentfrier Deutsch spricht und vor einem Dreivierteljahr

Query results : 8 fields and 164 records

## 5 Appendix

### 5.1 Number of sentences for each selected verb

verb	unsorted	sorted	selected	VE-koll	V2-koll
01 aufgeben	144	107	88	0	0
01 verlassen	262	155	121	0	0
02 aufhalten	53	36	34	21	14
03 öffnen	84	70	55	30	11
04 planen	29	28	26	35	14
04 vorsehen	16	16	16	83	7
05 reduzieren	27	22	22	46	13
05 unterbrechen	25	22	20	34	20
06 verstehen	180	95	89	0	0
07 rufen	43	22	18	33	23
08 fragen	77	29	19	54	18
09 fortsetzen	108	83	73	31	27
10 decken	33	27	25	22	23
11 beteiligen	92	81	78	0	0
12 erkennen	97	68	65	44	19
13 halten	331	177	165	0	0
14 erhalten	151	130	127	0	0
15 betreiben	44	38	38	70	30
15 üben	18	11	11	18	30
16 zeigen	88	79	79	0	0
17 tragen	138	109	86	0	0
18 üben	18	11	11	18	30
18 versuchen	11	10	10	13	21
19 realisieren	41	38	33	36	15
20 erreichen	262	195	156	0	0

### 5.2 Verbs and their subcategorization frames

#### ABBANDONARE

-- aufgeben SC:4

auf#geben haben SADAW (subj(NP\_nom)) CELEX

auf#geben haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW

auf#geben haben RULE1 (subj(NP\_nom),obj(NP\_acc),iobj(NP\_dat)) CELEX

auf#geben haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(PP\_als)) HGC

-- verlassen SC:2

verlassen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW

verlassen haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(PP\_als)) HGC

#### ARRESTARE

-- festnehmen SC:1

fest#nehmen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW

-- aufhalten SC:1+3refl  
auf#halten haben RULE1 (subj(NP\_nom),obj(NP\_acc)) HGC  
auf#halten haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) SADA  
au#halten haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_mit\_dat)) HGC  
auf#halten haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),mod-loc(AdvP)) HGC

#### APRIRE

-- öffnen SC:2+3refl  
öffnen haben SADA (subj(NP\_nom)) HGC  
öffnen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA  
öffnen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) HGC  
öffnen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) SADA  
öffnen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),iobj(NP\_dat)) HGC  
öffnen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_für\_acc)) HGC

-- eröffnen SC:3+1refl  
eröffnen haben SADA (subj(NP\_nom)) HGC  
eröffnen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA  
eröffnen haben RULE1 (subj(NP\_nom),obj(NP\_acc),iobj(NP\_dat)) HGC  
eröffnen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) HGC

#### PREVEDERE

-- vorsehen/planen SC:2+1(2)refl/SC:2  
vor#sehen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) HGC  
vor#sehen haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(PP\_als)) HGC  
vor#sehen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_für\_acc)) SADA  
vor#sehen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_vor\_dat)) SADA

planen haben SADA (subj(NP\_nom)) HGC  
planen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA

-- vermuten/ahnen SC:1/SC:2  
vermuten haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA

ahnen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA  
ahnen haben SADA (subj(NP\_nom),iobj(NP\_dat)) CELEX

#### TAGLIARE

-- reduzieren SC:2+1refl  
reduzieren haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA  
reduzieren haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_auf\_acc)) HGC  
reduzieren haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) HGC

```

-- unterbrechen SC:3(4)
unterbrechen haben SADA (subj(NP_nom)) HGC
unterbrechen haben RULE1 (subj(NP_nom),obj(NP_acc)) SADA
unterbrechen haben RULE1 (subj(NP_nom),obj(NP_acc),p-obj(PP_bei_dat))
SADA
unterbrechen haben RULE1 (subj(NP_nom),obj(NP_acc),p-obj(PP_mit_dat)) SADA

