Optimality Theory as a Framework for Lexical Acquisition

Thierry Poibeau

Optimality Theory as a Framework for Lexical Acquisition

Thierry Poibeau

Laboratoire LATTICE
PSL*: Paris Sciences et Lettres ⋆
1 rue Maurice Arnoux
92120 Montrouge France
thierry.poibeau@ens.fr

Abstract. This paper re-investigates a lexical acquisition system initially developed for French. We show that, interestingly, the architecture of the system reproduces and implements the main components of Optimality Theory. However, we formulate the hypothesis that some of its limitations are mainly due to a poor representation of the constraints used. Finally, we show how a better representation of the constraints used would yield better results.

1 Introduction

Natural Language Processing (NLP) aims at developing techniques for processing natural language texts using computers. In order to yield accurate results, NLP requires resources containing various information (sub-categorization frames, semantic roles, selection restrictions, etc.). Unfortunately, such resources are not available for most languages and are very costly to develop manually. A recent trend of research has tried to overcome these limitations through the development of automatic acquisition methods from corpora.

Automatic lexical acquisition is an engineering task aiming at providing comprehensive—even if not fully accurate—resources for NLP. As natural languages are complex, lexical acquisition needs to take into account a wide range of parameters and constraints. However, surprisingly, in the acquisition community, relatively few investigations have been done on the structure of the linguistic constraints themselves, beyond the engineering point of view (but note that this work has been extensively done for parsing, see [1]).

In this paper, we want to take another look at some experiments recently done on the automatic acquisition of lexical resources from textual corpora, more specifically on French. In a way, acquisition is converse to parsing: the task consists, from a surface form, in trying to find an abstract lexical-conceptual structure that justify the surface construction (taking into account the relevant set of constraints for the given language).

⋆ This work has received support of TransferS (laboratoire d?excellence, program "Investissements d'avenir" ANR-10-IDEX-0001-02 PSL* and ANR-10-LABX-0099)

Here, in order to get a tractable model, we limit ourselves to the acquisition of sub-categorization frames from corpora. The task is challenging since surface forms incorporate adverbs, modifiers, interpolated clauses and some flexibility in the ordering of the arguments.

Most approaches, including ours, are based on simple filtering techniques. If a complement appears very rarely associated with a given predicate, the acquisition process will assume that this is an incidental co-occurrence that should be left out. However, as we will see, even if this technique is efficient for high frequency items, it leaves a lot of phenomena aside.

Following these observations, we get interested in Optimality Theory (OT). OT is based on a number of assumptions which are absolutely relevant for the lexical acquisition context [2–4]:

- Linguistic well-formedness is relative, not absolute. Perfect satisfaction of all linguistic constraints is attained rarely, and perhaps never.
- Linguistic well-formedness is a matter of comparison or competition among candidate output forms (none of which is perfect).
- Linguistic constraints are ranked and violable. Higher ranking constraints can compel violation of lower ranking constraints. Violation is minimal, however. And even low ranking constraints can make crucial decisions about the winning output candidate.
- The grammar of a language is a ranking of constraints. Ranking may differ from language to language, even if the constraints do not.

However, despite these observations, OT has been mainly applied to phonology, more rarely to morphology or syntax [5, 1]. In this paper, we would like to show, on a precise example, that OT provides a very competitive framework for sub-categorization acquisition.

In order to apply OT to lexical acquisition, we first need to model all the language properties as constraints. The task consists then in identifying the relevant set of constraints that allow one to map a lexical structure to actual (surface) constructions. Note that the task is highly challenging since constraints interact with each other, must be ranked and can be violated.

2 From Corpus to Resources

2.1 OT and Syntax

OT has been mainly applied to syntax in the framework of the Principles and Parameters (P&P) theory developed by Chomsky [6] as part of his Minimalist Program. The central idea of P&P is that a person's syntactic knowledge can be modeled with two formal mechanisms:

- A finite set of fundamental principles that are common to all languages; e.g., a sentence must always have a subject, even if it is not overtly pronounced.
- A finite set of parameters that determine syntactic variability amongst languages; e.g., a binary parameter that determines whether or not the subject of a sentence must be overtly pronounced.

Within this framework, the goal of linguistics is to identify all the principles and parameters that are universal to human languages (i.e. what defines the Universal Grammar).