```

#### COMPRENDERE

```

-- verstehen SC:3(4)+2(3)refl
verstehen haben SADA (subj(NP_nom)) HGC
verstehen haben RULE1 (subj(NP_nom),obj(NP_acc)) SADA
verstehen haben RULE1 (subj(NP_nom),obj(NP_acc),obj-pred(als_Adj)) HGC
verstehen haben RULE1 (subj(NP_nom),obj(NP_acc),obj-pred(PP_als)) HGC
verstehen haben RULE1 (subj(NP_nom),arg(PRON_refl-acc)) SADA
verstehen haben RULE1 (subj(NP_nom),arg(PRON_refl-acc),p-obj(PP_auf_acc)) SADA
verstehen haben RULE1 (subj(NP_nom),arg(PRON_refl-acc),p-obj(PP_mit_dat)) SADA

```

#### CHIAMARE

```

-- rufen SC:4(6)
rufen haben SADA (subj(NP_nom)) HGC
rufen haben RULE1 (subj(NP_nom),obj(NP_acc)) SADA
rufen haben RULE1 (subj(NP_nom),obj(NP_acc),iobj(NP_dat)) SADA
rufen haben SADA (subj(NP_nom),p-obj(PP_nach_dat)) HGC
rufen haben SADA (subj(NP_nom),p-obj(PP_um_acc)) HGC
rufen haben SADA (subj(NP_nom),p-obj(PP_zu_dat)) HGC

```

#### -- aufrufen SC:(3)

```

auf#rufen haben RULE1 (subj(NP_nom),obj(NP_acc)) SADA
auf#rufen haben SADA (subj(NP_nom),p-obj(PP_zu_dat)) HGC
auf#rufen haben RULE1 (subj(NP_nom),obj(NP_acc),p-obj(PP_zu_dat)) HGC

```

#### CHIEDERE

```

-- bitten SC:3
bitten haben RULE1 (subj(NP_nom),obj(NP_acc)) SADA
bitten haben SADA (subj(NP_nom),p-obj(PP_um_acc)) HGC
bitten haben RULE1 (subj(NP_nom),obj(NP_acc),p-obj(PP_um_acc)) SADA

```

#### -- fragen (nach) SC:2+2(nach)+1refl

```

fragen haben SADA (subj(NP_nom)) HGC
fragen haben RULE1 (subj(NP_nom),obj(NP_acc)) SADA
fragen haben SADA (subj(NP_nom),p-obj(PP_nach_dat)) HGC
fragen haben RULE1 (subj(NP_nom),obj(NP_acc),p-obj(PP_nach_dat)) SADA
fragen haben RULE1 (subj(NP_nom),arg(PRON_refl-acc)) SADA

```

#### PERSEGUIRE

```

-- fortsetzen SC:2+2refl

```

fort#setzen haben SADA (subj(NP\_nom)) HGC  
fort#setzen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) HGC  
fort#setzen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA  
fort#setzen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) HGC  
fort#setzen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) SADA  
fort#setzen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_in\_acc)) SADA

-- verfolgen SC:2  
verfolgen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA  
verfolgen haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(PP\_als)) HGC

#### COPRIRE

-- umfassen SC:2+2refl  
umfassen haben SADA (subj(NP\_nom),arg(NP\_acc)) HGC

-- decken  
decken haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA  
decken haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) HGC  
decken haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_mit\_dat)) HGC

-- schützen SC:4(6)+2(4)refl  
schützen haben SADA (subj(NP\_nom)) HGC  
schützen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA  
schützen haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_gegen\_acc)) SADA  
schützen haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_mit\_dat)) SADA  
schützen haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_vor\_dat)) SADA  
schützen haben SADA (subj(NP\_nom),p-obj(PP\_vor\_dat)) HGC  
schützen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) SADA  
schützen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_gegen\_acc)) SADA  
schützen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_mit\_dat)) SADA  
schützen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_vor\_dat)) SADA