OT provides a nice framework to implement P&P since the formalism is constraint-based. The input is a set of (universal) abstract candidate forms[1]. Thus, principles and parameters just have to be translated into constraints (CON); then an evaluation function (EVAL) computes the best output given the input and the set of constraints (the principles and parameters) for a given language.

To summarize, here are the three main components of OT: GEN (+input), CON and EVAL.

- GEN takes a series of surface forms and generates an infinite number of candidates, or possible realizations of that input. A language's grammar (its ranking of constraints) determines which of the infinite candidates will be assessed as optimal by EVAL.
- CON includes the set of constraints to be used to determine which of the input candidates is the most likely to be accepted.
- EVAL determines the best analysis among input candidates, taking into account the set of constraints CON. Given two candidates, A and B, A is better than B on a constraint hierarchy if A incurs fewer violations than B. Candidate A is better than B on an entire constraint hierarchy if A incurs fewer violations of the highest-ranked constraint distinguishing A and B. A is optimal in its candidate set if it is better on the constraint hierarchy than all other candidates.

However, the task here is slightly different (converse) since we try to find the best underlying representation from the output (a given utterance), more precisely, we try to learn syntactic frames from data.

2.2 Learning Syntactic Frames from Raw Data

As already said, comprehensive and accurate lexical resources are key components of Natural Language Processing (NLP) systems. Hand-crafting lexical resources is difficult and extremely labour-intensive— particularly as NLP systems require statistical information about the behavior of lexical items in context, and this statistical information changes from one domain to the other. For this reason automatic acquisition of lexical resources from corpora has become increasingly popular.

One of the most useful lexical information for NLP is that related to the predicate-argument structure. The sub-categorization frames (SCFs) of a predicate capture the different combinations of arguments that a given predicate can take. For example, in French, the verb "*acheter*" (*to buy*) sub-categorizes for a subject, a direct object and an indirect object (a prepositional phrase governed by the preposition *"à"*). This can be formalized as follows: *N0 acheter N1 à N2*.

[1] This point, which is much controversial, is based on the assumption that linguistic principles—in P&P Theory—are supposed to be universal. There is a huge literature on this hypothesis that we will not address in this paper. We do not claim any universal feature in this work; we just use OT as an interesting framework for modeling the constraints used.

Sub-categorization lexicons can benefit many NLP applications. For example, they can be used to enhance tasks such as parsing [7, 8] and semantic classification [9] as well as applications such as information extraction [10] and machine translation. They also make it possible to infer large multilingual semantic classifications [11].

Several sub-categorization lexicons are available for many languages, but most of them have been built manually. For French these include the large French dictionary *"Le Lexique Grammaire"* [12] and the more recent *Lefff* [13] and *Dicovalence* (`http://bach.arts.kuleuven.be/dicovalence/`) lexicons.

Some work has been conducted on automatic sub-categorization acquisition, mostly on English [14–17] but also on other languages, from which German is just one example [18]. This work has shown that although automatically built lexicons are not as accurate and detailed as manually built ones, they can be useful for real-world tasks. This is mostly because they provide what manually built resources do not generally provide: statistical information about the likelihood of SCFs for individual verbs.

In what follows, we show that statistical information, in order to yield accurate results, must take into consideration a huge number of constraints. First experiments have given interesting results but the nature and the structure of constraints must be further explored in order to strengthen the existing results. We show that OT provides an interesting framework to identify and structure the set of relevant constraints.

2.3 Introducing Gradience in Lexical Acquisition

As for most linguistic questions, there is no well-established definition of what to include in a SCF, but everybody agrees that a SCF should minimally include the number and the type of the complements depending on the verb (or more generally on the predicative item considered, since adjectives and nouns can also have a SCF). Most authors agree on the fact that complements should be divided between arguments and adjuncts but the distinction between these two categories is far from obvious. Some linguistic tests exist (can the complement be deleted without changing the meaning of the sentence? Can it be moved easily? Can it be pronominalized? etc.) but none of these tests is sufficient or discriminatory enough.

As outlined by Manning [19] "rather than maintaining a categorical argument / adjunct distinction and having to make in/out decisions about such cases, we might instead try to represent SCF information as a probability distribution over argument frames, with different verbal dependents expected to occur with a verb with a certain probability". For example, from the analysis of a large news corpus, one can observe that the French verb *venir (to come)* accepts the frame *PP[de (from)]* with a relative frequency of 59.1% whereas it accepts the frame *PP[à (to)]* with a relative frequency of 5%. This phenomenon can be seen as a kind of selectional "preference" of certain verbs for certain SCFs; the link with more semantic information remains to be done.

It is well known that the evaluation of probability distributions is difficult, since it is by definition dependent on a given corpus. Hand-crafted dictionaries generally do not include any frequency information. Moreover, very few lexical acquisition frameworks currently integrate an efficient way to deal with various phenomena such as multiword expressions (especially light verb constructions and semi-idiomatic expressions), complement optionality, etc. Therefore, current approaches have a tendency to produce two

many SCFs for a given items (semi-idiomatic expressions should be recognized as such and should not be added as new SCFs associated with head verbs, optionality should be handled to reduce the number of partial SCFs).

In the next section, we briefly present a state-of-the art system for French and its limitations; we show that the acquisition model corresponds to OT but does not take into consideration a precise enough set of constraints. We then make some proposals in order to get better results using a finer grain model of constraints.

3 ASSCI, A State-of-the Art Subcategorization Acquisition System for French

A system for the automatic acquisition of sub-categorization frames has recently been implemented for French. This system called ASSCI is capable of acquiring large scale lexicons from un-annotated corpora [20, 21].

This system is close to other systems developed for example for English [16, 22] in that it extracts SCFs from data parsed using a shallow dependency parser [23] and is capable of identifying a large number of SCFs. However, unlike most other systems that accept raw corpus data as input, it does not assume a list of predefined SCFs. The system is based on the assumption that the most relevant SCF corresponding to a given surface form will directly emerge from the application of the constraints on the various candidates, as postulated by OT.

ASSCI takes raw corpus data as input. Input text is first tagged and syntactically analyzed. Then, the system generates a list of candidate SCFs for each verb that occurs frequently enough in data (in the default setting, 200 occurrences of a given verb are necessary). *ASSCI* consists of three modules: a pattern extractor which extracts patterns for each target verb; a SCF builder which builds a list of candidate SCFs per verb (GEN), and a SCF filter (EVAL) which filters out SCFs deemed incorrect according to predefined parameters (CON). They are described briefly in the following sections. For a more detailed description of *ASSCI*, see [20].

3.1 Preprocessing : Morphosyntactic Tagging and Syntactic Analysis

The system first tags and lemmatizes corpus data using *TreeTagger* and then parses it thanks to *Syntex* [23]. *Syntex* is a shallow parser for French. It uses a combination of heuristics and statistics to find dependency relations between tokens in a sentence. It is a relatively accurate parser, e.g. it obtained the best precision and F-measure for written French text in the first EASY evaluation campaign (2006).

The below example illustrates the dependency relations detected by *Syntex* (2) for the input sentence in (1):

(1) La sécheresse s' abattit sur le Sahel en 1972-1973 .
(The drought came down on Sahel in 1972-1973.)

(2) DetFS|le|La|1|DET;2|
NomFS|sécheresse|sécheresse|2|SUJ;4|DET;1

```
Pro|se|s'|3|REF;4|
VCONJS|abattre|abattit|4|SUJ;2,REF;3,PREP;5,PREP;8
Prep|sur|sur|5|PREP;4|NOMPREP;7
DetMS|le|le|6|DET;7|
NomMS|sahel|Sahel|7|NOMPREP;5|DET;6
Prep|en|en|8|PREP;4|NOMPREP;9
NomXXDate|1972-1973|1972-1973|9|NOMPREP;8|
Typo|.|.|10||
```

Syntex does not make a distinction between arguments and adjuncts - rather, each dependency of a verb is attached to the verb.

3.2 Producing the Input (the Pattern Extractor)

The pattern extractor collects the dependencies found by the parser for each occurrence of a target verb. Some cases receive special treatment in this module. For example, if the pronoun *"se"* is one of the dependencies of a verb, the system considers this verb like a new one. In `(1)`, the pattern will correspond to *"s'abattre"* and not to *"abattre"*. If a preposition is the head of one of the dependencies, the module explores the syntactic analysis to find if it is followed by a noun phrase (`+SN]`) or an infinitive verb (`+SINF]`). `(3)` shows the output of the pattern extractor for the input in `(1)`.

```
(3) VCONJS|s'abattre :
Prep+SN|sur|PREP__Prep+SN|en|PREP
```