#### ENTRARE

-- passen (in) SC:3+1(in)+1(zu)  
passen haben SADA (subj(NP\_nom)) HGC  
passen haben SADA (subj(NP\_nom),iobj(NP\_dat)) HGC  
passen haben SADA (subj(NP\_nom),subj-pred(PP\_als)) HGC  
passen haben SADA (subj(NP\_nom),p-obj(PP\_in\_acc)) HGC  
passen haben SADA (subj(NP\_nom),p-obj(PP\_zu\_dat)) HGC

-- eintreten SC:3(4)  
ein#treten haben-variant SADA (subj(NP\_nom)) SADA  
ein#treten sein-variant SADA (subj(NP\_nom)) SADA  
ein#treten haben RULE1 (subj(NP\_nom),obj(NP\_acc)) HGC  
ein#treten haben-variant SADA (subj(NP\_nom),p-obj(PP\_für\_acc)) SADA  
ein#treten haben-variant SADA (subj(NP\_nom),p-obj(PP\_in\_acc)) SADA

ein#treten sein-variant SADAW (subj(NP\_nom),p-obj(PP\_für\_acc)) SADAW  
ein#treten sein-variant SADAW (subj(NP\_nom),p-obj(PP\_in\_acc)) SADAW

-- sich beteiligen SC:2+2refl  
beteiligen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW  
beteiligen haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_an\_acc)) HGC  
beteiligen haben RULE1 (subj(NP\_nom),arg(PRON\_refl\_acc)) HGC  
beteiligen haben RULE1 (subj(NP\_nom),arg(PRON\_refl\_acc),p-obj(PP\_an\_dat)) HGC

#### VEDERE

-- sehen SC:4(5)+1refl  
sehen haben SADAW (subj(NP\_nom)) HGC  
sehen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW  
sehen haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(PP\_als)) HGC  
sehen haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_an\_acc)) SADAW  
sehen haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_nach\_dat)) SADAW  
sehen haben RULE1 (subj(NP\_nom),arg(PRON\_refl\_acc)) SADAW

-- erkennen SC:3(5)+1refl  
erkennen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW  
erkennen haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(als\_Adj)) HGC  
erkennen haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(für\_Adj)) HGC  
erkennen haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(PP\_als)) HGC  
erkennen haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_an\_dat)) HGC  
erkennen haben RULE1 (subj(NP\_nom),arg(PRON\_refl\_acc)) SADAW

#### MANTENERE

-- halten SC:7(8)  
halten haben SADAW (subj(NP\_nom)) HGC  
halten haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW  
halten haben SADAW (subj(NP\_nom),p-obj(PP\_gegen\_acc)) HGC  
halten haben SADAW (subj(NP\_nom),mod-loc(AdvP)) HGC  
halten haben RULE1 (subj(NP\_nom),obj(NP\_acc),iobj(NP\_dat)) HGC  
halten haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(für\_Adj)) HGC  
halten haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_für\_acc)) SADAW  
halten haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_zu\_dat)) SADAW

#### MANTENERSI

-- andauern SC:2  
an#dauern haben SADAW (subj(NP\_nom)) HGC  
an#dauern haben SADAW (subj(NP\_nom),arg(NP\_acc)) HGC

-- sich halten (an) SC:2(3)  
halten haben RULE1 (subj(NP\_nom),arg(PRON\_refl\_acc)) SADAW  
halten haben RULE1 (subj(NP\_nom),arg(PRON\_refl\_acc),p-obj(PP\_an\_acc)) SADAW  
halten haben RULE1 (subj(NP\_nom),arg(PRON\_refl\_acc),p-obj(PP\_zu\_dat)) SADAW

PERCEPIRE

-- erhalten SC:1  
erhalten haben SADA (subj(NP\_nom),arg(NP\_acc)) HGC

ESERCITARE

-- tätig sein/betreiben SC:2  
betreiben haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA  
betreiben haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(PP\_als)) HGC

-- üben /sich üben SC2+2refl  
üben haben SADA (subj(NP\_nom)) HGC  
üben haben RULE1 (subj(NP\_nom),obj(NP\_acc)) HGC  
üben haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) SADA  
üben haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_in\_dat)) SADA