3.3 GEN (the SCF Builder)

The SCF builder extracts SCF candidates for each verb from the output of the pattern extractor and calculates the number of corpus occurrences for each SCF and verb combination. The syntactic constituents used for building the SCFs are the following:

1. `SN` for nominal phrases;
2. `SINF` for infinitive clauses;
3. `SP[`*prep*`+SN]` for prepositional phrases where the preposition is followed by a noun phrase. *prep* is the head preposition;
4. `SP[`*prep*`+SINF]` for prepositional phrases where the preposition is followed by an infinitive verb. *prep* is the head preposition;
5. `SA` for adjectival phrases;
6. `COMPL` for subordinate clauses.

When a verb has no dependency, its SCF is considered as `INTRANS`.
`(4)` shows the output of the SCF builder for `(1)`.

```
(4) S'ABATTRE+s'abattre ;;; SP[sur+SN]_SP[en+SN]
```

3.4 CON and EVAL (SCF Filter)

Each step of the process is fully automatic, so the output of the SCF builder is noisy due to tagging, parsing or other processing errors. It is also noisy because of the difficulty of the argument-adjunct distinction. The latter is difficult even for humans.

Many criteria that have been defined are not usable in our case because they either depend on lexical information which the parser cannot make use of (since the task is to acquire this information) or on semantic information which even the best parsers cannot yet learn reliably. The approach here is based on the assumption that true arguments tend to occur in argument positions more frequently than adjuncts. Thus many frequent SCFs in the system output are correct.

The strategy is then to filter low frequency entries from the SCF builder output. This is done using the maximum likelihood estimates [24]. This simple method involves calculating the relative frequency of each SCF (for a verb) and comparing it to an empirically determined threshold. The relative frequency of the SCF *i* with the verb *j* is calculated as follows:

$$rel\_freq(scf_i, verb_j) = \frac{|scf_i, verb_j|}{|verb_j|}$$

$|scf_i, verb_j|$ is the number of occurrences of the SCF *i* with the verb *j* and $|verb_j|$ is the total number of occurrences of the verb *j* in the corpus.

If, for example, the frequency of the SCF `SP[sur+SN]_SP[en+SN]` is below the empirically defined threshold, the SCF is rejected by the filter. The MLE filter is not perfect because it is based on rejecting low frequency SCFs. Although relatively more low than high frequency SCFs are incorrect, sometimes rejected frames are correct. The filter incorporates special heuristics for cases where this assumption tends to generate too many errors. With prepositional SCFs involving one PP or more, the filter determines which one is the less frequent PP. It then re-assigns the associated frequency to the same SCF without this PP.

For example, `SP[sur+SN]_SP[en+SN]` could be split to 2 SCFs : `SP[sur+SN]` and `SP[en+SN]`. In this example, `SP[en+SN]` is the less frequent prepositional phrase and the final SCF for the sentence `(1)` is `(5)`.

`(5) SP[sur+SN]`

Note that `SP[en+SN]` is here an adjunct.

4 Some Limitations of this Approach

This approach is very efficient to deal with large corpora. However, some issues remain. As the approach is based on automatic tools (especially parsers) that are far from perfect, the obtained resources always contain errors and have to be manually validated. Moreover, the system needs to get enough examples to be able to infer relevant information. Therefore, there is generally a lack of information for a lot of low productivity items (the famous “sparsity problem”).

More fundamentally, some constructions are difficult to acquire and characterize automatically. On the one hand, idioms are not recognized as such by most acquisition systems. On the other hand, some adjuncts appear frequently with certain verbs (eg. some verbs like *dormir – to sleep* – frequently appear with location complements). The system then assumes that these are arguments, whereas linguistic theory would say without any doubt that these are adjuncts. Lastly, surface cues are sometimes insufficient to recognize ambiguous constructions (cf. *...manger une glace à la vanille...* vs *...manger une glace à la terrasse d'un café... — to eat a vanilla ice-cream* vs *to eat an ice-cream at an outdoor cafe*).

In a traditional architecture, the filtering process incorporates in one modules the set of constraints (CON) and the evaluation function (EVAL). This makes the system less readable than if the constraints were modeled apart from the EVAL function. There is thus a need to refine the set of constraints

5 A Solution: Provide an Explicit Modeling of the Set of Constraints (CON)

We have shown in the previous section that a part of the errors produced were due to an over-simplification of the initial model. It is thus necessary to take other parameters into considerations in order to yield better results. This can be done by refining the set of constraints (CON).

5.1 Refining CON

The issues we have reported in the previous section do not mean that automatic methods are flawed, but they have a number of drawbacks that should be addressed. The acquisition process, based on an analysis of co-occurrences of the verb with its immediate complements (along with filtering techniques) makes the approach highly functional. It is a good approximation of the problem. However, this model does not take into account external constraints.

The analysis of the co-occurrences of the verb with its complement is meaningful but is not sufficient to fully grasp the problem. The fact that some phrasal complements (with a specific head noun) frequently co-occur with a given verb is most of the time useful, especially to identify idioms [25], colligations [26] and light verb constructions [27]. On the other hand, the fact that a given prepositional phrase appear with a large number of verbs may indicate that the preposition introduces an adjunct rather than an argument.

So, instead of simply capturing the co-occurrences of a verb with its complements, a number of important features should be taken into account:

- indicator of the dispersion of the prepositional phrases (PP) depending on the nature of the preposition (if a PP with a given preposition appears with a wide range of different verbs, it is more likely to be a modifier);
- indicator of the co-occurrence of the PP depending on the nature of the head noun (if a verb appears frequently with the same PP frame, it is more likely to form a semi-idiomatic expression);

– indicator of the complexity of the sentence to be processed (if a sentence is complex, its analysis is less reliable).

In order to do this, the pattern extractor has to be modified in order to keep most of the information that were previously rejected as not relevant. These indicators then need to be calculated so as to be taken into account by EVAL.

5.2 Modifying EVAL

All the constraints can be evaluated separately, so as to obtain for each of them an ideal evaluation of the parameter. There are two ways of doing this: *i*) by automatically inferring the different weights from a set of annotated data or *ii*) by estimating the results of various manually defined weights. We are currently using this last method since data annotation is very costly. However, the first approach would certainly lead to more accurate results.

The weight and the ranking of the different constraints must then be examined. A linear model can provide a first approximation but there are surely better ways to integrate the different constraints. Some studies provide some cues but they need to be proper evaluated in order to be integrated in this framework [5].

5.3 Manual Validation

Lastly, the approach requires a manual validation. Rather than leaving the validation process apart for further examination by a linguist, we propose to integrate it in the acquisition process itself. Taking into consideration the number of examples and the complexity of the sentences used for training, it is possible to associate confidence scores with the different constructions of a given verb: the linguist is then able to quickly focus on the most problematic cases. It is also possible to propose tentative constructions to the linguist, when not enough occurrences are available for training. In the end, when too few examples are available, the linguist can provide relevant information to the machine. However, with a well-designed and dynamic validation process, it is possible to obtain accurate and comprehensive lexicons, using only a small fraction of the time that would be necessary to manually develop a lexicon from scratch.

6 Conclusion

Tn this paper, we have proposed a new approach for the automatic acquisition of lexical knowledge from corpora using Optimality Theory. Using this model, it is possible to represent a large part of the language activity through constraints. We have shown that the individual evaluation of each constraint yields very accurate and precise results.

An implementation of this model is currently being done for Japanese [28]. The model provides a better integration of the linguistic contraints within the automatic processing system. First results were competitive with other approaches while providing a more accurate linguistic description.