PRESENTARE

-- zeigen SC:5+2refl  
zeigen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA  
zeigen haben SADA (subj(NP\_nom),p-obj(PP\_auf\_acc)) HGC  
zeigen haben RULE1 (subj(NP\_nom),obj(NP\_acc),iobj(NP\_dat)) SADA  
zeigen haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(PP\_als)) HGC  
zeigen haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_auf\_acc)) SADA  
zeigen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) HGC  
zeigen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_an\_dat)) HGC  
zeigen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_in\_dat)) HGC

-- präsentieren SC:3(4)+2refl  
präsentieren haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA  
präsentieren haben RULE1 (subj(NP\_nom),obj(NP\_acc),iobj(NP\_dat)) HGC  
präsentieren haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(als\_Adj)) HGC  
präsentieren haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(PP\_als)) HGC  
präsentieren haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_auf\_dat)) SADA  
präsentieren haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_von\_dat)) SADA  
präsentieren haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) SADA  
präsentieren haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_auf\_dat)) SADA

PORTARE

-- bringen SC:4(5)  
bringen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADA  
bringen haben RULE1 (subj(NP\_nom),obj(NP\_acc),iobj(NP\_dat)) SADA  
bringen haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(PP\_als)) HGC  
bringen haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_aus\_dat)) SADA  
bringen haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_zu\_dat)) HGC

-- tragen SC:4+1refl



tragen haben SADAW (subj(NP\_nom)) HGC  
 tragen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW  
 tragen haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_mit\_dat)) SADAW  
 tragen haben RULE1 (subj(NP\_nom),obj(NP\_acc),mod-dir(AdvP)) HGC  
 tragen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) SADAW

**PROVARE**  
 -- üben/proben SC:3  
 proben haben SADAW (subj(NP\_nom)) HGC  
 proben haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW  
 proben haben SADAW (subj(NP\_nom),mod-loc(AdvP)) HGC

-- versuchen SC:1+2(4)refl  
 versuchen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW  
 versuchen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc)) SADAW  
 versuchen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_an\_dat)) HGC  
 versuchen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_in\_dat)) HGC  
 versuchen haben RULE1 (subj(NP\_nom),arg(PRON\_refl-acc),p-obj(PP\_mit\_dat)) SADAW

-- spüren/empfinden SC:3/SC:2(3)  
 spüren haben SADAW (subj(NP\_nom)) HGC  
 spüren haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW  
 spüren haben RULE1 (subj(NP\_nom),obj(NP\_acc),p-obj(PP\_von\_dat)) SADAW

empfinden haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW  
 empfinden haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(als\_Adj)) HGC  
 empfinden haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(PP\_als)) HGC

**REALIZZARE**  
 -- realisieren SC:1  
 realisieren haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW

-- begreifen SC:3  
 begreifen haben SADAW (subj(NP\_nom)) HGC  
 begreifen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) HGC  
 begreifen haben RULE1 (subj(NP\_nom),obj(NP\_acc),obj-pred(PP\_als)) HGC

**ARRIVARE**  
 -- erreichen SC:1  
 erreichen haben RULE1 (subj(NP\_nom),obj(NP\_acc)) SADAW

-- ankommen SC:6(8)  
 an#kommen sein RULE2 (subj(NP\_nom)) SADAW  
 an#kommen sein RULE2 (subj(NP\_nom),arg(NP\_acc)) HGC  
 an#kommen sein RULE2 (subj(NP\_nom),iobj(NP\_dat)) SADAW  
 an#kommen sein RULE2 (subj(NP\_nom),p-obj(PP\_auf\_acc)) SADAW

an#kommen sein RULE2 (subj(NP\_nom),p-obj(PP\_bei\_dat)) SADAW  
an#kommen sein RULE2 (subj(NP\_nom),p-obj(PP\_mit\_dat)) SADAW  
an#kommen sein RULE2 (subj(PRON\_dummy),p-obj(PP\_auf\_acc)) HGC  
an#kommen sein RULE2 (subj(NP\_nom),mod-loc(AdvP)) HGC