References

1. Aarts, B.: Syntactic Gradience: The Nature of Grammatical Indeterminacy. Oxford University Press, Oxford (2008)
2. Kager, R.: Optimality Theory. Cambridge University Press, Cambridge (1999)
3. McCarthy, J.: Doing Optimality Theory. Blackwell, Oxford (2008)
4. Prince, A., Smolensky, P.: Optimality Theory: Constraint Interaction in Generative Grammar. Blackwell, Oxford (2004)
5. Blache, P., Prost, J.P.: A Quantification Model of Grammaticality. Constraints Solving and Language Processing (2008)
6. Chomsky, N.: The Minimalist Program. The MIT Press, Cambridge, MA (1995)
7. John Carroll, G.M., Briscoe, T.: Can subcategorisation probabilities help a statistical parser? In: Proceedings of the 6th ACL/SIGDAT Workshop on Very Large Corpora, Montreal (Canada) (1998)
8. Arun, A., Keller, F.: Lexicalization in crosslinguistic probabilistic parsing: The case of French. In: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05), Ann Arbor, Michigan, Association for Computational Linguistics (June 2005) 306–313
9. Schulte im Walde, S., Brew, C.: Inducing German Semantic Verb Classes from Purely Syntactic Subcategorisation Information. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA (2002) 223–230
10. Surdeanu, M., Harabagiu, S.M., Williams, J., Aarseth, P.: Using Predicate-Argument Structures for Information Extraction. In: Proceedings of the Association of Computational Linguistics (ACL). (2003) 8–15
11. Sun, L., Korhonen, A., Poibeau, T., Messiant, C.: Investigating the Cross-linguistic Potential of VerbNet-style Classification. In: Proceedings of the 23rd International Conference on Computational Linguistics. COLING '10, Stroudsburg, PA, USA (2010) 1056–1064
12. Gross, M.: Méthodes en syntaxe. Hermann, Paris (1975)
13. Sagot, B., Clément, L., de La Clergerie, E., Boullier, P.: The Lefff 2 Syntactic Lexicon for French: Architecture, Acquisition, Use. In: Language Resource and Evaluation Conference (LREC), Genoa (2006)
14. Brent, M.R.: From Grammar to Lexicon: Unsupervised Learning of Lexical Syntax. Computational Linguistics **19** (1993) 203–222
15. Manning, C.D.: Automatic Acquisition of a Large Subcategorization Dictionary from Corpora. In: Proceedings of the Meeting of the Association for Computational Linguistics. (1993) 235–242
16. Briscoe, T., Carroll, J.: Automatic Extraction of Subcategorization from Corpora. In: Proceedings of the 5th ACL Conference on Applied Natural Language Processing, Washington, DC. (1997) 356–363
17. Korhonen, A., Krymolowski, Y., Briscoe, T.: A Large Subcategorization Lexicon for Natural Language Processing Applications. In: Proceedings of the 5th international conference on Language Resources and Evaluation, Genova, Italy (2006)
18. Schulte im Walde, S.: A Subcategorisation Lexicon for German Verbs induced from a Lexicalised PCFG. In: Proceedings of the 3rd Conference on Language Resources and Evaluation. Volume IV., Las Palmas de Gran Canaria, Spain (2002) 1351–1357
19. Manning, C.D.: Probabilistic syntax. In Press, M., ed.: Probabilistic Linguistics, R. Bod, J. Hay, S. Jannedy (2003) 289–341
20. Messiant, C.: ASSCI : A Subcategorization Frames Acquisition System For French. In: Proceedings of the Association for Computational Linguistics (ACL) Student Research Workshop, Colombus, Ohio, Association for Computational Linguistics (2008)

21. Messiant, C., Korhonen, A., Poibeau, T.: LexSchem: A Large Subcategorization Lexicon for French Verbs. In: Proceedings of the Language Resource and Evaluation Conference, Maroc (2008) sans pagination
22. Preiss, J., Briscoe, T., Korhonen, A.: A System for Large-Scale Acquisition of Verbal, Nominal and Adjectival Subcategorization Frames from Corpora. In: Proceedings of the Meeting of the Association for Computational Linguistics, Prague (2007) 912–918
23. Bourigault, D., Jacques, M.P., Fabre, C., Frérot, C., Ozdowska, S.: Syntex, analyseur syntaxique de corpus. In: Actes des 12èmes journées sur le Traitement Automatique des Langues Naturelles, Dourdan (2005)
24. Korhonen, A., Gorrell, G., McCarthy, D.: Statistical Filtering and Subcategorization Frame Acquisition. In: Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, Hong Kong (2000)
25. Fabre, C., Bourigault, D.: Exploiter des corpus annots syntaxiquement pour observer le continuum entre arguments et circonstants. Journal of French Language Studies **18**(1) (2008) 87–102
26. Firth, J.R.: Descriptive Linguistics and the Study of English. In: Selected Papers of John R. Firth. (1968)
27. Butt, M.: The Light Verb Jungle. Harvard Working Papers in Linguistics **9** (2003) 1–49
28. Marchal, P., Poibeau, T., Lepage, Y.: Representing the Continuum between Arguments and Adjuncts within Predicate-Frames. In: NINJAL International Symposium on "Valency Classes and Alternations in Japanese", Tokyo (2012)